Learning Anaphoricity and Antecedent Ranking Features for Coreference Resolution

Sam Wiseman[1] **Alexander M. Rush**[1,2] **Stuart M. Shieber**[1] **Jason Weston**[2]
[1]School of Engineering and Applied Sciences
Harvard University
Cambridge, MA, USA
{swiseman,srush,shieber}@seas.harvard.edu
[2]Facebook AI Research
New York, NY, USA
jase@fb.com

Abstract

We introduce a simple, non-linear mention-ranking model for coreference resolution that attempts to learn distinct feature representations for anaphoricity detection and antecedent ranking, which we encourage by pre-training on a pair of corresponding subtasks. Although we use only simple, unconjoined features, the model is able to learn useful representations, and we report the best overall score on the CoNLL 2012 English test set to date.

1 Introduction

One of the major challenges associated with resolving coreference is that in typical documents the number of mentions (syntactic units capable of referring or being referred to) that are *non-anaphoric* – that is, that are not coreferent with any previous mention – far exceeds the number of mentions that are anaphoric (Kummerfeld and Klein, 2013; Durrett and Klein, 2013).

This preponderance of non-anaphoric mentions makes coreference resolution challenging, partly because many basic coreference features, such as those looking at head, number, or gender match fail to distinguish between truly coreferent pairs and the large number of matching but nonetheless non-coreferent pairs. Indeed, several authors have noted that it is difficult to obtain good performance on the coreference task using simple features (Lee et al., 2011; Fernandes et al., 2012; Durrett and Klein, 2013; Kummerfeld and Klein, 2013; Björkelund and Kuhn, 2014) and, as a result, state-of-the-art systems tend to use linear models with complicated feature conjunction schemes in order to capture more fine-grained interactions. While this approach has shown success, it is not obvious which additional feature conjunctions will lead to improved performance, which is problematic as systems attempt to scale with new data and features.

In this work, we propose a data-driven model for coreference that does not require prespecifying any feature relationships. Inspired by recent work in learning representations for natural language tasks (Collobert et al., 2011), we explore neural network models which take only raw, unconjoined features as input, and attempt to learn intermediate representations automatically. In particular, the model we describe attempts to create independent feature representations useful for both detecting the anaphoricity of a mention (that is, whether or not a mention is anaphoric) and ranking the potential antecedents of an anaphoric mention. Adequately capturing anaphoricity information has long been thought to be an important aspect of the coreference task (see Ng (2004) and Section 7), since a strong non-anaphoric signal might, for instance, discourage the erroneous prediction of an antecedent for a non-anaphoric mention even in the presence of a misleading head match.

We furthermore attempt to encourage the learning of the desired feature representations by pre-training the model's weights on two corresponding subtasks, namely, anaphoricity detection and antecedent ranking of known anaphoric mentions.

Overall our best model has an absolute gain of almost 2 points in CoNLL score over a similar but linear mention-ranking model on the CoNLL 2012 English test set (Pradhan et al., 2012), and of over 1.5 points over the state-of-the-art coreference system. Moreover, unlike current state-of-the-art systems, our model does only local inference, and is therefore significantly simpler.

1.1 Problem Setting

We consider here the mention-ranking (or "mention-synchronous") approach to coreference

resolution (Denis and Baldridge, 2008; Bengtson and Roth, 2008; Rahman and Ng, 2009), which has been adopted by several recent coreference systems (Durrett and Klein, 2013; Chang et al., 2013). Such systems aim to identify whether a mention is coreferent with an antecedent mention, or whether it is instead non-anaphoric (the first mention in the document referring to a particular entity). This is accomplished by assigning a score to the mention's potential antecedents as well as to the possibility that it is non-anaphoric, and then predicting the greatest scoring option. We furthermore assume the more realistic "system mention" setting, where it is not known a priori which mentions in a document participate in coreference clusters, and so (all) mentions must be automatically extracted, typically with the aid of automatically detected parse trees.

Formally, we denote the set of automatically detected mentions in a document by $\mathcal{X}$. For a mention $x \in \mathcal{X}$, let $\mathcal{A}(x)$ denote the set of mentions appearing before x; we refer to this set as x's potential antecedents. Additionally let the symbol ϵ denote the empty antecedent, to which we will view x as referring when x is non-anaphoric.[1] Denoting the set $\mathcal{A}(x) \cup \{\epsilon\}$ by $\mathcal{Y}(x)$, a mention-ranking model defines a scoring function $s(x, y) : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$, and predicts the antecedent of x to be $y^* = \arg\max_{y \in \mathcal{Y}(x)} s(x, y)$.

It is common to be quite liberal when extracting mentions, taking, essentially, every noun phrase or pronoun to be a candidate mention, so as not to prematurely discard those that might be coreferent (Lee et al., 2011; Fernandes et al., 2012; Chang et al., 2012; Durrett and Klein, 2013). For instance, the Berkeley Coreference System (herein BCS) (Durrett and Klein, 2013), which we use for mention extraction in our experiments, recovers approximately 96.4% of the truly anaphoric mentions in the CoNLL 2012 training set, with an almost 3.5:1 ratio of non-anaphoric mentions to anaphoric mentions among the extracted mentions.

2 Mention Ranking Models

The structural simplicity of the mention-ranking framework puts much of the burden on the scoring function $s(x, y)$. We begin by considering mention-ranking systems using linear scoring functions. In the next section, we will extend these models to operate over learned non-linear representations.

Linear mention-ranking models generally utilize the following scoring function

$$s_{\text{lin}}(x, y) \triangleq \boldsymbol{w}^{\mathsf{T}} \boldsymbol{\phi}(x, y) \quad ,$$

where $\boldsymbol{\phi} : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}^d$ is a pairwise feature function defined on a mention and a potential antecedent, and $\boldsymbol{w}$ is a learned parameter vector.

To add additional flexibility to the model, linear mention ranking models may duplicate individual features in $\boldsymbol{\phi}$, with one version being used when predicting an antecedent for x, and another when predicting that x is non-anaphoric (Durrett and Klein, 2013). Such a scheme effectively gives rise to the following piecewise scoring function

$$s_{\text{lin}+}(x, y) \triangleq \begin{cases} \boldsymbol{u}^{\mathsf{T}} \begin{bmatrix} \boldsymbol{\phi}_{\text{a}}(x) \\ \boldsymbol{\phi}_{\text{p}}(x, y) \end{bmatrix} & \text{if } y \neq \epsilon \\ \boldsymbol{v}^{\mathsf{T}} \boldsymbol{\phi}_{\text{a}}(x) & \text{if } y = \epsilon \end{cases} \quad ,$$

where $\boldsymbol{\phi}_{\text{a}} : \mathcal{X} \rightarrow \mathbb{R}^{d_{\text{a}}}$ is a feature function defined on a mention and its context, $\boldsymbol{\phi}_{\text{p}} : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}^{d_{\text{p}}}$ is a pairwise feature function defined on a mention and a potential antecedent, and parameters $\boldsymbol{u}$ and $\boldsymbol{v}$ replace $\boldsymbol{w}$. Above, we have made an explicit distinction between pairwise features ($\boldsymbol{\phi}_{\text{p}}$) and those strictly on x and its context ($\boldsymbol{\phi}_{\text{a}}$), and moreover assumed that our features need not examine potential antecedents when predicting $y = \epsilon$.

We refer to the basic, unconjoined features used for $\boldsymbol{\phi}_{\text{a}}$ and $\boldsymbol{\phi}_{\text{p}}$ as *raw* features. Figure 2 shows two versions of these features, a base set BASIC and an extended set BASIC+. The BASIC set are the raw features used in BCS, and BASIC+ includes additional raw features used in other recent coreference sytems. For instance, BASIC+ additionally includes features suggested by Recasens et al. (2013) to be useful for anaphoricity, such as the number of a mention, its named entity status, and its animacy, as well as number and gender information. We additionally include bilexical head features, which are used in many well-performing systems (for instance, that of Fernandes et al. (2012)).

2.1 Problems with Raw Features

Many authors have observed that, taken individually, raw features tend to not be particularly predictive for the coreference task. We examine this phenomenon empirically in Figure 1. These

[1] We make this stipulation for modeling convenience; it is not intended to reflect any linguistic fact.

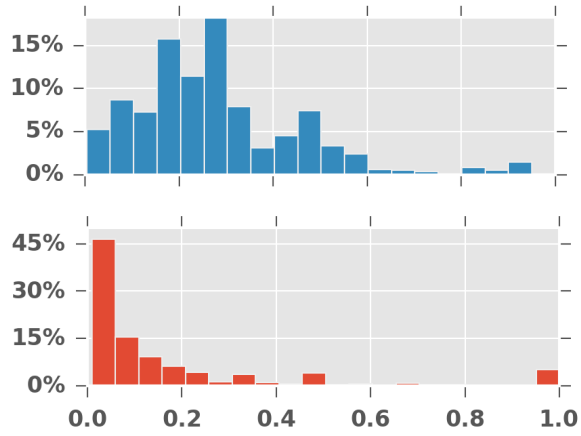

Figure 1: Two histograms illustrating the predictive ability of raw (unconjoined) features per feature occurrence: (top) mention-context features from ϕ_{a} as independent predictors of anaphoricity ($y \neq \epsilon$), and (bottom) antecedent-mention features from ϕ_{p} as independent predictors of coreferent mentions. Very few raw features are strong indicators of either anaphoricity or an antecedent match. Data taken from the CoNLL development set.

graphs show that the vast majority of individual features do not give a strong positive signal either of anaphoricity or for an antecedent match.

To address this issue, state-of-the-art mention-ranking systems often rely on manual or otherwise induced conjunction schemes to capture specific feature interactions. Durrett and Klein (2013), for instance, conjoin all raw features in ϕ_{a} with the *type* of the mention x, and all raw features in ϕ_{p} with the types of the current mention and antecedent. For these purposes, the *type* of a mention is either "nominal", "proper", or a canonicalization of the pronoun if it is a pronominal mention. Fernandes et al. (2012) and Björkelund and Kuhn (2014) use an automatic but complicated scheme to induce conjunctions by first extracting feature templates from a separately trained decision tree, and then doing greedy forward selection among the templates. These conjunctions add some non-linearity to the scoring function while still maintaining a tractable, though large, feature set.

3 Learning Features for Ranking

As an alternative to the aforementioned feature conjunction schemes, we consider learning feature representations in order to better capture relevant aspects of the task. Representation learning affords the model more flexibility in exploiting feature interactions, although it can make the underlying training problem more difficult.

	Mention Features (ϕ_{a})	
	Feature	Value Set
	Mention Head	$\mathcal{V}$
	Mention First Word	$\mathcal{V}$
	Mention Last Word	$\mathcal{V}$
	Word Preceding Mention	$\mathcal{V}$
	Word Following Mention	$\mathcal{V}$
	# Words in Mention	$\{1, 2, \ldots\}$
	Mention Synt. Ancestry	see BCS (2013)
	Mention Type	$\mathcal{T}$
+	Mention Governor	$\mathcal{V}$
+	Mention Sentence Index	$\{1, 2, \ldots\}$
+	Mention Entity Type	NER tags
+	Mention Number	{sing.,plur.,unk}
+	Mention Animacy	{an.,inan.,unk}
+	Mention Gender	{m,f,neut.,unk}
+	Mention Person	{1,2,3,unk}

	Pairwise Features (ϕ_{p})	
	Feature	Value Set
	BASIC features on Mention	see above
	BASIC features on Antecedent	see above
	Mentions between Ment., Ante.	$\{0 \ldots 10\}$
	Sentences between Ment., Ante.	$\{0 \ldots 10\}$
	i-within-i	{T,F}
	Same Speaker	{T,F}
	Document Type	{Conv.,Art.}
	Ante., Ment. String Match	{T,F}
	Ante. contains Ment.	{T,F}
	Ment. contains Ante.	{T,F}
	Ante. contains Ment. Head	{T,F}
	Mention contains Ante. Head	{T,F}
	Ante., Ment. Head Match	{T,F}
	Ante., Ment. Synt. Ancestries	see above
+	BASIC+ features on Ment.	see above
+	BASIC+ features on Ante.	see above
+	Ante., Ment. Numbers	see above
+	Ante., Ment. Genders	see above
+	Ante., Ment. Persons	see above
+	Ante., Ment., Entity Types	see above
+	Ante., Ment. Heads	see above
+	Ante., Ment. Types	see above

Figure 2: Features used for $\phi_{\text{a}}(x)$ and $\phi_{\text{p}}(x, y)$. The '+' indicates a feature is in BASIC+ feature set. $\mathcal{V}$ denotes the training vocabulary, and $\mathcal{T}$ denotes the set of mention types, viz., {nominal,proper} $\cup$ {canonical pronouns}, as defined in BCS. Conv. and Art. abbreviate conversation and article (resp.). Lexicalized features occurring fewer than 20 times in the training set back off to part-of-speech; bilexical heads occurring fewer than 10 times back off to an indicator feature. Animacy information is taken from a list and rules used in the Stanford Coreference system (Lee et al., 2013).

3.1 Model

We use a neural network to define our model as an extension to the mention-ranking model introduced in Section 2. We consider in particular the scoring function:

$$s(x, y) \triangleq \begin{cases} \boldsymbol{u}^{\mathsf{T}} \boldsymbol{g}(\begin{bmatrix} \boldsymbol{h}_{\text{a}}(x) \\ \boldsymbol{h}_{\text{p}}(x,y) \end{bmatrix}) + u_0 & \text{if } y \neq \epsilon \\ \boldsymbol{v}^{\mathsf{T}} \boldsymbol{h}_{\text{a}}(x) + v_0 & \text{if } y = \epsilon \end{cases},$$

where $\boldsymbol{h}_{\mathrm{a}}$ and $\boldsymbol{h}_{\mathrm{p}}$ are feature representations, non-linear functions of the features $\boldsymbol{\phi}_{\mathrm{a}}$ and $\boldsymbol{\phi}_{\mathrm{p}}$ (respectively), and $\boldsymbol{g}$ is a function of these representations. In particular, we define

$$\begin{aligned}\boldsymbol{h}_{\mathrm{a}}(x) &\triangleq \tanh(\boldsymbol{W}_{\mathrm{a}}\,\boldsymbol{\phi}_{\mathrm{a}}(x) + \boldsymbol{b}_{\mathrm{a}}) \\ \boldsymbol{h}_{\mathrm{p}}(x,y) &\triangleq \tanh(\boldsymbol{W}_{\mathrm{p}}\,\boldsymbol{\phi}_{\mathrm{p}}(x,y) + \boldsymbol{b}_{\mathrm{p}}) \quad ,\end{aligned}$$

and we take $\boldsymbol{g}$ to either be the identity function, in which case the above model is analogous to $s_{\mathrm{lin}+}$ but defined over non-linear feature representations, or to be an additional hidden layer: $\boldsymbol{g}(\left[\begin{smallmatrix}\boldsymbol{h}_{\mathrm{a}}(x)\\ \boldsymbol{h}_{\mathrm{p}}(x,y)\end{smallmatrix}\right]) = \tanh(\boldsymbol{W}\left[\begin{smallmatrix}\boldsymbol{h}_{\mathrm{a}}(x)\\ \boldsymbol{h}_{\mathrm{p}}(x,y)\end{smallmatrix}\right] + \boldsymbol{b})$. For ease of exposition, we will refer to these two settings of $\boldsymbol{g}$ as $\boldsymbol{g}_1$ and $\boldsymbol{g}_2$ (respectively) in what follows. As we will see below, both settings lead to comparable performance, but to a different error distribution.

In either case, by defining the functions $\boldsymbol{h}_{\mathrm{a}}$ and $\boldsymbol{h}_{\mathrm{p}}$, we allow the model to learn representations of the input features $\boldsymbol{\phi}_{\mathrm{a}}$ and $\boldsymbol{\phi}_{\mathrm{p}}$. The benefit of the added non-linearities is that, in theory, it is no longer necessary to explicitly specify feature conjunctions, since the model may learn them automatically if necessary. Accordingly, for this model we use only $\boldsymbol{\phi}_{\mathrm{a}}$ and $\boldsymbol{\phi}_{\mathrm{p}}$ consisting of the raw features in Figure 2 without conjunctions. Any interaction between these features must be learned by the feature representations $\boldsymbol{h}_{\mathrm{p}}$ and $\boldsymbol{h}_{\mathrm{a}}$.

3.2 Training

We can directly train our model using back-propagation. To specify the training problem, we first define notation for the training objective.

Define the set $\mathcal{C}(x)$ to contain just the mentions in $\mathcal{A}(x)$ that are coreferent with x. We then define

$$\mathcal{C}'(x) = \begin{cases}\mathcal{C}(x) & \text{if } x \text{ is anaphoric} \\ \{\epsilon\} & \text{otherwise} \end{cases} .$$

Finally, let $y_n^{\ell} = \arg\max_{y \in \mathcal{C}'(x_n)} s(x_n, y)$ be the highest scoring correct antecedent of x_n, which may be ϵ. (Thus, following recent work (Yu and Joachims, 2009; Fernandes et al., 2012; Chang et al., 2013; Durrett and Klein, 2013), we view each mention as having a "latent antecedent".[2]) We train to minimize the regularized, slack-rescaled, latent-variable loss[3] given by:

$$L(\boldsymbol{\theta}) = \sum_{n=1}^{N} \max_{\hat{y} \in \mathcal{Y}(x_n)} \Delta(x_n, \hat{y})(1 + s(x_n, \hat{y}) - s(x_n, y_n^{\ell})) + \lambda ||\boldsymbol{\theta}||_1,$$

where Δ is a mistake-specific cost function, which is 0 when $\hat{y} \in \mathcal{C}'(x_n)$. Above, we use $\boldsymbol{\theta}$ to refer to the full set of parameters $\{\boldsymbol{W}, \boldsymbol{u}, \boldsymbol{v}, \boldsymbol{W}_{\mathrm{a}}, \boldsymbol{W}_{\mathrm{p}}, \boldsymbol{b}_{\mathrm{a}}, \boldsymbol{b}_{\mathrm{p}}\}$.

For experiments, we define Δ to take on different costs for the three kinds of mistakes possible in a coreference task, as follows:

$$\Delta(x, \hat{y}) = \begin{cases}\alpha_1 & \text{if } \hat{y} \neq \epsilon \wedge \epsilon \in \mathcal{C}'(x) \\ \alpha_2 & \text{if } \hat{y} = \epsilon \wedge \epsilon \notin \mathcal{C}'(x) \\ \alpha_3 & \text{if } \hat{y} \neq \epsilon \wedge \hat{y} \notin \mathcal{C}'(x) \end{cases} .$$

The α_i determine the trade-off between these mistakes (and thus precision and recall). Adopting the terminology of BCS, we refer to these mistakes as "false link" (FL), "false new" (FN), and "wrong link" (WL), respectively.

4 Representations from Subtasks

While we could train our full model directly, it is known to be difficult to train high performing non-convex neural-network models from a random initialization (Erhan et al., 2010). In order to overcome the problems associated with training from this setting, and to learn feature representations useful for the full coreference task, we pretrain subparts of the model on the subtasks targeting the desired feature representations. We then train the entire model on the full coreference task (from the pre-trained initializations). As we will see, the pre-training scheme outlined below helps the model achieve improved performance.

The proposed pre-training scheme involves learning the parameters associated with $\boldsymbol{h}_{\mathrm{a}}$ and $\boldsymbol{h}_{\mathrm{p}}$ using two natural subtasks: anaphoricity detection and antecedent ranking. In particular, we (1) train $\boldsymbol{h}_{\mathrm{a}}$ on the task of predicting whether a particular mention is anaphoric or not, and (2) train $\boldsymbol{h}_{\mathrm{p}}$ on the task of predicting the antecedent of mentions known to be anaphoric.

4.1 Anaphoricity Detection

For the first subtask we attempt to predict whether a mention is anaphoric or not based only on its

[2] Note that this renders the objectives of even models with a linear scoring function non-convex.

[3] Previous work divides between log-loss and margin loss. We use the latter because gradient updates (within backprop) for the non-probabilistic objectives only involve terms relating to $\hat{y}$ and y_n^{ℓ}, and are therefore faster.

Feat. (Conj.)	Model	Anaphoric P	Anaphoric R	Anaphoric F_1	Ante Acc.
BASIC (N)	Lin.	74.15	74.20	74.18	69.10
BASIC (Y)	Lin.	73.98	75.04	74.51	79.76
BASIC (N)	NN	75.30	75.36	75.33	81.65
BASIC+ (N)	Lin.	74.14	74.71	74.43	74.02
BASIC+ (Y)	Lin.	74.24	75.39	74.81	80.44
BASIC+ (N)	NN	75.84	76.02	75.93	82.86

Table 1: Performance of the two subtasks on the CoNLL 2012 development set by feature set and model type. "Conj." indicates whether conjunctions are used. The linear anaphoric system is an SVM (LibLinear implementation (Fan et al., 2008)), and the linear antecedent system is a linear model with the margin-based objective.

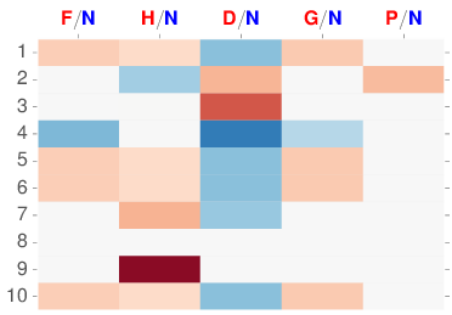

Figure 3: Visualization of the representation matrix $\boldsymbol{W}_{\mathrm{p}}$. A subset of the raw features were manually grouped into five classes indicating: full lexical match [F], head match [H], mention/sentence distance [D] (near versus far), gender/number match [G], and type [P] (pronoun versus other). The heat map illustrates 10-columns of $\boldsymbol{W}_{\mathrm{p}}$ as a weighted combination of these classes, roughly illustrating the combination of raw features required for this dimension of the representation.

local context.[4] Anaphoricity detection in various forms has been used as an initial step in several coreference systems (Ng and Cardie, 2002; Bengtson and Roth, 2008; Rahman and Ng, 2009; Björkelund and Farkas, 2012), and the related question of whether a mention can be determined to be a *singleton* or not has been explored recently by Recasens et al. (2013), Ma et al. (2014), and others.[5]

Formally, let $t_n \in \{-1, 1\}$ indicate whether $\epsilon \in \mathcal{C}'(x_n)$ or not (respectively). That is, $t_n = 1$ if and only if x_n is anaphoric. Define the subtask scoring function $s_{\mathrm{a}} : \mathcal{X} \to \mathbb{R}$ as

$$s_{\mathrm{a}}(x) \triangleq \boldsymbol{v}_{\mathrm{a}}{}^{\mathsf{T}} \boldsymbol{h}_{\mathrm{a}}(x) + \nu_0 \quad ,$$

where the vector $\boldsymbol{v}_{\mathrm{a}}$ and the bias ν_0 are specific to this subtask and are discarded after pre-training.

We train this model to minimize the following slack-rescaled objective

$$L_{\mathrm{a}}(\boldsymbol{\theta}_{\mathrm{a}}) = \sum_{n=1}^{N} \Delta_{\mathrm{a}}(t_n)[1 - t_n\, s_{\mathrm{a}}(x_n)]_+ + \lambda ||\boldsymbol{\theta}_{\mathrm{a}}||_1,$$

where Δ_{a} is a class-specific cost used to help encourage anaphoric decisions given the imbalanced data set, and $\boldsymbol{\theta}_{\mathrm{a}} = \{\boldsymbol{v}_{\mathrm{a}}, \boldsymbol{W}_{\mathrm{a}}, \boldsymbol{b}_{\mathrm{a}}\}$ are the parameters of the subtask.

4.2 Antecedent Ranking

For the second subtask, antecedent ranking, we predict the antecedent for mentions known a priori to be anaphoric. This subtask is inspired by the "gold mention" version of the coreference task. Systems designed for this task are forced to handle many fewer non-anaphoric mentions and can often successfully utilize richer feature representations.

The setup for this task is similar to the full coreference problem, except that we discard any mention x_n such that $\epsilon \in \mathcal{C}'(x_n)$. Thus, we define the pairwise scoring function $s_{\mathrm{p}} : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ as

$$s_{\mathrm{p}}(x, y) \triangleq \boldsymbol{u}_{\mathrm{p}}{}^{\mathsf{T}} \boldsymbol{h}_{\mathrm{p}}(x, y) + \upsilon_0 \quad .$$

As before, $\boldsymbol{u}_{\mathrm{p}}$ and υ_0 are discarded after training for this subtask, but we keep the rest of the parameters. For training, we use an analogous latent-variable loss function to that used for the full coreference task, except we replace $\mathcal{C}'$ with $\mathcal{C}$, and the cost $\Delta(x, \hat{y})$ is always 1 (when it is nonzero).

4.3 Subtask Performance

As a preliminary experiment, we train models for these two subtasks using both the BASIC and BASIC+ raw features. Table 1 shows the results. For the first subtask, experiments look at the precision, recall, and F_1 score of predicting anaphoric mentions on the CoNLL 2012 development set. As a baseline we use an L1-regularized SVM implemented using LibLinear (Fan et al., 2008), both using raw features and using features conjoined according to the BCS scheme. For the second subtask, experiments look at the accuracy of the model in predicting the correct antecedent on known anaphoric mentions. As a baseline we use a linear mention ranking model, with and without

[4] While performance on this *subtask* can in fact be improved further by looking at previous mentions, features learned in this way led to inferior performance on the full task.

[5] Note that singleton detection is slightly different from anaphoricity detection, since a mention can be non-anaphoric but not a singleton if it is the first mention in a cluster.

conjunctions, trained using the same margin-based loss.

In both subtasks, the neural network model performs quite well, significantly better than the unconjoined baselines and better than the model trained with manually conjoined features. We provide a visual representation of the antecedent ranking features learned in Figure 3. While the improved subtask performance does not imply better performance on the full coreference task, it shows that model can learn useful feature representations with only raw input features.

5 Coreference Experiments

Our experiments examine performance as compared with other coreference systems, as well as the effect of features, pre-training, and model architecture. We also perform a qualitative comparison of our model with the analogous linear model on some challenging non-anaphoric cases.

5.1 Methods

All experiments use the CoNLL 2012 English dataset (Pradhan et al., 2012), which is based on the OntoNotes corpus (Hovy et al., 2006). The data set contains 3,493 documents consisting of 1.6 million words. We use the standard experimental split with the training set containing 2,802 documents and 156K annotated mentions, the development set containing 343 documents and 19K annotated mentions, and the test set containing 348 documents and 20K annotated mentions. For all experiments, we use BCS (Durrett and Klein, 2013) to extract system mentions and to compute some of the features.

For training, we minimize the loss described above using the composite mirror descent AdaGrad update (Duchi et al., 2011) with document sized mini-batches.[6] We tuned the AdaGrad learning rate and regularization parameters using a grid search over possible learning rates $\eta \in \{0.001, 0.002, 0.01, 0.02, 0.1, 0.2\}$ and over regularization parameters $\lambda \in \{10^{-6}, \ldots, 10^{-1}\}$. For the full coreference task, we use a different learning rate for the pre-trained weights and for the second-layer weights, using $\eta_1 = 0.1$ and $\eta_2 = 0.001$, respectively, and $\lambda = 10^{-6}$. When initializing weight-matrices that were not pre-trained we used the sparse initialization technique proposed by Sutskever et al. (2013). For all experiments we use the cost-weights $\boldsymbol{\alpha} = \langle 0.5, 1.2, 1 \rangle$ in defining Δ.

For the anaphoricity representations the matrix dimensions used are $\boldsymbol{W}_{\mathrm{a}} \in \mathbb{R}^{128 \times d_{\mathrm{a}}}$, and for the pairwise representations the matrix dimensions used are $\boldsymbol{W}_{\mathrm{p}} \in \mathbb{R}^{700 \times d_{\mathrm{p}}}$. In the $\boldsymbol{g}_2$ model, the outer matrix dimensions are $\boldsymbol{W} \in \mathbb{R}^{128 \times (d_{\mathrm{p}} + d_{\mathrm{a}})}$. With the BASIC+ features, d_{p} and d_{a} come out to be slightly less than 10^6 and 10^4, respectively, with bilexical head features accounting for the vast majority of d_{p}.[7] We tuned all hyper-parameters (as well as those of baseline systems) on the development set.

We use the CoNLL 2012 scoring script v8.01[8] (Pradhan et al., 2014; Luo et al., 2014), which scores based on 3 metrics, including MUC (Vilain et al., 1995), CEAF_e (Luo, 2005), and B^3 (Bagga and Baldwin, 1998), as well as the CoNLL score, which is the arithmetic mean of the 3 metrics.

Code implementing our models is available at `https://github.com/swiseman/nn_coref`. The system trains in time comparable to that of linear systems, mainly because we use only raw features and sparse margin-based gradient updates.

5.2 Results

Our main results are shown in Table 2. This table compares the performance of our system with the performance reported by several other state-of-the art systems on the CoNLL 2012 English coreference test set. Our full models achieve the best F_1 score across two of the three metrics and have the best aggregate (CoNLL) score, with an improvement of over 1.5 points over the best reported result, and of almost 2 points over the best mention-ranking system. Our F_1 improvements on all three metrics are significant ($p < 0.05$ under the bootstrap resample test (Koehn, 2004)) as compared with both Björkelund and Kuhn (2014), and Durrett and Klein (2014), the two most recent, state-of-the-art systems.

Since our full models use some additional raw features (although an order of magnitude fewer total features than the comparable conjunction-

[6] In preliminary experiments we also used Nesterov's accelerated gradient (Nesterov, 1983), but found AdaGrad to perform better.

[7] Note that the BCS conjunction scheme, for instance, applied to our raw features gives a d_{p} and d_{a} that are over an order of magnitude larger.

[8] `http://conll.github.io/reference-coreference-scorers/`

System	MUC			B^3			$CEAF_e$			
	P	R	F_1	P	R	F_1	P	R	F_1	CoNLL
BCS (2013)	74.89	67.17	70.82	64.26	53.09	58.14	58.12	52.67	55.27	61.41
Prune&Score (2014)	81.03	66.16	**72.84**	66.9	51.10	57.94	68.75	44.34	53.91	61.56
B&K (2014)	74.3	67.46	70.72	62.71	54.96	58.58	59.4	52.27	55.61	61.63
D&K (2014)	72.73	69.98	71.33	61.18	56.60	58.80	56.20	54.31	55.24	61.79
This work ($\boldsymbol{g}_2$)	76.96	68.10	72.26	66.90	54.12	59.84	59.02	53.34	56.03	62.71
This work ($\boldsymbol{g}_1$)	76.23	69.31	72.60	66.07	55.83	**60.52**	59.41	54.88	**57.05**	**63.39**

Table 2: Results on CoNLL 2012 English test set. We compare against recent state-of-the-art systems, including (in order) Durrett and Klein (2013), Ma et al. (2014), Björkelund and Kuhn (2014), and Durrett and Klein (2014) (rescored with the v8.01 scorer). F_1 gains are significant ($p < 0.05$ under the bootstrap resample test (Koehn, 2004)) compared with both B&K and D&K for all metrics.

Model	Features	MUC	B^3	$CEAF_e$	CoNLL
Lin.		70.44	59.10	55.57	61.71
NN ($\boldsymbol{g}_2$)	BASIC	71.59	60.56	57.45	63.20
NN ($\boldsymbol{g}_1$)		71.86	60.9	57.90	63.55
Lin.		70.92	60.05	56.39	62.45
NN ($\boldsymbol{g}_2$)	BASIC+	72.68	61.70	58.32	64.23
NN ($\boldsymbol{g}_1$)		72.74	61.77	58.63	64.38

Table 3: F_1 performance comparison between state-of-the-art linear mention-ranking model (Durrett and Klein, 2013) and our full models on CoNLL 2012 development set for different feature sets.

Model	MUC	B^3	$CEAF_e$	CoNLL
Fully Conn. 1 Layer	71.80	60.93	57.51	63.41
Fully Conn. 2 Layer	71.77	60.84	57.05	63.22
$\boldsymbol{g}_1$ + RI	71.92	61.06	57.59	63.52
$\boldsymbol{g}_1$ + PT	72.74	61.77	58.63	64.38
$\boldsymbol{g}_2$ + RI	72.31	61.79	58.06	64.05
$\boldsymbol{g}_2$ + PT	72.68	61.70	58.32	64.23

Table 4: Comparison of performance (in F_1 score) of various models on CoNLL 2012 development set using BASIC+ features. "PT" and "RI" refer to pretraining and random initialization respectively. "Fully Conn." refers to baseline fully connected networks. See text for further model descriptions.

based linear model), we are interested in what part of the improvement in performance comes from features rather than modeling power. Table 3 compares the full model to BCS, a system effectively using the $s_{\text{lin}+}$ scoring function together with a manual conjunction scheme, on both BASIC and BASIC+ features. While our models outperform BCS in both cases, we see that as we add more features (as in the BASIC+ set), the performance gap between our model and the linear system becomes even more pronounced.

We may also wonder whether the architecture represented by our scoring function, where the intermediate representations $\boldsymbol{h}_{\text{a}}$ and $\boldsymbol{h}_{\text{p}}$ are separated in the first layer, is necessary for these results. We accordingly compare with the fully connected versions of these two models (which are equivalent to 1 and 2 layer multi-layer perceptrons) using the BASIC+ features in Table 4.[9] There, we also evaluate the effect of pre-training on these models by comparing with the results of training from a random initialization. We see that while even randomly initialized models are capable of excellent performance, pre-training is beneficial, especially for $\boldsymbol{g}_1$.

[9]We also experimented with bilinear models both with and without non-linearities; these were also inferior.

6 Discussion

We attempt to gain insight into our model's errors using using two different error breakdowns. In Table 5 we show the errors as reported by the analysis tool of Kummerfeld and Klein (2013). In Table 6 we show a more fine-grained breakdown inspired by a similar analysis in Durrett and Klein (2013). In the latter table, we categorize the errors made by our system on the CoNLL 2012 development data in terms of (1) whether or not the mention has a head match with a previously occurring mention in the document, unless it is a pronominal mention, which we treat separately, (2) in terms of the status of the mention in the gold clustering, namely, singleton, first-in-cluster, or anaphoric, and (3) in terms of the type of error made (which, as discussed in Section 3, are one of FL, FN, and WL).

We note that the two models have slightly different error profiles, with $\boldsymbol{g}_1$ being slightly better at recall and $\boldsymbol{g}_2$ being slightly better at precision. Indeed, we see from Table 6 that the two models make a comparable number of total errors ($\boldsymbol{g}_1$ makes only 17 fewer errors overall). The increased precision of the $\boldsymbol{g}_2$ model is presumably due to the second layer around $\boldsymbol{h}_{\text{a}}$ and $\boldsymbol{h}_{\text{p}}$ in $\boldsymbol{g}_2$ allowing for antecedent evidence to interact with anaphoricity

Error Type	BCS	NN ($\boldsymbol{g}_1$)	NN ($\boldsymbol{g}_2$)
Conflated Entities	1603	1434	1371
Extra Mention	651	568	529
Extra Entity	655	623	561
Divided Entity	1989	1837	1835
Missing Mention	1004	997	1005
Missing Entity	1070	1026	1114

Table 5: Absolute error counts from the coreference analysis tool of Kummerfeld and Klein (2013). The upper set roughly corresponds to the precision and the lower to the recall of the coreference clusters produced by the model.

NN ($\boldsymbol{g}_1$)	Singleton		1st in clust.		Anaphoric	
	FL	#	FL	#	FN + WL	#
HM	817	8.2K	147	0.8K	700 + 318	4.7K
No HM	86	19.8K	41	2.4K	677 + 59	1.0K
Pron.	948	2.6K	257	0.5K	434 + 875	7.3K
NN ($\boldsymbol{g}_2$)	Singleton		1st in clust.		Anaphoric	
	FL	#	FL	#	FN + WL	#
HM	770	8.2K	130	0.8K	803 + 306	4.7K
No HM	73	19.8K	39	2.4K	699 + 52	1.0K
Pron.	896	2.6K	249	0.5K	456 + 869	7.3K

Table 6: Errors made by NN ($\boldsymbol{g}_1$) (top) and NN ($\boldsymbol{g}_2$) (bottom) on CoNLL 2012 English development data. Rows correspond to (1) mentions with a (previous) head match (HM), that is, mentions x such that $\mathcal{A}(x)$ contains another mention with the same head word, (2) with no previous head match (no HM), and (3) to pronominal mentions, respectively. The 3 column groups correspond to singleton, first-in-cluster, and anaphoric mentions (resp.), as determined by the gold clustering, with the number and type of errors on the left and the total number of mentions in the category (#) on the right.

evidence in a more complicated way. Ultimately, however, coreference systems operating over system mentions are already biased toward precision, and so the increased precision of $\boldsymbol{g}_2$ is not as helpful as the increased recall of $\boldsymbol{g}_1$ in the final CoNLL score.

In further analysis we found that many of the correct predictions made by the $\boldsymbol{g}_2$ model not made by $\boldsymbol{g}_1$ and the linear model involve predicting non-anaphoric even in the presence of highly misleading antecedent features like head-match. Figure 4 shows some examples of mentions with previous head matches that the linear system predicted as anaphoric and that our system correctly identifies as non-anaphoric.

We illustrate how the features in Figure 2 might be useful in such cases by considering the first example in Figure 4. There, a comma follows "the Nika TV company" in the text (and is picked up by the "word following" feature), perhaps indicating an appositive, which makes anaphoricity unlikely. The model can also learn that the

Non-Anaphoric (x)	Spurious Antecedent (y)
the Nika TV company	an independent company
Lexus sales	GM 's domestic car sales
The storage area	the harbor area
the Budapest location	Radio Free Europe 's new location
the synagogue	the synagogue too or something
the equity market	The junk market
their silver coin	one silver coin
the international school	The Hong Kong elementary school
the 1970s	the early 1970s
the 2003 season	the 2001 season

Figure 4: Example mentions x that were correctly marked non-anaphoric by $\boldsymbol{g}_2$, but incorrectly marked anaphoric with y as an antecedent by the BASIC+ linear model. These examples highlight the difficult case where there is a spurious head-match between non-coreferent pairs. See text for further details.

"company-company" head match is often misleading, and, in general, distance features may also rule out head matches. Note that while these features on their own may be more or less correlated with a mention being non-anaphoric, the model learns to combine them in a predictive way.

6.1 Further Improving Coreference Systems

Table 6 also gives a sense of where coreference systems such as ours need to improve. It is first important to note that the case of resolving an anaphoric mention that has no previous head matches (e.g., identifying that "the team" and "the New York Giants" are coreferent), which is often taken to be one of the major challenges facing coreference systems because it presumably requires semantic information, is not the largest source of errors. In fact, we see from Table 6 (second row, third column in both sub-tables) that while these cases do indeed account for a substantial percentage of errors, we make hundreds more errors predicting singleton pronominal mentions to be anaphoric (in the case of $\boldsymbol{g}_1$) and on incorrectly linking anaphoric pronominal mentions (in the case of $\boldsymbol{g}_2$). Further analysis indicates that these errors are almost entirely related to incorrectly linking pleonastic pronouns, such as "it" or "you," and that moreover the incorrectly predicted antecedent for these pleonastic pronouns is almost always (another instance of) the same pronoun.

That these pleonastic cases are so problematic is interesting when considered against the backdrop of the inference strategies typically employed by coreference systems, which we briefly mention here but discuss more fully in the next section. Currently, coreference systems divide be-

tween those using "local" models, which choose antecedents for potentially anaphoric mentions independently of each other, and "non-local" models, which make predictions that take into account predictions made for previous mentions, and perhaps even attempt to jointly predict all mentions in a document. While our model is entirely local, other recent high performing systems, such as that of Björkelund and Kuhn (2014), are not. One might suspect, then, that "non-local" inference might allow us to capture the fact that, for instance, a cluster of coreferent mentions should generally not consist solely of pronouns, and thereby avoid predicting (identical) pronominal antecedents for pleonastic pronouns.

As it turns out, however, almost 30% of the anaphoric pronominal mentions in the CoNLL development data participate in pronoun-only clusters (primarily in the context of broadcast or telephone conversations), which suggests that such a "non-local" rule may not be particularly useful, though further experiments are required. It is also worth noting that a suitably modified loss function may also be able to prevent excessive pronoun-pronoun linking, even in a local model.

7 Related Work

There is a voluminous literature on machine learning approaches to coreference resolution, effectively beginning with Soon et al. (2001). The recent introduction of the CoNLL datasets (Pradhan et al., 2012) has spurred research that takes advantage of more fine-grained features and richer models (Björkelund and Farkas, 2012; Chang et al., 2012; Durrett and Klein, 2013; Chang et al., 2013; Björkelund and Kuhn, 2014; Ma et al., 2014). Of these approaches, our model is related to the mention-ranking approaches (Bengtson and Roth, 2008; Denis and Baldridge, 2008; Rahman and Ng, 2009; Durrett and Klein, 2013; Chang et al., 2013), as opposed to those that focus on non-local, structured prediction (McCallum and Wellner, 2003; Culotta et al., 2006; Haghighi and Klein, 2010; Fernandes et al., 2012; Stoyanov and Eisner, 2012; Björkelund and Farkas, 2012; Wick et al., 2012; Björkelund and Kuhn, 2014; Durrett and Klein, 2014).

In motivation, our work is most similar to that of Ng (2004), who notes that anaphoricity information is useful within the broader coreference task, and who accordingly attempts to "globally" optimize performance based on this information, as well as that of Denis et al. (2007), who do joint decoding of anaphoricity and coreference predictions using ILP. Both of these works are taken to contrast with the more popular approach of doing an initial non-anaphoric pruning step (Ng and Cardie, 2002; Rahman and Ng, 2009; Recasens et al., 2013; Lee et al., 2013). In contrast, we jointly learn non-linear functions of anaphoricity and antecedent features, rather than tune a threshold, or jointly decode based on independently trained classifiers (as in Denis et al. (2007)). In a similar vein, several authors have also proposed using the output of an anaphoricity classifier as a feature in a downstream coreference system (Ng, 2004; Bengtson and Roth, 2008). In our framework we (re)learn features jointly with the full task, after a pre-training scheme that targets anaphoricity as well antecedent representations.

There has also been some work on automatically inducing feature conjunctions for use in coreference systems (Fernandes et al., 2012; Lassalle and Denis, 2013), though the approach we present here is somewhat simpler, and unlike that of Lassalle and Denis (2013) is designed for use on system rather than gold mentions.

There has been much interest recently in using neural networks for classic natural language tasks such as tagging and semantic role labeling Collobert et al. (2011), sentiment analysis (Socher et al., 2011; Socher et al., 2012), prepositional phrase attachment (Belinkov et al., 2014) among others. These systems often use some form of pre-training for initialization, often word-embeddings learned from external tasks. However, there has been little work of this form for coreference resolution.

8 Conclusion

We have presented a simple, local model capable of learning feature representations useful for coreference-related subtasks, and of thereby achieving state-of-the-art performance. Because our approach automatically learns intermediate representations given raw features, directions for further research might alternately explore including additional (perhaps semantic) raw features, as well as developing loss functions that further discourage learning representations that allow for common errors (such as those involving pleonastic pronouns).

References

Amit Bagga and Breck Baldwin. 1998. Algorithms for Scoring Coreference Chains. In *The first international conference on language resources and evaluation workshop on linguistics coreference*, volume 1, pages 563–566. Citeseer.

Yonatan Belinkov, Tao Lei, Regina Barzilay, and Amir Globerson. 2014. Exploring Compositional Architectures and Word Vector Representations for Prepositional Phrase Attachment. *Transactions of the Association for Computational Linguistics*, 2:561–572.

Eric Bengtson and Dan Roth. 2008. Understanding the Value of Features for Coreference Resolution. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 294–303. ACL.

Anders Björkelund and Richárd Farkas. 2012. Data-driven Multilingual Coreference Resolution using Resolver Stacking. In *Joint Conference on EMNLP and CoNLL-Shared Task*, pages 49–55. ACL.

Anders Björkelund and Jonas Kuhn. 2014. Learning structured perceptrons for coreference Resolution with Latent Antecedents and Non-local Features. *ACL, Baltimore, MD, USA, June*.

Kai-Wei Chang, Rajhans Samdani, Alla Rozovskaya, Mark Sammons, and Dan Roth. 2012. Illinois-coref: The UI System in the CoNLL-2012 Shared Task. In *Joint Conference on EMNLP and CoNLL-Shared Task*, pages 113–117. 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 2013 Conference on Empirical Methods in Natural Language Processing*, pages 601–612.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural Language Processing (almost) from Scratch. *The Journal of Machine Learning Research*, 12:2493–2537.

Aron Culotta, Michael Wick, Robert Hall, and Andrew McCallum. 2006. First-order Probabilistic Models for Coreference Resolution. *NAACL-HLT*.

Pascal Denis and Jason Baldridge. 2008. Specialized Models and Ranking for Coreference Resolution. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 660–669. ACL.

Pascal Denis, Jason Baldridge, et al. 2007. Joint Determination of Anaphoricity and Coreference Resolution using Integer Programming. In *HLT-NAACL*, pages 236–243. Citeseer.

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

Greg Durrett and Dan Klein. 2013. Easy Victories and Uphill Battles in Coreference Resolution. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1971–1982.

Greg Durrett and Dan Klein. 2014. A Joint Model for Entity Analysis: Coreference, Typing, and Linking. *Transactions of the Association for Computational Linguistics*, 2:477–490.

Dumitru Erhan, Yoshua Bengio, Aaron Courville, Pierre-Antoine Manzagol, Pascal Vincent, and Samy Bengio. 2010. Why does unsupervised pre-training help deep learning? *The Journal of Machine Learning Research*, 11:625–660.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. Liblinear: A Library for Large Linear Classification. *The Journal of Machine Learning Research*, 9:1871–1874.

Eraldo Rezende Fernandes, Cícero Nogueira Dos Santos, and Ruy Luiz Milidiú. 2012. Latent Structure Perceptron with Feature Induction for Unrestricted Coreference Resolution. In *Joint Conference on EMNLP and CoNLL-Shared Task*, pages 41–48. Association for Computational Linguistics.

Aria Haghighi and Dan Klein. 2010. Coreference Resolution in a Modular, Entity-centered Model. In *The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 385–393. Association for Computational Linguistics.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. Ontonotes: the 90% Solution. In *Proceedings of the human language technology conference of the NAACL, Companion Volume: Short Papers*, pages 57–60. Association for Computational Linguistics.

Philipp Koehn. 2004. Statistical Significance Tests for Machine Translation Evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395. Citeseer.

Jonathan K. Kummerfeld and Dan Klein. 2013. Error-driven Analysis of Challenges in Coreference Resolution. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, Seattle, WA, USA, October.

Emmanuel Lassalle and Pascal Denis. 2013. Improving Pairwise Coreference Models through Feature Space Hierarchy Learning. In *ACL 2013-Annual meeting of the Association for Computational Linguistics*.

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 Fifteenth Conference on Computational Natural Language Learning: Shared Task*, pages 28–34. Association for Computational Linguistics.

Heeyoung Lee, Angel Chang, Yves Peirsman, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. 2013. Deterministic Coreference Resolution based on Entity-centric, Precision-ranked Rules. *Computational Linguistics*, 39(4):885–916.

Xiaoqiang Luo, Sameer Pradhan, Marta Recasens, and Eduard Hovy. 2014. An Extension of BLANC to System Mentions. *Proceedings of ACL, Baltimore, Maryland, June*.

Xiaoqiang Luo. 2005. On Coreference Resolution Performance Metrics. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 25–32. Association for Computational Linguistics.

Chao Ma, Janardhan Rao Doppa, J Walker Orr, Prashanth Mannem, Xiaoli Fern, Tom Dietterich, and Prasad Tadepalli. 2014. Prune-and-score: Learning for Greedy Coreference Resolution. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*.

Andrew McCallum and Ben Wellner. 2003. Toward Conditional Models of Identity Uncertainty with Application to Proper Noun Coreference. *Advances in Neural Information Processing Systems 17*.

Yurii Nesterov. 1983. A Method of Solving a Convex Programming Problem with Convergence Rate O (1/k2). In *Soviet Mathematics Doklady*, volume 27, pages 372–376.

Vincent Ng and Claire Cardie. 2002. Identifying Anaphoric and Non-anaphoric Noun Phrases to Improve Coreference Resolution. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics.

Vincent Ng. 2004. Learning Noun Phrase Anaphoricity to Improve Coreference Resolution: Issues in Representation and Optimization. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 151. Association for Computational Linguistics.

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*, pages 1–40. ACL.

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*.

Altaf Rahman and Vincent Ng. 2009. Supervised Models for Coreference Resolution. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*, pages 968–977. ACL.

Marta Recasens, Marie-Catherine de Marneffe, and Christopher Potts. 2013. The Life and Death of Discourse Entities: Identifying Singleton Mentions. In *HLT-NAACL*, pages 627–633.

Richard Socher, Jeffrey Pennington, Eric H Huang, Andrew Y Ng, and Christopher D Manning. 2011. Semi-supervised Recursive Autoencoders for Predicting Sentiment Distributions. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 151–161. ACL.

Richard Socher, Brody Huval, Christopher D Manning, and Andrew Y Ng. 2012. Semantic Compositionality through Recursive Matrix-vector Spaces. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1201–1211. ACL.

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.

Veselin Stoyanov and Jason Eisner. 2012. Easy-first Coreference Resolution. In *COLING*, pages 2519–2534. Citeseer.

Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. 2013. On the Importance of Initialization and Momentum in Deep Learning. In *Proceedings of the 30th International Conference on Machine Learning*, pages 1139–1147.

Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. 1995. A Model-theoretic Coreference Scoring Scheme. In *Proceedings of the 6th conference on Message Understanding*, pages 45–52. ACL.

Michael Wick, Sameer Singh, and Andrew McCallum. 2012. A Discriminative Hierarchical Model for Fast Coreference at Large Scale. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 379–388. Association for Computational Linguistics.

Chun-Nam John Yu and Thorsten Joachims. 2009. Learning Structural SVMs with Latent Variables. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 1169–1176. ACM.