Pre-training Mention Representations in Coreference Models

Yuval Varkel
Tel Aviv University

Amir Globerson
Tel Aviv University and Google Research

Abstract

Collecting labeled data for coreference resolution is a challenging task, requiring skilled annotators. It is thus desirable to develop coreference resolution models that can make use of unlabeled data. Here we provide such an approach for the powerful class of neural coreference models. These models rely on representations of mentions, and we show these representations can be learned in a self-supervised manner towards improving resolution accuracy. We propose two self-supervised tasks that are closely related to coreference resolution and thus improve mention representation. Applying this approach to the GAP dataset results in new state of the arts results.

1 Introduction

Coreference resolution models cluster mentions by their referring entities. Almost all such models rely on vector representations of mentions (Clark and Manning, 2016; Lee et al., 2017, 2018; Denis and Baldridge, 2008; Rahman and Ng, 2009; Durrett et al., 2013; Chang et al., 2013; Wiseman et al., 2016; Martschat and Strube, 2015). The representations for all mentions are then compared (usually sequentially) and mention pairs judged to be most similar are considered coreferent.

Thus, the mention representation is a key component in modern coreference resolution models. Indeed, it has recently been shown that improving this representation leads to improved resolution performance. For example, BERT embeddings were used in (Joshi et al., 2019b) and SpanBERT (Joshi et al., 2019a) further improved performance.

However, both BERT and SpanBERT representations are trained on self-supervised tasks that seem quite distant from coreference resolution (e.g., masked-word-prediction in BERT, and masked-whole-span-prediction in SpanBERT). This suggests the possibility that unlabeled data can be used for further improving coreference resolution if we use self-supervised tasks that are more closely related to coreference resolution.

Motivated by the above, we ask: which self-supervised tasks should be used to improve mention representation for coreference resolution.

Two recent attempts for pre-training coreference models have focused on tasks such as language modeling (Liu et al., 2019) and masked-word-prediction for name resolution (Kocijan et al., 2019). Here we propose self-supervision tasks that train the coreference model directly (rather than just the underlying BERT), resulting in improved mention representations and resolution accuracy.

We identify two signals in a text that are highly informative for coreference resolution and show how to use them for self-supervision. The first signal is that the same name can appear multiple times in a text, and these mentions very likely corefer. Thus we can train mention representations to be similar for these mentions. The second signal is pronouns. Since each pronoun is likely to refer to *some* mention, we optimize mention representations to maximize the accuracy of this prediction.

We describe a training procedure for both these losses and show that together they result in new state of the art results on the GAP coreference dataset. Importantly this is a fairly small dataset, and thus our results demonstrate the power of unsupervised pre-training of mention representations.

2 Baseline Model

The coreference resolution task corresponds to extracting the set of mentions from a text and clustering these, such that clusters correspond to all mentions of a specific entity.

As a baseline model, we use the work of Joshi et al. (2019b), which builds on top of Lee et al. (2018) and uses SpanBERT (Joshi et al., 2019a). SpanBERT has the same architecture as BERT (Devlin et al., 2018) but is trained with a different objective, where whole spans are masked and span

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 8534–8540,
November 16–20, 2020. ©2020 Association for Computational Linguistics

boundary representations are optimized to predict all tokens of the masked span. This feature proves useful for coreference since, in many cases, entity mentions are spans of tokens, and span ranking models benefit from improved span representations.

The input to the coreference model is a span representation $\boldsymbol{r}_i$ for each mention i. In what follows, dependence on $\boldsymbol{r}_i$ is implied from dependence on i. A scoring function $s_m(i)$ is defined to score whether a span i is a mention. Only a portion of the top-scored mentions are kept for antecedent matching. An antecedent scoring function $s(i, j)$ is defined to score whether j is an antecedent of i. Using the pairwise function, for each span i, a distribution $P(y_i)$ over antecedents is defined:

$$P(y_i) = \frac{e^{s(i,y_i)}}{\sum_{y \in \mathcal{Y}(i)} e^{s(i,y)}} ,$$

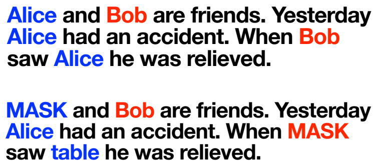
where $\mathcal{Y}(i) = \{\epsilon, 1, \dots, i-1\}$ and ϵ is a dummy antecedent to represent the event that span i is not a mention or it has no antecedents. Note that both the $s_m(i)$ and the pairwise function $s(i, j)$ depend on span representations.

Next, span representations are "refined" using the antecedent distribution as an attention mechanism (see Lee et al., 2018) which in turn affects $P(y_i)$. Finally, to cluster mentions, the antecedent distribution is queried for the most probable antecedent for each mention. The mention clusters are induced by these links. Mentions with ϵ as most probable antecedent indicate a new cluster, but if no other mention links to them as antecedents, they are pruned. Training on labeled data is done by maximizing the probability of ground-truth antecedents.

3 Pre-training Process

Our goal is to propose an approach for pre-training mention representations on unlabeled data, such that they can be more readily used for coreference resolution. Namely, the goal is that after pre-training, we can use the mention representation to train a coreference resolution system with relatively little labeled data. Next, we propose two objectives for this pre-training process.

Most previous approaches to pre-training (Joshi et al., 2019a) use only the BERT model while pre-training on unlabeled data. Here we propose to pre-train the coreference model in Sec. 2. The motivation is that we want to directly train the mention representation and scores of this model such that it will be "ready" for training on labeled data.



Alice and Bob are friends. Yesterday
Alice had an accident. When Bob
saw Alice he was relieved.

MASK and Bob are friends. Yesterday
Alice had an accident. When MASK
saw table he was relieved.

Figure 1: An illustration of the name masking objective. On the top is the original sentence with colors corresponding to repeated names. Bottom is the same sentence with some of the mentions replaced with random tokens and [MASK]. The self-supervision task is to cluster the red and blue mentions using the coreference model.).

3.1 Pre-training via Name Masking

Texts typically contain multiple appearances of the same-named entity. For example: "Alice was late. When Bob saw Alice he was relieved". In these cases, it is almost certain that the two occurrences of Alice correspond to the same cluster. Our key observation is that this signal persists even in the sentence "Alice was late. When [MASK] saw [MASK] he was relieved". In this case, the information that the two mentions have the same name is no longer available, but sentence context is sufficient for understanding that the second [MASK] corefers with Alice. Here we further consider a more challenging setup where instead of [MASK] we use a random token.

The above intuition provides a highly effective task for training coreference models: take sentences with multiple names, mask some of the occurrences, and train a coreference model to place the masked names in the correct cluster. To implement this idea, we need to decide on the candidate mentions set. We do not want to use all mentions in the text since, for most of these, we don't have ground-truth clusters. Thus, we only consider mentions that contain proper names that appear one or more times. We set the ground truth for these mentions according to their names, and then replace some of these with [MASK] or random tokens.

To summarize, given a text, we find mention clusters based on repeated names, create ground truth clusters based on those, and replace some of the mentions with [MASK] or random tokens. Finally, we use as loss the standard coreference loss of the model in Sec. 2 when restricted only to these mentions. See Fig. 1 for an illustration.

3.2 Pre-training via Pronoun Masking

Pronouns are abundant in text, and of course highly informative about coreference structure. Next, we show that pronouns can provide a simple yet effective self-supervision signal. Consider the sentence "Bob knew Alice and thought very highly of [MASK]", and assume we know that [MASK] is a pronoun. In this case, we have enough information about Alice to correctly predict the masked pronoun is "her". In particular, we would like the mention representation of [MASK] to be sufficient for predicting the pronoun since this reflects that the representation carries information about the mention that is relevant for coreference decisions.

Formally, let S denote the set of personal pronouns. Given a sentence with pronoun $w \in S$ in the ith mention, replace the pronoun with [MASK], and obtain the representation $\boldsymbol{r}_i$. Next, predict a pronoun from $\boldsymbol{r}_i$ using a feed-forward neural network with one hidden layer, FFNN, and take the cross-entropy loss for the ground-truth pronoun w. Formally, we optimize: $\text{CE}(\text{softmax}(\text{FFNN}(\boldsymbol{r}_i)), w)$.

4 Fine-tuning

After the pre-training process in Sec. 3, we fine-tune the model on the GAP dataset. Unlike Ontonotes, GAP is partially labeled: only one pronoun and two names are labeled, even if additional entities exist in the text. Partial labeling poses a challenge for coreference models that have a learnable mention detection phase, since the ground truth excludes mentions, and during training the model learns to falsely label their spans as non-mentions. To accommodate this, we change the baseline loss to consider only gold mentions, i.e., we optimize the log-likelihood of the correct antecedent filtered only for the gold mentions. We define it as $\mathcal{L}_1(D)$ for a document D.

We found it useful to additionally train for correct mention detection in the objective. We add a mention auxiliary loss $\mathcal{L}_2(D)$ in the form of cross-entropy on the predicted mention score s_m. Finally, we optimize:

$$\mathcal{L}_1(D) + \lambda \cdot \mathcal{L}_2(D) \qquad (1)$$

5 Related Work

Several works have set out to improve mention representations for coreference resolution (Lee et al., 2018; Kantor and Globerson, 2019; Joshi et al., 2019b). Lee et al. (2018) have refined the mention representation using attention over the antecedents of each mention. Kantor and Globerson (2019) showed the Entity Equalization approach to represent each mention in a cluster via an approximation of the sum of all mentions in its cluster. Employing Devlin et al. (2018) to extract mention representations boosts coreference resolution accuracy (Joshi et al., 2019b). Joshi et al. (2019a) increased it even further using masked whole-span prediction.

Several works have explored pre-training for coreference resolution. The recent work of Wu et al. (2019) uses Question Answering as part of the model and can thus train on QA datasets.

Other works explored self-supervision for this goal. Liu et al. (2019) uses a language model objective to train a memory network, which can resolve coreference links. Kocijan et al. (2019) finds pairs of sentences with at least two distinct personal names such that one of them is repeated. One non-first occurrence of the repeated candidate is masked, and the goal is to predict the masked name, given the correct and one incorrect candidates. They collect examples with no more than two sentences, limiting the background context the model can extract. Since only one name occurrence is masked and needs to be inferred, the model is not forced to resolve all person clusters.

Ye et al. (2020) use an approach similar to Kocijan et al. (2019), but include a Language Modeling objective. Emami et al. (2019) use names and pronouns gender information to generate links between a pronoun and a name. Again, the model needs to resolve a single coreference link for each example, instead of resolving multiple clusters.

Our proposed name masking is conceptually different since we do not try to predict names or single links, but rather use masking and random tokens to create hard coreference problems from the data. Our proposed pre-training procedure generates rich examples with multiple clusters and mentions. This lets us train a coreference model directly rather than just a BERT model.

6 Experiments

Dataset: The GAP dataset (Webster et al., 2018) is a corpus of Wikipedia snippets. Each snippet is annotated with one gender-balanced pronoun, two names, and two flags indicating whether the pronoun is coreferent with the first name, the second name, or neither (if both flags are false). The goal of the model is to detect mentions in the snippet and group them into coreference clusters. The model is then evaluated on the coreference links between the two names and the pronoun. We note that training on GAP alone is challenging since it contains only 2000/2000/454 snippets for development/test/validation sets.

The above evaluation scheme does not use the fact that there are only three marked mentions in each snippet. There are however previous works (Attree, 2019; Chada, 2019) that consider the *gold-two-mention* task (Webster et al., 2018), where the locations of the gold names and pronoun are used during inference as well[1]. We will compare our results in both scenarios: *detected-mentions*, where models need to detect the mentions by themselves, and *gold-two-mention*.

The metrics measured in this task are the overall F1, F1 on feminine and masculine examples, and bias defined as $\frac{\text{Feminine F1}}{\text{Masculine F1}}$. We use the official scorer.[2]

Training: Our experimental setup and code is built on top of the code in Joshi et al. (2019b) and SpanBERT (Joshi et al., 2019a). We pre-train on English Wikipedia unlabeled text[3] for 700k steps on the objectives defined in 3 using SpaCy NER[4] for person names extraction. See masking strategy and model's hyperparameters in Appendix A.

7 Results

We report masculine, feminine and overall F1 and feminine F1 to masculine F1 bias (Webster et al., 2018). All fine-tuning results are averages of 5 runs. Test set results for the *detected-mentions* scenario are shown in Table 1. Our baseline is the SpanBERTCoref based model from Joshi et al. (2019b), trained on GAP using the filtered loss in Sec. 4, which achieves 83.88 and 86.64 overall F1 for the base and large models, respectively. Pre-training the model before fine-tuning on GAP improves overall F1 by 2.02 and 1.92 for the base and large models. Ablation tests on the validation set shown in Table 3 indicate that each of the pre-training objectives has a significant contribution.

Test set results for the *gold-two-mention* task are shown in Table 2. The large pretrained SpanBERTCoref model achieves 92.86 overall F1, improving on Attree (2019) best single model by 0.36. Our results set a new state of the art for the GAP coreference resolution task for both the scenarios, *detected-mentions* and *gold-two-mention* for single models.

Fine-Tuning on Ontonotes: We explore another training setting of Joshi et al. (2019b), where training is only on Ontonotes (Pradhan et al., 2012) and not GAP. In this setting SpanBERT Base and Large yield an overall F1 of 85.76 and 87.5 on GAP, respectively. Our pre-training, followed by Ontonotes training, improves these to 86.12 and 87.66. Improvement is smaller than when training only on GAP, since Ontonotes is a much larger labeled dataset than GAP, thus reducing the effect of pre-training. We can also compare our results to the recent work of Wu et al. (2019). They also consider the setting of fine-tuning on Ontonotes, and report an F1 of 87.5 on GAP, as compared to our 87.66 in the same setting. However, the models are not completely comparable because Wu et al. (2019) pre-train on Quoref and SQUAD, whereas we pre-train on Wikipedia.

	F_1^M	F_1^F	$\frac{F_1^F}{F_1^M}$	F_1
Parallelism	69.4	64.4	0.93	66.9
+ URL	72.3	68.8	0.95	70.6
Lee et al. (2017)	67.8	66.3	0.98	67.0
Liu et al. (2019)	80.3	77.4	0.96	78.8
SpanBERTCoref Base	85.44	82.3	0.96	83.88
+ Our Pre-training	88.06	83.72	0.95	85.9
SpanBERTCoref Large	87.48	85.78	0.98	86.64
+ Our Pre-training	90.2	86.9	0.96	**88.56**

Table 1: Results on the GAP test set for the *detected-mentions* task. An average of 5 runs is reported. Parallelism is Webster et al. (2018)'s baseline. Lee et al. (2017) result is re-trained and reported in Liu et al. (2019). SpanBERTCoref is Joshi et al. (2019b) using SpanBERT, trained on GAP with the filtered loss defined in Sec. 4. Improvements on base and large models are significant at $p < 0.001$ (t-test).

8 Conclusion

We proposed two self-supervision tasks to improve span representations of coreference resolution models. Our approach directly optimizes the mention representations used by the coreference model, allowing it to be fine-tuned on relatively little data,

[1] Candidate mentions that may corefer with the given pronoun are set to two given mentions.

[2] github.com/google-research-datasets/gap-coreference

[3] Oct 1 2019 dump at https://dumps.wikimedia.org/enwiki

[4] https://spacy.io/usage/linguistic-features#named-entities

	F_1^M	F_1^F	$\frac{F_1^F}{F_1^M}$	F_1
Chada (2019)	91.1	87.1	0.95	89.1
Attree (2019)	94.0	91.1	0.97	92.5
Ionita et al. (2019)	92.7	90	0.97	91.4
SpanBERTCoref+Pretraining	94.2	91.58	0.97	**92.86**

Table 2: Results on the GAP test set for the *gold-two-mention* task. Best single models are reported for previous work. Last line is our model with SpanBERT Large.

	$\frac{F_1^F}{F_1^M}$	F_1
SpanBERTCoref Base	1.04	85.12
+ Pre-training names	0.99	86.54
+ Pre-training pronouns	1.01	88.08

Table 3: Ablation tests on the GAP validation set for the *detected-mentions* task. Average of 5 runs is reported.

with improved accuracy. Our results demonstrate the potential of pre-training for coreference. We believe there is much potential for additional self-supervision tasks and leave those for future work.

Acknowledgments

This project received funding from the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme (grant ERC HOLI 819080).

References

Sandeep Attree. 2019. Gendered ambiguous pronouns shared task: Boosting model confidence by evidence pooling. In *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, pages 134–146, Florence, Italy. Association for Computational Linguistics.

Rakesh Chada. 2019. Gendered pronoun resolution using BERT and an extractive question answering formulation. In *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, pages 126–133, Florence, Italy. 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.

Kevin Clark and Christopher D Manning. 2016. Deep reinforcement learning for mention-ranking coreference models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2256–2262.

Pascal Denis and Jason Baldridge. 2008. Specialized models and ranking for coreference resolution. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 660–669. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Greg Durrett, David Hall, and Dan Klein. 2013. Decentralized entity-level modeling for coreference resolution. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 114–124.

Ali Emami, Paul Trichelair, Adam Trischler, Kaheer Suleman, Hannes Schulz, and Jackie Chi Kit Cheung. 2019. The KnowRef coreference corpus: Removing gender and number cues for difficult pronominal anaphora resolution. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3952–3961, Florence, Italy. Association for Computational Linguistics.

Matei Ionita, Yury Kashnitsky, Ken Krige, Vladimir Larin, Atanas Atanasov, and Dennis Logvinenko. 2019. Resolving gendered ambiguous pronouns with BERT. In *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, pages 113–119, Florence, Italy. Association for Computational Linguistics.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2019a. SpanBERT: Improving pre-training by representing and predicting spans. *arXiv preprint arXiv:1907.10529*.

Mandar Joshi, Omer Levy, Luke Zettlemoyer, and Daniel Weld. 2019b. 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.

Vid Kocijan, Oana-Maria Camburu, Ana-Maria Cretu, Yordan Yordanov, Phil Blunsom, and Thomas Lukasiewicz. 2019. WikiCREM: A large unsupervised corpus for coreference resolution. 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 4303–4312, Hong Kong, China. 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.

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)*, volume 2, pages 687–692.

Fei Liu, Luke Zettlemoyer, and Jacob Eisenstein. 2019. The referential reader: A recurrent entity network for anaphora resolution. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5918–5925, Florence, Italy. Association for Computational Linguistics.

Sebastian Martschat and Michael Strube. 2015. Latent structures for coreference resolution. *Transactions of the Association for Computational Linguistics*, 3:405–418.

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. 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. Association for Computational Linguistics.

Kellie Webster, Marta Recasens, Vera Axelrod, and Jason Baldridge. 2018. Mind the gap: A balanced corpus of gendered ambiguou. In *Transactions of the ACL*, page to appear.

Sam Wiseman, Alexander M Rush, and Stuart M Shieber. 2016. Learning global features for coreference resolution. *arXiv preprint arXiv:1604.03035*.

Wei Wu, Fei Wang, Arianna Yuan, Fei Wu, and Jiwei Li. 2019. Coreference resolution as query-based span prediction.

Deming Ye, Yankai Lin, Jiaju Du, Zhenghao Liu, Maosong Sun, and Zhiyuan Liu. 2020. Coreferential reasoning learning for language representation.

A Training

A.1 Masking

For names and pronouns, we use the stochastic masking strategy of BERT: mask with 80% probability, replace with a random token with 10% probability, and keep the original tokens with 10% probability. Perturbing almost all names and pronouns mandates the coreference model to store contextual information in each mention representation. We down-sample masculine pronouns to reduce gender-bias by excluding 60% of them from the objective. Following these steps, we are able to generate 10M examples.

A.2 Simultaneous Optimization of Objectives

Masking names and pronouns in the same text would render the pronoun completion task extremely difficult and, in most cases, impossible. Instead, we defined an objective per each self-supervised task and alternate between masked names examples with the coreference resolution objective and masked pronouns examples with the pronoun completion objective. Practically, we have examples of both types in the same batch. This strategy allows for simultaneous optimization of both objectives without feeding the model a text segment with all its names and pronouns masked.

A.3 Model

The feed-forward neural network for pronoun completion is defined with the same hyperparameters of the FFNN defined in the baseline model (Joshi et al., 2019b,a): one hidden layer with 3000 units. We implemented gradient accumulation, i.e., we run a forward and backward pass for n examples sequentially, sum the gradients, and only then apply them on the model's weights. Using this process, we multiply the effective batch size by n. While the actual batch size is 1 in each forward and backward pass, using gradient accumulation, we increased the effective batch size to 16 and 18 for the base and large models, respectively. The base model, containing 160M parameters, was pre-trained using a V100 GPU for 24 days, and the large model, containing 409M parameters, was pre-trained on 6 V100 for 12 days. For fine-tuning, λ is set to 32, maximizing overall F1 on the validation set (λ's search space is $\{\frac{1}{16}, \frac{1}{8}, \frac{1}{4}, ..., 64\}$). We fine-tune the model with the original training configuration, with our objective as defined in (1). The base model is fine-tuned using a Titan X GPU for

4 hours, and large using V100 for 10 hours. For both pre-training and fine-tuning, the rest of the hyperparameters are kept from previous work.