Learning Anaphoricity and Antecedent Ranking Features for Coreference Resolution

Sam Wiseman, Alexander M. Rush,
Stuart M. Shieber, and Jason Weston

HARVARD
School of Engineering
and Applied Sciences

Facebook AI Research

A Preliminary Example (CoNLL Dev Set, wsj/2404)

Cadillac posted a 3.2% increase despite new competition from Lexus, the fledgling luxury-car division of Toyota Motor Corp. Lexus sales weren't available; the cars are imported and Toyota reports their sales only at month-end.

With Coreferent Mentions Annotated

Cadillac posted a 3.2% increase despite new competition from [Lexus, the fledgling luxury-car division of [Toyota Motor Corp]] . [Lexus] sales weren't available; the cars are imported and [Toyota] reports [their] sales only at month-end.

Mention Ranking [??]

- Model each mention x as having a single "true" antecedent
- Score potential antecedents y of each mention x with a scoring function $s(x, y)$
  - Common to use $s_{\text{lin}}(x, y) \triangleq \boldsymbol{w}^{\mathsf{T}} \widetilde{\boldsymbol{\phi}}(x, y)$ as scoring function
- Predict $y^* = \arg\max_{y \in \mathcal{Y}(x)} s(x, y)$
- If only clusters annotated, "true" antecedent a latent variable when training [???]

$s(x, y_1) = 0.4$

$s(x, y_2) = 0.9$

. . . [Lexus] sales weren't available . . . [Toyota] reports [their]

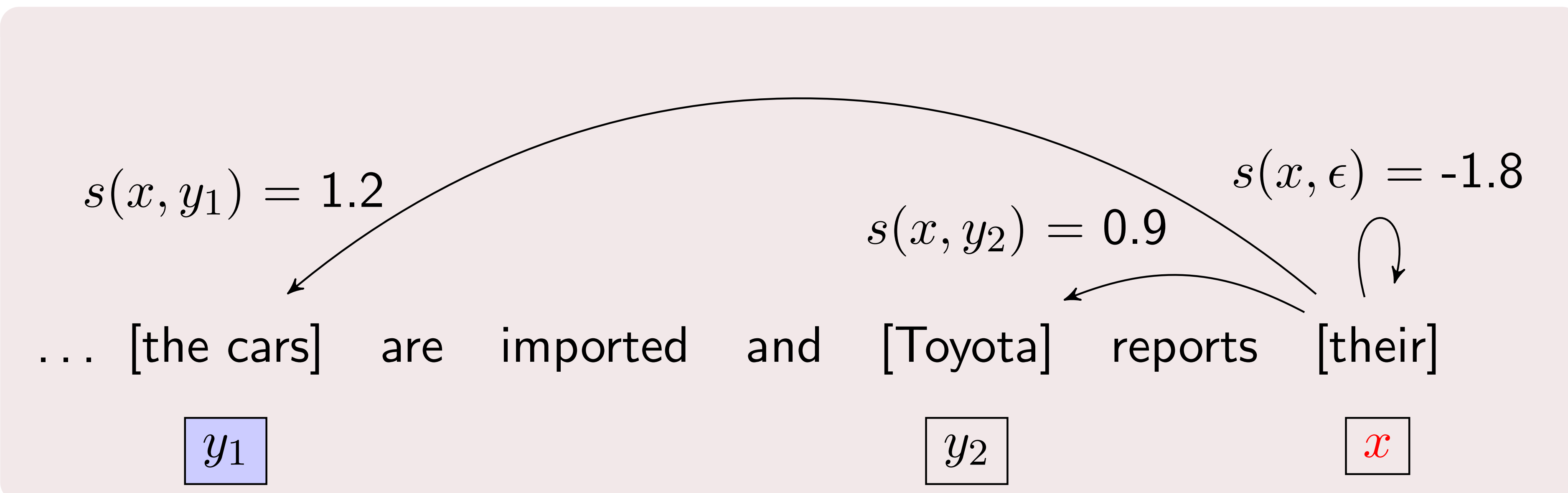
$\boxed{y_1}$ $\boxed{y_2}$ $\boxed{x}$

But Wait: Non-Anaphoric Mentions

[Cadillac] posted a [3.2% increase] despite [new competition from [Lexus, the fledgling luxury-car division of [Toyota Motor Corp]]] . [[Lexus] sales] weren't available; [the cars] are imported and [Toyota] reports [[their] sales] only at [month-end] .

Mention Ranking II

- Also score possibility that x non-anaphoric, denoted by $y = \epsilon$
- Can still use $s_{\text{lin}}(x, y) \triangleq \boldsymbol{w}^{\mathsf{T}} \widetilde{\boldsymbol{\phi}}(x, y)$ as scoring function
- Now $\mathcal{Y}(x) = \{\text{mentions before } x\} \cup \{\epsilon\}$
- Again predict $y^* = \arg\max_{y \in \mathcal{Y}(x)} s(x, y)$



Mention Ranking III

- Can duplicate features for a more flexible model:

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

- features on mention context (capture anaphoricity info)
- features on mention, antecedent pair (capture pairwise affinity)
- Above equivalent to model of **?**

Problems with Simple Features

[Cadillac] posted a [3.2% increase] despite [new competition from [Lexus, the fledgling luxury-car division of [Toyota Motor Corp]]] . [[Lexus] sales] weren't available; [the cars] are imported and [Toyota] reports [[their] sales] only at [month-end] .

Misleading Head Matches

[Lexus **sales**] and [their **sales**] not coreferent!

Problems with Simple Features

[Cadillac] posted a [3.2% increase] despite [new competition from [Lexus, the fledgling luxury-car division of [Toyota Motor Corp]]]. [[Lexus] sales] weren't available; [the cars] are imported and [Toyota] reports [[their] sales] only at [month-end].

Misleading Number Matches

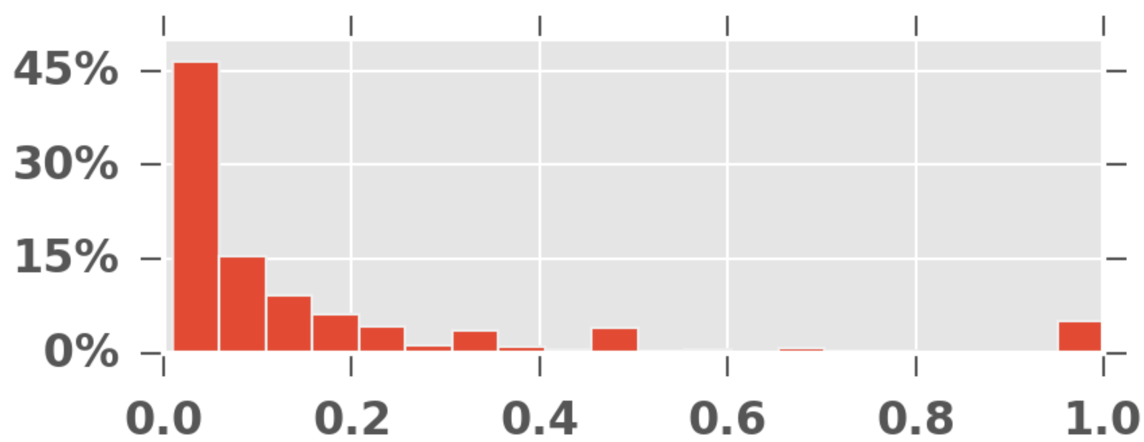
[the **cars**] and [**their**] not coreferent!

Simple Antecedent/Pairwise Features Not Discriminative

E.g., is [Lexus sales] the antecedent of [their sales]?

- Common antecedent features: String/Head Match, Sentences Between, Mention-Antecedent Numbers/Heads/Genders, etc.

$$\phi_{\mathrm{p}}(\text{[their sales]},\text{[Lexus sales]}) = \left\{ \begin{array}{c} \text{string-match=false} \\ \text{head-match=true} \\ \text{sentences-between=0} \\ \text{ment-ant-numbers=plur.,plur.} \\ \vdots \end{array} \right\}$$



Dealing with the Feature Problem

Finding discriminative features a major challenge for coreference systems [??]

- Typical to define (or search for) feature conjunction-schemes to improve predictive performance [**???**]. For instance:
  - string-match$(x, y) \wedge$ type$(x) \wedge$ type(y) [**?**], where

$$\text{type}(x) = \begin{cases} \text{Nom.} & \text{if } x \text{ is nominal} \\ \text{Prop.} & \text{if } x \text{ is proper} \\ \text{citation-form}(x) & \text{if } x \text{ is pronominal} \end{cases}$$

  - substring-match(head$(x), y) \wedge$ substring-match$(x$, head$(y)) \wedge$ coarse-type$(y) \wedge$ coarse-type(x) [**?**]
- Not just a problem for Mention Ranking systems!

Our Approach

Motivation: Current conjunction schemes perhaps not optimal, and in any case hard to scale as more features added.

Accordingly, we:

- Develop a model that learns good representations automatically
- Use only raw, unconjoined features
- Introduce pre-training scheme to improve quality of learned representations

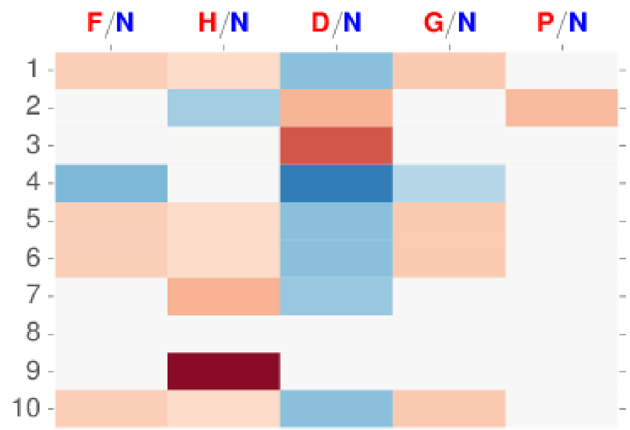
Extending the Piecewise Model I

Goal: learn higher order feature representations

We first define the following nonlinear feature representations:

$$\boldsymbol{h}_{\mathrm{a}}(x) \triangleq \tanh(\boldsymbol{W}_{\mathrm{a}}\, \boldsymbol{\phi}_{\mathrm{a}}(x) + \boldsymbol{b}_{\mathrm{a}})$$

$$\boldsymbol{h}_{\mathrm{p}}(x, y) \triangleq \tanh(\boldsymbol{W}_{\mathrm{p}}\, \boldsymbol{\phi}_{\mathrm{p}}(x, y) + \boldsymbol{b}_{\mathrm{p}})$$



- **Here, $\phi_{\mathrm{a}}, \phi_{\mathrm{p}}$ are raw, unconjoined features!**

Extending the Piecewise Model II

Use the scoring function

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

($\boldsymbol{g}_1$) If $\boldsymbol{g}$ is identity, obtain version of $s_{\mathrm{lin+}}$ with nonlinear features.

($\boldsymbol{g}_2$) If $\boldsymbol{g}$ is an additional hidden layer, further encourage nonlinear interactions between $\boldsymbol{h}_{\mathrm{a}}, \boldsymbol{h}_{\mathrm{p}}$

Training

To train, we use the following margin-based loss:

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

- Slack-rescale with a mistake-specific cost function $\Delta(x_n, \hat{y})$
- y_n^{ℓ} a latent antecedent: equal to highest scoring antecedent in same cluster (or ϵ) [**????**]
- Note that even if s were linear, would still be non-convex!

Pre-training Subtasks I

Two very natural subtasks for pre-training $\boldsymbol{h}_{\mathrm{a}}$ and $\boldsymbol{h}_{\mathrm{p}}$

Antecedent Ranking

Predict antecedents of known anaphoric mentions with scoring function

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

Anaphoricity Detection

Predict anaphoricity of mentions with scoring function

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

- We use similar, margin-based objectives for training

Pre-training Subtasks I

Two very natural subtasks for pre-training $\boldsymbol{h}_{\mathrm{a}}$ and $\boldsymbol{h}_{\mathrm{p}}$

Antecedent Ranking

Predict antecedents of known anaphoric mentions with scoring function

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

Anaphoricity Detection

Predict anaphoricity of mentions with scoring function

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

- We use similar, margin-based objectives for training

Pre-training Subtasks I

Two very natural subtasks for pre-training $\boldsymbol{h}_{\mathrm{a}}$ and $\boldsymbol{h}_{\mathrm{p}}$

Antecedent Ranking

Predict antecedents of known anaphoric mentions with scoring function

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

Anaphoricity Detection

Predict anaphoricity of mentions with scoring function

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

- We use similar, margin-based objectives for training

Pre-training Subtasks I

Two very natural subtasks for pre-training $\boldsymbol{h}_{\mathrm{a}}$ and $\boldsymbol{h}_{\mathrm{p}}$

Antecedent Ranking

Predict antecedents of known anaphoric mentions with scoring function

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

Anaphoricity Detection

Predict anaphoricity of mentions with scoring function

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

- We use similar, margin-based objectives for training

Pre-training Subtasks II

- Antecedent ranking of known anaphoric mentions very similar to "gold mention" version of coreference task (but slightly easier)
- Anaphoricity/Singleton detection has a long history in coreference resolution, generally as an initial step in a pipeline [**?????**]

Subtask Performance

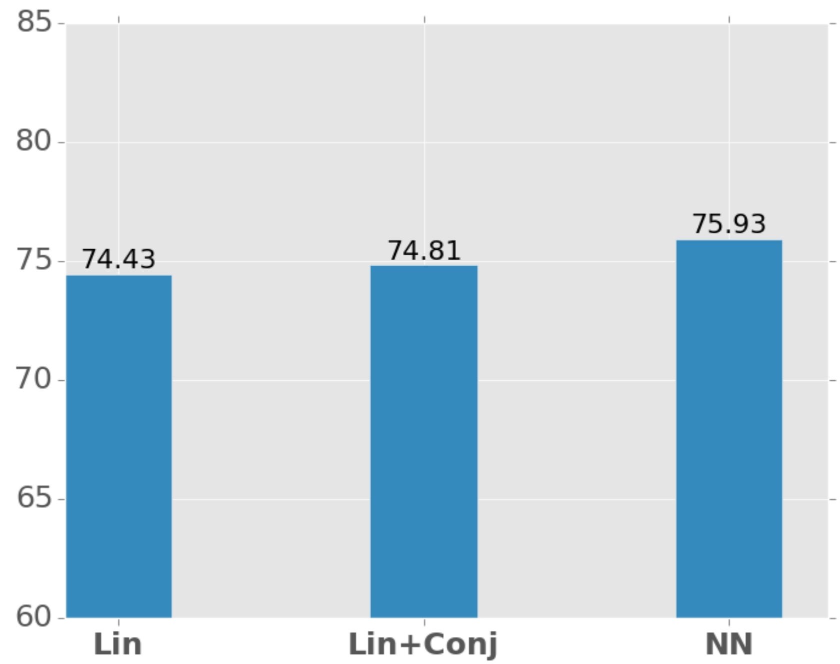


Figure: Anaphoricity Detection F_1 Score

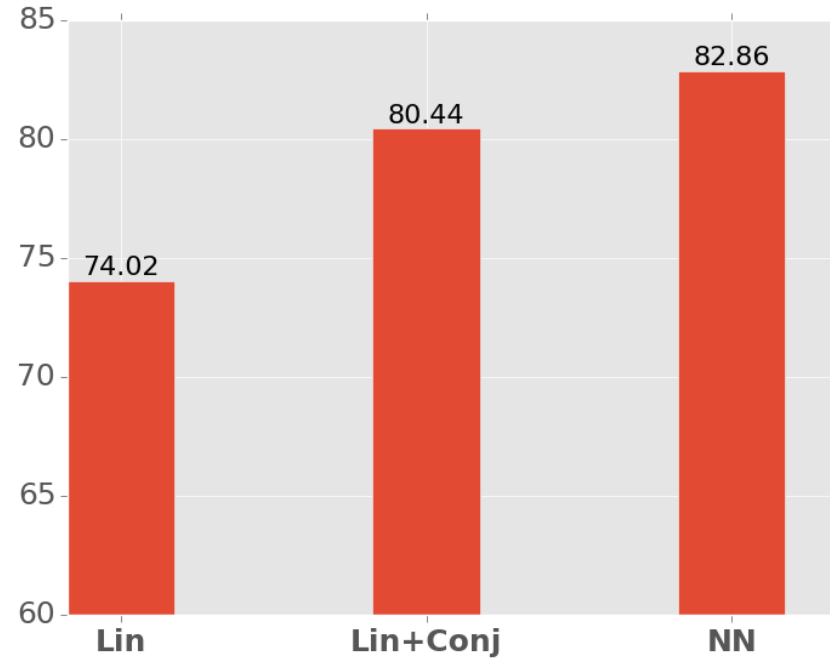


Figure: Antecedent Ranking Accuracy

- Subtask performance itself not crucial, but want to see that networks can learn good representations

Experimental Setup

- Used standard CoNLL 2012 English dataset experimental split
- Results scored with CoNLL 2012 scoring script v8.01
- Used Berkeley Coreference System [?] for mention extraction
- All optimization with Composite Mirror-Descent flavor of AdaGrad
- All hyperparameters (learning rates and regularization coefficients) tuned with grid-search on development set

Main Results

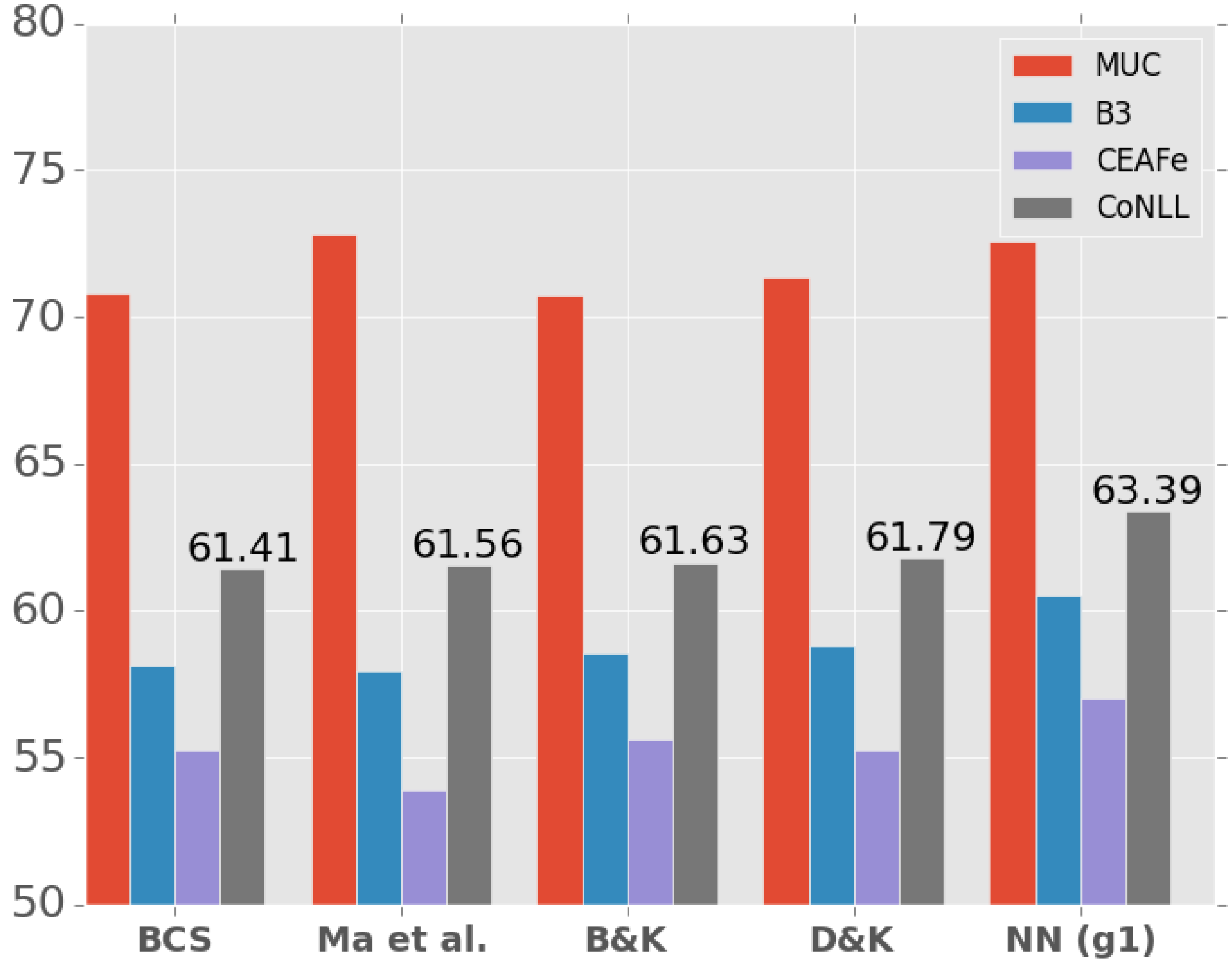


Figure: Results on CoNLL 2012 English test set. We compare with (in order) **?**, **?**, **?**, and **?**. F_1 gains are significant ($p < 0.05$) compared with both B&K and D&K for all metrics.

Main Results (Full Table)

	MUC			B^3			$CEAF_e$			
	P	R	F_1	P	R	F_1	P	R	F_1	CoNLL
BCS	74.89	67.17	70.82	64.26	53.09	58.14	58.12	52.67	55.27	61.41
Ma et al.	81.03	66.16	**72.84**	66.90	51.10	57.94	68.75	44.34	53.91	61.56
B&K	74.30	67.46	70.72	62.71	54.96	58.58	59.40	52.27	55.61	61.63
D&K	72.73	69.98	71.33	61.18	56.60	58.80	56.20	54.31	55.24	61.79
NN($\boldsymbol{g}_2$)	76.96	68.10	72.26	66.90	54.12	59.84	59.02	53.34	56.03	62.71
NN($\boldsymbol{g}_1$)	76.23	69.31	72.60	66.07	55.83	**60.52**	59.41	54.88	**57.05**	**63.39**

Table: Results on CoNLL 2012 English test set. We compare with (in order) **?**, **?**, **?**, and **?**. F_1 gains are significant ($p < 0.05$ under the bootstrap resample test **?**) compared with both B&K and D&K for all metrics.

Model Ablations

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

Table: F_1 performance on CoNLL 2012 development set

- Top sub-table examines whether separating $\boldsymbol{h}_{\mathrm{p}}, \boldsymbol{h}_{\mathrm{a}}$ (in first layer) actually helpful
- Bottom two sub-tables examine whether pre-training is helpful

Scaling to More Features

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: F_1 performance comparison between state-of-the-art linear mention-ranking model **?** and our full models on CoNLL 2012 development set for different feature sets.

Discussion: What are we getting wrong?

Mention Ranking models make error analysis very simple:

- Highest <u>percentage</u> error ($\frac{736}{1000}$) on anaphoric mentions with no previous occurring head-match
  - e.g., [the team] and [the New York Giants]
- Highest <u>number</u> of errors ($\frac{1309}{7300}$) on anaphoric pronouns
  - Almost all were errors on pleonastic pronouns ("it", "you"). About 2/3 involved incorrectly predicting another instance of same pronoun as antecedent.
  - An argument for more structure?
    - 30% of anaphoric pronominal mentions in CoNLL dev data are in pronoun-only clusters!

Summary

(1) Possible to achieve state-of-the-art performance with

- Very simple, local model and powerful scoring function
    - Note most recent state-of-the-art models non-local!
- Only raw, unconjoined features
- Over 1.5 pt increase over previous state-of-the-art in CoNLL score

(2) Separating anaphoricity and antecedent ranking (learned) representations beneficial

- Natural to pre-train on corresponding subtasks

Discussion: preliminaries

Note that Mention Ranking models make error analysis very simple!

Three Kinds of Errors Possible

(Adopting terminology of **?**):

(FL) **False Link** errors: predicting a mention to be anaphoric when it is non-anaphoric

(FN) **False New** errors: predicting a mention to be non-anaphoric when it is anaphoric

(WL) **Wrong Link** errors: predicting an incorrect antecedent for an anaphoric mention

Discussion: What are we getting wrong?

	Singleton		1^{st} in clust.		Anaphoric	
	FL	#	FL	#	FN + WL	#
Ment. w/ prev. head match	817	8.2K	147	0.8K	700 + 318	4.7K
Ment. w/o prev. head match	86	19.8K	41	2.4K	677 + 59	1.0K
Pronominal mentions	948	2.6K	257	0.5K	434 + 875	7.3K

Largest % error on anaphoric mentions with no previous head match

- The classic "hard" coreference case, presumably requiring knowledge, understanding

But make most errors (by far) on pronouns!

Pronoun Problems

Which pronominal mentions are we missing?

- FL and WL pronominal errors almost entirely on pleonastic pronominal mentions (e.g., "it", "you")
- Predicted antecedent almost always (another instance of) same pronoun

An argument for non-local inference?

- Note that 30% of anaphoric pronominal mentions in CoNLL development data in pronoun-only clusters

Thanks!

Thanks!

All Features

Mention Features (ϕ_{a})
Mention Head
Mention First Word
Mention Last Word
Word Preceding Mention
Word Following Mention
Words in Mention
Mention Synt. Ancestry
Mention Type
Mention Governor
Mention Sentence Index
Mention Entity Type
Mention Number
Mention Animacy
Mention Gender
Mention Person

Pairwise Features (ϕ_{p})
ϕ_{a}(Mention); ϕ_{a}(Antecedent)
Mentions between Ment., Ante.
Sentences between Ment., Ante.
i-within-i
Same Speaker
Document Type
Ante., Ment. String Match
Ante. contains Ment.
Ment. contains Ante.
Ante. contains Ment. Head
Mention contains Ante. Head
Ante., Ment. Head Match
Ante., Ment. Synt. Ancestries; Numbers; Genders; Persons; Entity Types; Heads; Types

Preliminary Embeddings Experiments

Can get ante up to 83.3462

on dev full task get: received MUC: 75.980000 69.490000 72.590000ESC received BCUB: 66.490000 58.030000 61.970000 received CEAFe: 61.120000 56.490000 58.710000 received CoNLL: 64.423333