Revealing the Myth of Higher-Order Inference in Coreference Resolution

Liyan Xu
Computer Science
Emory University, Atlanta, GA
liyan.xu@emory.edu

Jinho D. Choi
Computer Science
Emory University, Atlanta, GA
jinho.choi@emory.edu

Abstract

This paper analyzes the impact of higher-order inference (HOI) on the task of coreference resolution. HOI has been adapted by almost all recent coreference resolution models without taking much investigation on its true effectiveness over representation learning. To make a comprehensive analysis, we implement an end-to-end coreference system as well as four HOI approaches, attended antecedent, entity equalization, span clustering, and cluster merging, where the latter two are our original methods. We find that given a high-performing encoder such as SpanBERT, the impact of HOI is negative to marginal, providing a new perspective of HOI to this task. Our best model using cluster merging shows the Avg-F1 of 80.2 on the CoNLL 2012 shared task dataset in English.

1 Introduction

Coreference resolution has always been considered one of the unsolved NLP tasks due to its challenging aspect of document-level understanding (Wiseman et al., 2015, 2016; Clark and Manning, 2015, 2016; Lee et al., 2017). Nonetheless, it has made a tremendous progress in recent years by adapting contextualized embedding encoders such as ELMo (Lee et al., 2018; Fei et al., 2019) and BERT (Kantor and Globerson, 2019; Joshi et al., 2019, 2020). The latest state-of-the-art model shows the improvement of 12.4% over the model introduced 2.5 years ago, where the major portion of the improvement is derived by representation learning (Figure 1).

Most of these previous models have also adapted higher-order inference (HOI) for the global optimization of coreference links, although HOI clearly has not been the focus of those works, for the fact that gains from HOI have been reported marginal. This has inspired us to analyze the impact of HOI on modern coreference resolution models in order to envision the future direction of this research.

To make thorough ablation studies among different approaches, we implement an end-to-end coreference system in PyTorch (Sec 3.1), and two HOI approaches proposed by previous work, attended antecedent and entity equalization (Sec 3.2), along with two of our original approaches, span clustering and cluster merging (Sec 3.3). These approaches are experimented with two Transformer encoders, BERT and SpanBERT, to assess how effective HOI is even when coupled with those high-performing encoders (Sec 4). To the best of our knowledge, this is the first work to make a comprehensive analysis on multiple HOI approaches side-by-side for the task of coreference resolution.[1]

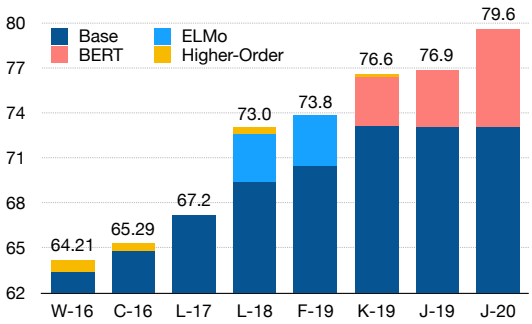

Figure 1: Performance of the recent state-of-the-art models on the CoNLL 2012 shared task. W-16: Wiseman et al. (2016), C-16: Clark and Manning (2016), L-17: Lee et al. (2017), L-18: Lee et al. (2018), F-19: Fei et al. (2019), K-19: Kantor and Globerson (2019), J-19: Joshi et al. (2019), J-20: Joshi et al. (2020).

2 Related Work

Most neural network-based coreference resolution models have adapted antecedent-ranking (Wiseman et al., 2015; Clark and Manning, 2015; Lee et al., 2017, 2018; Joshi et al., 2019, 2020), which relies on the local decisions between each mention and its

[1] Source codes and models are available at https://github.com/emorynlp/coref-hoi.

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 8527–8533,
November 16–20, 2020.

antecedents. To achieve deeper global optimization, Wiseman et al. (2016); Clark and Manning (2016); Yu et al. (2020) built entity representations in the ranking process, whereas Lee et al. (2018); Kantor and Globerson (2019) refined the mention representation by aggregating its antecedents' information.

There is no secret that the integration of contextualized embeddings has played the most critical role in this task. While the followings are based on the same end-to-end coreference model (Lee et al., 2017), Lee et al. (2018); Fei et al. (2019), Peters et al. (2018) reported 3.3% improvement by adapting ELMo in the encoders. Kantor and Globerson (2019); Joshi et al. (2019) gained additional 3.3% by adapting BERT (Devlin et al., 2019). Joshi et al. (2020) introduced SpanBERT that gave another 2.7% improvement over Joshi et al. (2019). Most recently, Wu et al. (2020) proposes a new model that adapts question-answering framework on coreference resolution, and achieves state-of-the-art result of 83.1 on the CoNLL'12 shared task.

3 Approach

3.1 End-to-End Coreference System

We reimplement the end-to-end *c2f-coref* model introduced by Lee et al. (2018) that has been adapted by every coreference resolution model since then. It detects mention candidates through span enumeration and aggressive pruning. For each candidate span x, the model learns the distribution over its antecedents $y \in \mathcal{Y}(x)$:

$$P(y) = \frac{e^{s(x,y)}}{\sum_{y' \in \mathcal{Y}(x)} e^{s(x,y')}} \tag{1}$$

where $s(x, y)$ is the local score involving two parts: how likely the spans x and y are valid mentions, and how likely they refer to the same entity:

$$\begin{aligned} s(x,y) &= s_m(x) + s_m(y) + s_c(x,y) \\ s_m(x) &= w_m \texttt{FFNN}_m(g_x) \\ s_c(x,y) &= w_c \texttt{FFNN}_c(g_x, g_y, \phi(x,y)) \end{aligned} \tag{2}$$

g_x, g_y are the span embeddings of x and y, $\phi(x, y)$ is the meta-information (e.g., speakers, distance), and w_m, w_c are the mention and coreference scores, respectively (FFNN: feedforward neural network). We use different Transformers-based encoders, and follow the "*independent*" setup for long documents as suggested by Joshi et al. (2019).

3.2 Span Refinement

Two HOI methods presented by recent coreference work are based on span refinement that aggregates non-local features to enrich the span representation with more "global" information. The updated span representation g'_x can be derived as in Eq. 3, where g'_x is the interpolation between the current and refined representation g_x and a_x, and W_f is the gate parameter. g'_x is used to perform another round of antecedent-ranking in replacement of g_x.

$$\begin{aligned} g'_x &= f_x \circ g_x + (1 - f_x) \circ a_x \\ f_x &= \sigma(W_f [g_x, a_x]) \end{aligned} \tag{3}$$

The following two methods share the same updating process for g'_x, but with different ways to obtain the refined span representation a_x.

Attended Antecedent (AA) takes the antecedent information to enrich g'_x (Lee et al., 2018; Fei et al., 2019; Joshi et al., 2019, 2020). The refined span a_x is the attended antecedent representation over the current antecedent distribution $P(y)$, where $g_{y \in \mathcal{Y}(x)}$ is the antecedent representation:

$$a_x = \sum_{y \in \mathcal{Y}(x)} P(y) \cdot g_y \tag{4}$$

Entity Equalization (EE) takes the clustering relaxation as in Eq. 5 to model the entity distribution (Kantor and Globerson, 2019), where $Q(x \in E_{y'})$ is the probability of the span x referring to an entity $E_{y'}$ in which the span y' is the first mention. $P(y)$ is the current antecedent distribution.

$$Q(x \in E_{y'}) = \begin{cases} \sum_{k=y'}^{x-1} P(y = k) \cdot Q(k \in E_{y'}) & y' < x \\ P(y = \epsilon) & y' = x \\ 0 & y' > x \end{cases} \tag{5}$$

The refined span a_x is the attended entity representation, where $e_y^{(x)}$ is the entity representation to which the span y belongs till the span x:

$$e_x^{(t)} = \sum_{y=1}^{t} Q(y \in E_x) \cdot g_y \tag{6}$$

$$a_x = \sum_{y=1}^{x} Q(x \in E_y) \cdot e_y^{(x)} \tag{7}$$

3.3 HOI with Clustering

This section introduces two new HOI methods for a more extensive study in HOI.

Span Clustering (SC) is also based on span refinement, and it constructs the actual clusters and obtains the "true" predicted entities using $P(y)$ instead of modeling the "soft" entity clusters through the relaxation as in EE (Section 3.2). This way, although we lose the differentiable property, the obtaining of true entities with the same empirical inference time as EE has made SC desirable.
The entity representation e_i for an entity cluster C_i is given by the attended spans in this cluster:

$$\alpha_t = w_\alpha \text{FFNN}_\alpha(g_t)$$
$$\alpha_{i,t} = \frac{\exp(\alpha_t)}{\sum_{k \in C_i} \exp(\alpha_k)}$$
$$e_i = \sum_{t \in C_i} \alpha_{i,t} \cdot g_t$$

The entity clusters C_i are constructed in the same way as in the final cluster prediction. The refined span a_x is then equal to the representation of entity e_i to which it belongs ($g_x \in C_i$).

Cluster Merging (CM) performs sequential antecedent ranking combining both antecedent and entity information to gradually build up the entity clusters, which is distinguished from span refinement methods that simply re-rank antecedents. Algorithm 1 describes the ranking process for CM. g_i is the i'th span, $\mathcal{Y}(i)$ is the indices of g_i's antecedents, and C_i is the cluster that g_i belongs to. The ranking score $s_x(y)$ consists of both antecedent score f_a (see Eq. 2) and cluster score f_c. To avoid overlapping between f_a and f_c, we set f_c as 0 if the cluster is the initial cluster (L6). Thus, f_c becomes the consultation such that when $f_c > 0$, the span g_x is likely to match the cluster C_y, and vice versa. f_c is computed by FFNN similar to f_a, and $\phi(C_y)$ is the meta-feature such as the cluster size.

Algorithm 1 Antecedent Ranking for CM

```
1:  procedure RANKING(g_1, ..., g_N)
2:      C_{i=1,...,N} ← g_i
3:      R ← ranking_order(g_1, ..., g_N)
4:      for x = R_1 ... R_N do
5:          for y ∈ Y(x) do                          ▷ Parallelized
6:              f_c(g_x, C_y) ← 0 if C_y = g_y
7:              s_x(y) ← f_a(g_x, g_y) + f_c(g_x, C_y, φ(C_y))
8:          y' ← argmax_{y∈Y(x)} s_x(y)
9:          if y' ≠ ε then
10:             merge C_x and C_{y'}
11:     return s_1, ..., s_N
```

Two simple configurations can be tuned for CM. We can have the sequential left-to-right ranking order or the easy-first order (L3) whose sequence is ordered by each span's max antecedent score, building the most confident clusters first (Ng and Cardie, 2002; Clark and Manning, 2016). There can be element-wise mean or max-reduction among the spans in the two merging clusters (L10).

Distinguished from Wiseman et al. (2016), clusters in CM are searched and merged in training without the use of oracle clusters, closing the gap between training and test time.

4 Experiments

For our experiments, the CoNLL 2012 English shared task dataset is used (Pradhan et al., 2012). Given the end-to-end coreference system in Section 3.1, six models are developed as follows:[2]

- BERT: BERT (Devlin et al., 2019) as the encoder
- SpanBERT: SpanBERT (Joshi et al., 2020) as the encoder
- +AA: SpanBERT with attended antecedent (§3.2)
- +EE: SpanBERT with entity equalization (§3.2)
- +SC: SpanBERT with span clustering (§3.3)
- +CM: SpanBERT with cluster merging (§3.3)

Note that BERT and SpanBERT completely rely on only local decisions without any HOI. Particularly, +AA is equivalent to Joshi et al. (2020).

4.1 Results

Table 1 shows the best results in comparison to previous state-of-the-art systems. We also report the mean scores and standard deviations from 5 repeated developments, which we could not find from the previous works.

The impact of SpanBERT over BERT is clear, showing 2.4% improvement on average. However, none of the HOI models shows a clear advantage over SpanBERT which adapts no HOI. In fact, all HOI models except for CM show negative impact. The best result is achieved by CM with the Avg-F1 of 80.2, surpassing the previous best result of 79.6 based on *c2f-coref* reported by Joshi et al. (2020).

4.2 Impact Analysis of HOI

Three HOI methods based on span refinement, AA, EE, and SC, show negative impact upon local decisions. We suspect that error propagation from antecedent-ranking may downgrade the quality of refinement. On the other hand, CM shows marginal improvement, suggesting that maintaining entity clusters can be superior to span refinement, at the

[2] Appdendix A.1 provides details of our experimental settings.

	MUC			B^3			$CEAF_{\phi_4}$				
	P	R	F1	P	R	F1	P	R	F1	Avg. F1	Avg-M
L-17	78.4	73.4	75.8	68.6	61.8	65.0	62.7	59.0	60.8	67.2	-
L-18	81.4	79.5	80.4	72.2	69.5	70.8	68.2	67.1	67.6	73.0	-
F-19	85.4	77.9	81.4	77.9	66.4	71.7	70.6	66.3	68.4	73.8	-
K-19	82.6	84.1	83.4	73.3	76.2	74.7	72.4	71.1	71.8	76.6	-
J-19	84.7	82.4	83.5	76.5	74.0	75.3	74.1	69.8	71.9	76.9	-
J-20	85.8	84.8	85.3	78.3	77.9	78.1	76.4	74.2	75.3	79.6	-
BERT	85.0	82.5	83.8	77.3	74.0	75.6	74.9	70.7	72.8	77.4	77.3 (±0.1)
SpanBERT	85.7	85.3	85.5	78.6	78.6	78.6	**76.8**	74.8	75.8	79.9	79.7 (±0.1)
+ AA	**86.1**	84.8	85.4	**79.3**	77.3	78.3	76.0	74.7	75.4	79.7	79.4 (±0.2)
+ EE	85.7	84.5	85.1	78.5	77.4	77.9	76.7	73.4	75.0	79.4	78.9 (±0.4)
+ SC	85.5	85.2	85.4	78.4	78.5	78.4	76.5	74.1	75.2	79.7	79.2 (±0.3)
+ CM	85.9	**85.5**	**85.7**	79.0	**78.9**	**79.0**	76.7	**75.2**	**75.9**	**80.2**	**79.9** (±0.2)

Table 1: Best results on the test set of the CoNLL'12 English shared task. The averaged F1 of MUC, B^3, $CEAF_{\phi_4}$ is the main evaluation metric. Avg-M: the mean Avg-F1 and its standard deviation from five developments. The mean and stdev of other metrics are provided in Appendix A.2. See Figure 1 for acronyms of the previous works.

cost of more inference time from the sequential ranking process. To analyze the direct impact of HOI, we take the trained models of each HOI method and evaluate them on the test set while turning off HOI, making it compatible to SpanBERT.

The averaged performance drop with respect to Avg-F1 after turning off HOI is less than 0.2 for all methods (Appendix A.3), implying that none of the HOI method has a significantly direct impact to the final performance of the model using SpanBERT. In further investigation, we examine the change of coreferent links with respect to correctness. Specifically, Table 2 shows the four types of link changes before and after HOI. It demonstrates that the benefits from HOI is diminished because the effects are two-sided: there are roughly same amounts of links (about 1%) becoming correct or wrong after HOI, therefore neither HOI method leads to much improvement overall.

	W2C	C2W	C2C	W2W
+ AA	240.8 (1.3)	241.2 (1.3)	16262.2	2168.4
+ EE	244.1 (1.3)	245.3 (1.3)	16183.3	2136.3
+ SC	248.2 (1.3)	262.0 (1.4)	16184.4	2146.0
+ CM	226.4 (**1.2**)	235.0 (**1.2**)	16446.0	2180.0

Table 2: Averaged statistics on the test set prediction of four HOI approaches. W2C represents the number of mentions that are linked to a **W**rong antecedent before HOI and are linked to a **C**orrect antecedent after HOI; vice versa for C2W. C2C/W2W is the number of mentions that are both linked to **C**orrect/**W**rong antecedents before and after HOI. Parentheses indicate the percentage of corresponding numbers per row.

It is worth mentioning that the impact of HOI is not limited to only global decisions. HOI implicitly serves as a way of regularization that impacts local decisions as well, since HOI and local ranking are mutually dependent during training. Such indirect influence of HOI makes it difficult to assess its true impact, which we will explore more in the future.

4.3 Analysis of Pronoun Resolution

Direct Inference For the error analysis, we examine the direct inference between two personal pronouns.[3] SP/PS in Table 3 shows the numbers of links that one pronoun incorrectly selects another pronoun with different plurality as its antecedent. We find that adapting HOI shows slightly higher impact than switching to a more advanced encoder. AA can reinforce the pronoun representation to bias towards singularity and lead to lower SP error and higher PS error, while the difference between BERT and SpanBERT is trivial on SP/PS.

We also look at the general types of coreferent errors involving two pronouns. False Link (FL) falsely links a non-anaphoric pronoun to another pronoun as antecedent; Wrong Link (WL) links an anaphoric pronoun to another wrong pronoun as antecedent. Table 3 shows that EE and CM reduce FL errors by 4+%, suggesting that the aggregation of non-local features indeed leads to more conservative linking decisions. However, adapting an advanced encoder shows higher impact on WL errors, as SpanBERT reduces almost 10% compared to BERT, implying that representation learning is still more important for semantic matching.

[3] Ambiguous pronouns such as "you" are excluded in direct inference analysis, and included in indirect inference analysis.

Indirect Inference The plurality of ambiguous pronouns such as *you* depends on the context. Two indirect links of (*he*, *you*) and (*you*, *they*) can be common to induce incorrect clusters that contain both singular and plural pronouns (Wiseman et al., 2016; Lee et al., 2018). Table 3 shows the numbers of these erroneous clusters in prediction. Surprisingly, very few of these clusters contain ambiguous pronouns in either approach. This observation moderates the long-standing movitation of HOI.

Additionally, the change of representation from BERT to SpanBERT has far more impact that reduces 10% of these erroneous clusters, while the four HOI methods fail to show significant difference compared to `SpanBERT`.

	SP	PS	FL	WL	BC
`BERT`	2.3	6.5	213.8	186.3	48.8 (3.5)
`SpanBERT`	2.8	6.6	218.3	168.0	**43.8** (2.7)
+ `AA`	**1.8**	8.8	214.2	**159.4**	44.8 (2.4)
+ `EE`	**1.8**	**5.5**	210.0	165.3	44.0 (2.5)
+ `SC`	3.8	7.2	223.6	170.0	45.4 (3.0)
+ `CM`	3.0	6.6	**208.0**	162.2	**43.8** (2.6)

Table 3: Averaged statistics on the test set prediction of different approaches. SP is the number of coreferent links from **S**ingular to **P**lural personal pronouns; vice versa for PS. FL (False Link) and WL (Wrong Link) is the number of conreferent link errors that involve two personal pronouns. BC is the number of clusters that contain both singular and plural pronouns, and the parentheses indicate the numbers of BC that contain ambiguous pronouns such as "you".

5 Conclusion

We implement the end-to-end coreference resolution model and investigate four higher-order inference methods, including two of our own methods. Our best model shows the new result of 80.2 on the CoNLL 2012 dataset. We thoroughly analyze the empirical effectiveness of HOI and demonstrate why it fails to boost performance on the CoNLL 2012 dataset compared to the improvement from encoders. We show that current HOI does not meet up with the original motivation, suggesting that a new perspective of HOI is needed for this task in the era of deep learning-based NLP.

Acknowledgments

We gratefully acknowledge the support of the AWS Machine Learning Research Awards (MLRA). Any contents in this material are those of the authors and do not necessarily reflect the views of AWS.

References

Kevin Clark and Christopher D. Manning. 2015. Entity-centric coreference resolution with model stacking. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1405–1415, Beijing, China. Association for Computational Linguistics.

Kevin Clark and Christopher D. Manning. 2016. Improving coreference resolution by learning entity-level distributed representations. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 643–653, Berlin, Germany. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Hongliang Fei, Xu Li, Dingcheng Li, and Ping Li. 2019. End-to-end deep reinforcement learning based coreference resolution. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 660–665, Florence, Italy. Association for Computational Linguistics.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Mandar Joshi, Omer Levy, Luke Zettlemoyer, and Daniel Weld. 2019. BERT for coreference resolution: Baselines and analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5803–5808, Hong Kong, China. Association for Computational Linguistics.

Ben Kantor and Amir Globerson. 2019. Coreference resolution with entity equalization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 673–677, Florence, Italy. Association for Computational Linguistics.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197, Copenhagen, Denmark. Association for Computational Linguistics.

Kenton Lee, Luheng He, and Luke Zettlemoyer. 2018. Higher-order coreference resolution with coarse-to-fine inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 687–692, New Orleans, Louisiana. Association for Computational Linguistics.

Vincent Ng and Claire Cardie. 2002. Improving machine learning approaches to coreference resolution. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 104–111, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. 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, Jeju Island, Korea. Association for Computational Linguistics.

Sam Wiseman, Alexander M. Rush, Stuart Shieber, and Jason Weston. 2015. Learning anaphoricity and antecedent ranking features for coreference resolution. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1416–1426, Beijing, China. Association for Computational Linguistics.

Sam Wiseman, Alexander M. Rush, and Stuart M. Shieber. 2016. Learning global features for coreference resolution. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 994–1004, San Diego, California. Association for Computational Linguistics.

Wei Wu, Fei Wang, Arianna Yuan, Fei Wu, and Jiwei Li. 2020. CorefQA: Coreference resolution as query-based span prediction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6953–6963, Online. Association for Computational Linguistics.

Juntao Yu, Alexandra Uma, and Massimo Poesio. 2020. A cluster ranking model for full anaphora resolution. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 11–20, Marseille, France. European Language Resources Association.

A Appendices

A.1 Experimental Settings

We implement the experimented models using PyTorch. $\text{BERT}_{\text{Large}}$ and $\text{SpanBERT}_{\text{Large}}$ are used as encoders. For each experiment, the best performed model on the development set is selected and evaluated on the test set.

Hyperparameters and Implementation Similar to Joshi et al. (2019, 2020), documents are split into independent segments with maximum 384 word pieces for $\text{BERT}_{\text{Large}}$ and 512 for $\text{SpanBERT}_{\text{Large}}$. In our final setting, BERT-parameters and task-parameters have separate learning rates (1×10^{-5} and 3×10^{-4} respectively), separate linear decay schedule, and separate weight decay rates (10^{-2} and 0 respectively). Models are trained 24 epochs with dropout rate 0.3.

The implementation of EE is based on the Tensorflow implementation from Kantor and Globerson (2019) which requires $\mathcal{O}(k^2)$ memory with k being the number of extracted spans, while other HOI approaches only requires $\mathcal{O}(k)$ memory [4]. To keep the GPU memory usage within 32GB, we limit the maximum number of span candidates for EE to be 300, which may have a negative impact on the performance.

Experiments are conducted on Nvidia Tesla V100 GPUs with 32GB memory. The average training time is around 7 hours for BERT and SpanBERT without HOI, and ranges from 9 - 15 hours with HOI methods.

A.2 Results

Table 4 reports the macro-average F1 scores out of 5 repeated developments of each approach. CM still has the best performance with 79.9 averaged F1 score. Span refinement-based HOI approaches, AA, EE, and SC, still have lower F1 scores than the local-only SpanBERT.

We do not find different configurations for CM make any huge impact to the performance. The final configuration for CM is sequential order and max reduction (Algorithm 1).

A.3 Analysis

AA	-0.02 ($\pm$ 0.06)
EE	0.03 ($\pm$ 0.07)
SC	0.11 ($\pm$ 0.10)
CM	0.04 ($\pm$ 0.04)

Table 5: Performance drop on CoNLL'12 English test set after turning off the corresponding HOI in trained models.

Table 5 shows the averaged performance drop and its standard deviations w.r.t Avg-F1 after turning off the corresponding HOI in trained models, to see the direct performance impact of HOI over local decisions.

Pronoun Resolution In our analysis, the following personal pronouns are regarded as ambiguous pronouns: "you", "your", "yours".

[4] The maximum number of antecedents for all models is set to 50 which is constant.

	MUC	B^3	CEAF_{ϕ_4}	
	F1	F1	F1	Avg. F1
BERT	83.7 ($\pm$ 0.1)	75.5 ($\pm$ 0.1)	72.6 ($\pm$ 0.1)	77.3 ($\pm$ 0.1)
SpanBERT	85.3 ($\pm$ 0.1)	78.4 ($\pm$ 0.1)	75.5 ($\pm$ 0.3)	79.7 ($\pm$ 0.1)
+ AA	85.2 ($\pm$ 0.2)	78.1 ($\pm$ 0.2)	75.0 ($\pm$ 0.2)	79.4 ($\pm$ 0.2)
+ EE	85.0 ($\pm$ 0.1)	77.7 ($\pm$ 0.2)	74.7 ($\pm$ 0.2)	78.9 ($\pm$ 0.4)
+ SC	85.1 ($\pm$ 0.2)	77.9 ($\pm$ 0.3)	74.7 ($\pm$ 0.3)	79.2 ($\pm$ 0.3)
+ CM	**85.5** ($\pm$ 0.2)	**78.5** ($\pm$ 0.3)	**75.6** ($\pm$ 0.2)	**79.9** ($\pm$ 0.2)

Table 4: Results on the test set of the CoNLL'12 English shared task data. Macro-average is reported for each F1 score from 5 repeated developments of each approach. See Section 4 for the approaches.