PreCo: A Large-scale Dataset in Preschool Vocabulary for Coreference Resolution

Hong Chen[1], Zhenhua Fan[1, 2], Hao Lu[1], Alan L. Yuille[3] and Shu Rong[1]
[1]Preschool Lab, Yitu Tech
[2]Shandong Normal University
[3]Department of Cognitive Science and Department of Computer Science,
Johns Hopkins University
{hong.chen,zhenhua.fan,hao.lv,shu.rong}@yitu-inc.com
ayuille1@jhu.edu

Abstract

We introduce PreCo, a large-scale English dataset for coreference resolution. The dataset is designed to embody the core challenges in coreference, such as entity representation, by alleviating the challenge of low overlap between training and test sets and enabling separated analysis of mention detection and mention clustering. To strengthen the training-test overlap, we collect a large corpus of 38K documents and 12.5M words which are mostly from the vocabulary of English-speaking preschoolers. Experiments show that with higher training-test overlap, error analysis on PreCo is more efficient than the one on OntoNotes, a popular existing dataset. Furthermore, we annotate singleton mentions making it possible for the first time to quantify the influence that a mention detector makes on coreference resolution performance. The dataset is freely available at https://preschool-lab.github.io/PreCo/.

1 Introduction

Coreference resolution, identifying mentions that refer to the same entities, is an important NLP problem. Resolving coreference is critical for many downstream applications, such as reading comprehension, translation, and text summarization. Identifying a mention depends not only on its lexicons but also its contexts, and requires representations of all the entities before the mention. This is still a challenging task for the approaches based on the cutting-edge word2vec-like lexical representation. For example, it is hard to identify the mention "he" between two entities "Tom" and "Jerry" because they have almost the same word embeddings.

A number of datasets have been proposed to study the coreference resolution problem, such as MUC (Hirschman and Chinchor, 1997), ACE (Doddington et al., 2004), and OntoNotes (Pradhan et al., 2012). The most popular one is OntoNotes, and recent work on coreference resolution (Clark and Manning, 2016a,b; Lee et al., 2017; Peters et al., 2018) evaluated their models on it. Other datasets were rarely studied after OntoNotes was published.

Previous work (Sadat Moosavi and Strube, 2017) suggests that the overlap between training and test sets makes significant impact on the performance of current coreference resolvers. In OntoNotes, which has relatively low training-test overlap, this impact is mixed together with the core challenges of coreference resolution. For example, consider the failure of referencing "them" to "the wounded" in "..., the wounded were carried off so fast and it was difficult to count them". It is hard to tell whether the algorithm can succeed if the currently low-frequency phrase "the wounded" has not been seen enough times in the training set. From a machine learning perspective, high overlap is needed to ensure that the training and test datasets have similar statistics.

Another limitation of OntoNotes is that it only has annotations for non-singleton mentions, while singleton mentions are not annotated. Most of the algorithms for coreference resolution have two steps: mention detection and mention clustering (Wiseman et al., 2016; Clark and Manning, 2016a,b). The lack of singleton mention annotations makes training and evaluation of mention detectors more difficult.

To address both limitations of OntoNotes, we build a new dataset, PreCo. To alleviate the negative impact of low training-test overlap, we restrict the data domain and collect a sufficient amount of data to achieve a relatively high training-test overlap. Restricting the data domain is a common way to enable better studies of unsolved NLP tasks, such as language modeling (Hill et al., 2015) and

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 172–181
Brussels, Belgium, October 31 - November 4, 2018. ©2018 Association for Computational Linguistics

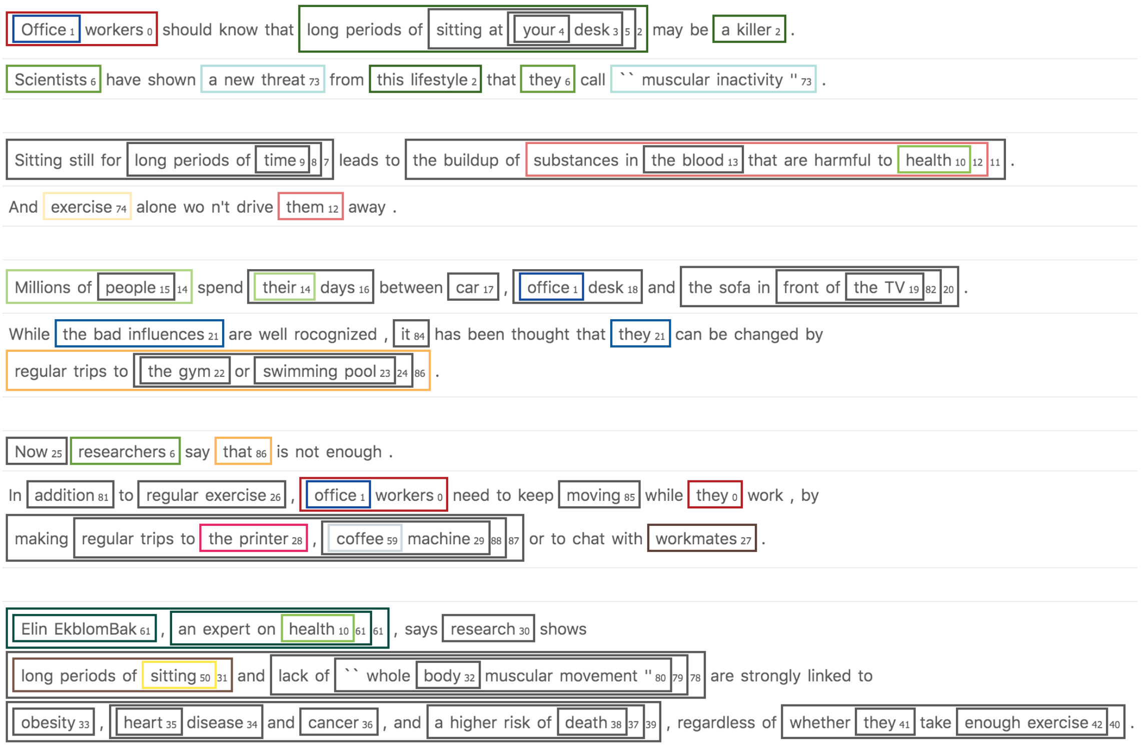

Office 1 workers 0 should know that long periods of sitting at your 4 desk 3 5 2 may be a killer 2 .
Scientists 6 have shown a new threat 73 from this lifestyle 2 that they 6 call `` muscular inactivity '' 73 .
Sitting still for long periods of time 9 8 7 leads to the buildup of substances in the blood 13 that are harmful to health 10 12 11 .
And exercise 74 alone wo n't drive them 12 away .
Millions of people 15 14 spend their 14 days 16 between car 17 , office 1 desk 18 and the sofa in front of the TV 19 82 20 .
While the bad influences 21 are well recognized , it 84 has been thought that they 21 can be changed by
regular trips to the gym 22 or swimming pool 23 24 86 .
Now 25 researchers 6 say that 86 is not enough .
In addition 81 to regular exercise 26 , office 1 workers 0 need to keep moving 85 while they 0 work , by
making regular trips to the printer 28 , coffee 59 machine 29 88 87 or to chat with workmates 27 .
Elin EkblomBak 61 , an expert on health 10 61 61 , says research 30 shows
long periods of sitting 50 31 and lack of `` whole body 32 muscular movement '' 80 79 78 are strongly linked to
obesity 33 , heart 35 disease 34 and cancer 36 , and a higher risk of death 38 37 39 , regardless of whether they 41 take enough exercise 42 40 .

Figure 1: An Example from PreCo. In the example, mentions are indicated by boxes, and mention clustering is indicated by the subscripted numbers. If two mentions have the same number, they refer to the same entity.

visual question answering (Johnson et al., 2017).

We select our data from English reading comprehension tests for middle and high school Chinese students, which has several advantages. On one hand, the vocabulary size is appropriate. The English vocabulary of a typical Chinese high school student contains about 3000 commonly used words. This is similar to the vocabulary of a preschool English-speaking child (Wikipedia, 2018). Most words from the English tests are in this limited vocabulary. On the other hand, it is practical to collect enough data of this type from the Internet. With 12.5M words, PreCo is about 10 times larger than OntoNotes. Large scale datasets, e.g. ImageNet (Deng et al., 2009), SQuAD (Rajpurkar et al., 2016), have played an important role for driving computer vision and NLP forward.

We use the rate of out-of-vocabulary (OOV) words between training and test sets to measure their overlap. PreCo shows much higher training-test overlap than OntoNotes by having an OOV rate of 0.8%, which is about 1/3 of OntoNotes's 2.1%. At the same time, PreCo presents a good challenge for coreference resolution research since its documents are in the open domain and have various writing styles. We test a state-of-the-art system (Peters et al., 2018) on PreCo and get an F1 score of 81.5. However, a modest human performance (87.9, which will be described in 4.1) is much higher, verifying there remain challenges.

To help training and evaluation of mention detection, we annotate singleton mentions in PreCo. Besides singleton mentions, we follow most other annotation rules of OntoNotes to label the new dataset. We show that in a state-of-the-art coreference resolution system (Peters et al., 2018), we can improve the model performance from 77.3 to 81.6 F1 on a training set of 2.5K PreCo documents by using an oracle mention detector, and the remaining gap of 18.4 F1 to the perfect 100 F1 can only be reduced by improving mention clustering. This indicates that future work should concern more about mention clustering than mention detection.

The advantages of our proposed dataset over existing ones in coreference resolution can be summarized as follows:

- Its OOV rate is about 1/3 of OntoNotes.
- It has about 10 times larger corpus size than OntoNotes.
- It has annotated singleton mentions.

2 Related Work

Existing Datasets. The first two resources for coreference resolution study were MUC-6 and MUC-7 (Hirschman and Chinchor, 1997). The MUC datasets are too small for training and testing, containing a total of 127 documents with 65K words. The next standard dataset was ACE (Doddington et al., 2004) which has a much larger corpus of 1M words. But its annotations are restricted to a small subset of entities and are less consistent. OntoNotes (Pradhan et al., 2012) was presented to overcome those limitations. Machine learning based approaches, especially deep learning based, benefitted from this well annotated and large-scale (1.3M words) dataset. Continuous research on OntoNotes over the past 6 years improved performance by 10 F1 score (Durrett and Klein, 2013; Peters et al., 2018). Datasets after OntoNotes, such as WikiCoref (Ghaddar and Langlais, 2016), are seldom studied. Therefore, we mainly compare PreCo with OntoNotes in this paper. With a much larger scale, PreCo builds on the advantages of OntoNotes. Some of these existing datasets also have corpus in other languages, but we just focus on coreference resolution in English.

Out-of-domain Evaluation. (Sadat Moosavi and Strube, 2017) show that if coreference resolvers mainly rely on lexical representation, as it is the case in state-of-the-art ones, they are weak at generalizing to unseen domains. Even in the seen domains, the low degree of overlap for non-pronominal mentions between the training and test sets cause serious deterioration of coreference resolution performance. As a conclusion, (Sadat Moosavi and Strube, 2017) suggested that out-of-domain evaluation is a must in the literature. But we think the problem can be relieved by expanding the training data for the target domains to increase overlap, so that the field can pay more attention to the other challenges of coreference resolution.

Data Simplification. Many simplified datasets were built to enable better study on unsolved tasks. Such simplifications can guide researchers to the core problems and make data collection easier. For example, (Hill et al., 2015) introduced the Children's Book Test to distinguish the task of predicting syntactic function words from that of predicting low-frequency words for language model. The dataset helped them to develop a generalizable model with explicit memory representations. The reading comprehension dataset SQuAD (Rajpurkar et al., 2016) imposes the constraint that every answer is always a segment of the input text. This constraint benefits both labeling and evaluation of the dataset, which has significant influences in terms of benchmarks. Similarly, the reinforcement learning literature develops algorithms by studying games instead of the real world environment (Mnih et al., 2013). We hope that, with high training-test overlap, PreCo can serve as a valuable resource for research on coreference resolution.

3 Dataset Creation

We discuss the data collection and annotation in this section. The overview of the process is shown in Figure 2.

3.1 Corpus Collection

We crawl English tests from several web sites. The web pages often contain the full English tests in a lot of formats. We build an annotation website and hire annotators to manually extract the relevant contents. We have a total of 80 part-time Chinese annotators, most of whom are university students. They are required to have a minimum score in standard English tests. During annotation training, the annotators read the annotation rules, and take several practice tasks, in which they annotate sample articles, and their results are compared with ground truth side by side for them to study. Before formal annotation, the annotators will need to pass an assessment.

Some data cleaning is done during annotation, such as unifying paragraph separators, etc. The questions with answers in these tests are also extracted for future research. Finally, we use NLTK's sentence and word tokenizer (Bird et al., 2009) to tokenize the crawled text.

In addition to having annotators manually clean the data, we also use heuristic rules to further clean the data. For example, in some cases the whitespaces between two words are missing. We use a spell checker to identify and correct most of these cases. We also use heuristic rules to fix some sentence partition boundaries, e.g., to make sure opening quotes are placed at the beginning of a sentence, instead of being wrongly placed at the end of a previous sentence (closing quotes are handled similarly).

In addition to the crawled data, we include the

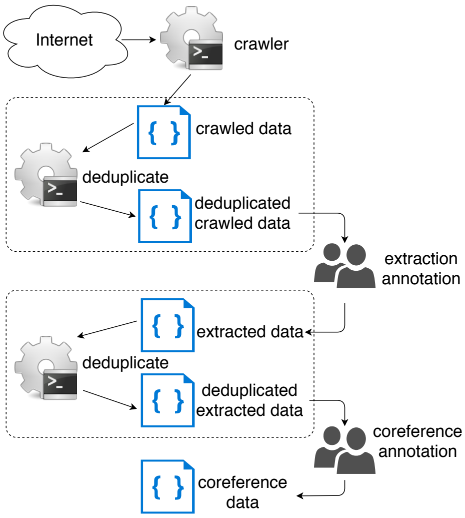


Figure 2: Overview of dataset creation.

documents from the RACE dataset (Lai et al., 2017). RACE is a reading comprehension dataset from English tests for middle and high school Chinese students, which has similar types of data sources as PreCo. About 2/3 of PreCo documents are from the RACE dataset.

Since documents are from several data sources, we want to remove duplicated documents, and documents that are not exactly the same but have a high rate of repetitions. The similarity of two documents D_1 and D_2 is estimated using the bag-of-words model. Assume S_1 and S_2 are bag-of-words multisets to represent the two documents. The similarity between D_1 and D_2 is defined as $\max(\frac{|S_1 \cap S_2|}{|S_1|}, \frac{|S_1 \cap S_2|}{|S_2|})$. If the similarity between two documents are larger than 0.9, we remove the shorter one. This process is referred as *deduplicate* in Figure 2.

3.2 Data Partition

The dataset has a total of 38K documents. We use 500 documents for the development set, 500 documents for the test set, and the rest 37K documents for the training set. The development and test documents were randomly selected from RACE's development and test sets.

3.3 Coreference Annotation and Refinement

We manually annotate coreferences on these documents. The annotation rules are slightly different from OntoNotes (Pradhan et al., 2012). We modify some of the rules to make the definition of coreference more consistent and easier to be understood by the annotators. The major differences are listed in Table 1. Figure 1 shows an example document in PreCo with annotations.

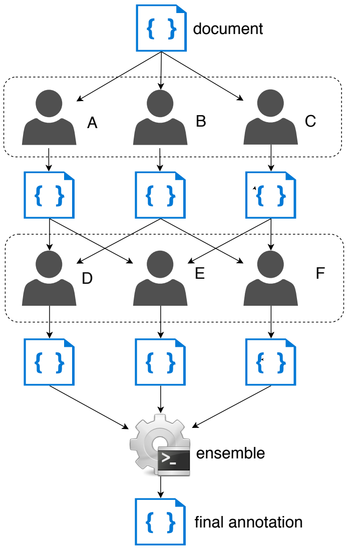


Figure 3: Process of annotation refinement. A document is firstly annotated by 3 annotators A, B, and C, independently. Then another annotator D merges annotations from A and B. Similarly, annotator E merges annotations from A and C, and annotator F merges annotations from B and C. Finally, annotations from D, E and F are merged using an ensemble algorithm.

Good quality control of annotation is essential, since the rules are complicated and coreference resolution depends on meticulous reading of the whole document over and over. We found that annotators get low recall and insufficient precision mainly because of negligence, as opposed to the lack of annotation rules or other ambiguities. For example, two co-referred mentions could be far apart and require careful searches, and an annotator may miss it. Therefore we further refine annotations as shown in Figure 3. Annotators can think about the complicated inconsistent cases when merging annotations, and the voting process will fix some errors while preserving the mentions and coreferences that are found only once by individual annotators.

The quality of different annotation processes is shown in Table 2. OntoNotes took 2 individual annotations for each document and got an adjudicated version based on them. Taking the adjudicated version as ground truth, the average MUC score (Vilain et al., 1995) [1] of individual annota-

[1] MUC score is one of the metrics to evaluate the quality of coreference resolution.

Type	Example	OntoNotes	PreCo
verbs	Sales [grew] 10%. [The growth] is exciting.	Verbs can be coreferred.	Usually, verbs cannot be coreferred. Certain gerunds can.
generic mentions	[Parents] are usually busy. [Parents] should get involved.	Generic mentions can only be coreferred by pronouns.	Generic mentions can be coreferred directly.
non-proper modifiers	[Wheat] is important. [Wheat] fields are everywhere.	Non-proper modifiers cannot be coreferred.	Non-proper modifiers can be coreferred as generic mentions.
copular structures	[John] is [a good teacher].	The referent and the attribute cannot be coreferred.	The referent and the attribute can be coreferred.
appositives	$[[\text{John}]_a$, [a linguist I know$]_b]_c$, ...	Sub-spans are not coreferred with the whole-span. a and b are not coreferent with c.	Sub-spans are coreferred with the whole-span. a and b are coreferred with c.
misc.	The [U.S.] policy ... [Secretary of State] [Colin Powell] ...	Nationality acronyms and job titles in appositives cannot be coreferred.	Nationality acronyms and all job titles can be coreferred.

Table 1: Major differences of annotation rules between PreCo and OntoNotes. The annotation rules of OntoNotes are described in (OntoNotes Guidelines)

tions is 89.6, and the inter-annotator MUC score is 83.0. The corresponding numbers for PreCo are 85.3 and 77.5. The actual gap of individual annotation quality between OntoNotes and PreCo is not as large as it looks like. Note that, OntoNotes's two individual coreference annotations of each document are based on the same syntactic annotations of the document, so they could be more consistent than PreCo's which are annotated on raw text. Therefore, if we want to fairly compare PreCo with OntoNotes, we should take into account OntoNotes's inter-annotator consistency of syntactic parsing annotations. As it has a rough upper bound of 98.5 F1 score according to the reannotation of English Treebank on OntoNotes by the principal annotator a year after the original annotation (Weischedel et al., 2011), we could infer that the individual annotation quality of PreCo is quite close to OntoNotes.

Labeling the whole dataset is costly because each annotation from scratch or comparison takes an average of about 10 minutes. Prompts from an algorithm do not help since they do not speed up the annotation much but instead introduce biases. We observed some biases when using an algorithm to help annotation. We have two models, M_1 and M_2, and we have a test set T which is annotated manually, and a test set T' which uses prompts from model M_1 to help annotation. While M_1 and M_2 have similar performance on T, M_1's performance is much higher than M_2's on T', which shows the biases.

Because of limited annotation resources, we have only finished the refinements on the development and test sets with the process shown in Figure 3. We refine the training set annotations as follows: for each document, two annotators annotate it separately, and a third annotator compares and merges the two annotations. We use a training set of 2.5K documents to quantify the impact of this annotation refinement to model performance. Table 3 shows the model performances of the training set that is annotated once, and the training set of the merged annotation. The performance difference is quite significant. Furthermore, the difference is consistent with Table 2: the "AB-merge" model has a similar precision as the "Once" model, but it has a much higher recall. It indicates that a further refinement of the training set such as DEF-voting could be essential. A more interesting question is: how to make the definition of coreference more consistent and executable? We leave it as future work.

Process	Avg. Prec	Avg. Rec	Avg. F1
Once	87.3	71.7	78.7
ABC-voting	93.5	76.1	83.9
AB-merge	87.5	88.3	87.9
DEF-voting	100.0	100.0	100.0

Table 2: Annotation quality. DEF-voting is taken as the ground truth to evaluate other annotation processes. The annotation "AB-merge" is merged by annotator G, who is different from D, E and F.

Annotation	Avg. Prec	Avg. Rec	Avg. F1
Once	79.3	69.1	73.9
AB-merge	78.1	76.5	77.3

Table 3: The annotation quality's impact on model performance. Each row shows the development set performance of the EE2E-Coref model (training details in Section 4.1) trained by data of different annotation quality. Each training set contains 2.5K documents. In the training set "Once", each document is annotated by one annotator. In the training set "AB-merge", each document is annotated by two annotators independently, and the annotations are compared and merged by a third annotator.

3.4 Dataset Properties

Table 4 shows some properties of OntoNotes and PreCo. As intended, PreCo has a lower OOV rate than OntoNotes. For a training set with vocabulary $\mathcal{V}$ and a test set with n tokens $[t_1, t_2, ..., t_n]$, ignoring the tokens with non-alphabetic characters, the OOV rate is defined by:

$$\frac{\sum_i o(t_i)}{n}, \text{where } o(t_i) = \begin{cases} 0 & \text{if } t_i \in \mathcal{V} \\ 1 & \text{if } t_i \notin \mathcal{V} \end{cases}$$

The OOV rate can be extended to the rate of low-frequency words which also indicates the training-test overlap, by simply replacing $\mathcal{V}$ in the definition above with the non-low-frequency vocabulary of the training set. We find that the OOV rate is consistent to the rates of low-frequency words in different levels. So we use the OOV rate for convenience.

In PreCo, about 50.8% of the mentions are singleton mentions. Figure 4 shows the distribution of cluster sizes within non-singleton clusters. The distribution is similar between OntoNotes and PreCo.

4 Analysis

To verify our assumption that PreCo embodies the core challenges of coreference, we evaluate a strong baseline coreference resolver on it. Specifically, we (i) estimate the room for improvement of the baseline system to show that the dataset is challenging, (ii) study the impact of training-test overlap to model performance and error analysis to show the advantages of PreCo, and (iii) quantitatively evaluate the mention detector to understand the bottlenecks of the coreference resolution system.

Property	OntoNotes	PreCo
Training documents	2.8K	37K
Training tokens	1.3M	12.2M
Dev-test documents	0.7K	1K
Dev-test tokens	0.3M	0.3M
Tokens per document	467	330
OOV rate	2.1%	0.8%
Non-singleton mentions		
Mention length	2.29	2.02
Mention density	0.12	0.16
Cluster size	4.40	4.49
Cluster density	0.027	0.035
Singleton mentions		
Mention length	N/A	3.32
Mention density	N/A	0.16
Singleton mention rate	N/A	50.8%

Table 4: Properties of OntoNotes and PreCo. The mention (cluster) density is defined by: number of mentions (clusters) / number of tokens.

4.1 Baseline Performance

We use the end-to-end neural coreference resolver, E2E-Coref (Lee et al., 2017), enhanced by the deep contextualized word representations (Peters et al., 2018) as the baseline system, and we refer to this system as EE2E-Coref. This is the state-of-the-art model on OntoNotes, achieving a test average F1 score of 70.4, which is the main evaluation metric for coreference resolution. The metric is computed by averaging the F1 of MUC, B^3, and CEAF$_{\phi 4}$, which are three metrics of coreference resolution that have different focuses.

Our implementation EE2E-Coref[2] gets 81.5 Avg. F1 score on PreCo. We follow the setting of most hyperparameters on OntoNotes and do grid-search for the decay parameter of the learning rate and the size of the hidden layers on the development set, since these two hyperparameters are relatively sensitive to the scale of the training data. The F1 score increment from OntoNotes to PreCo is probably due to the higher overlap between the training and test sets in PreCo.

[2] It gets an F1 score of 70.0±0.3 on OntoNotes, slightly lower than the F1 score reported in the original paper.

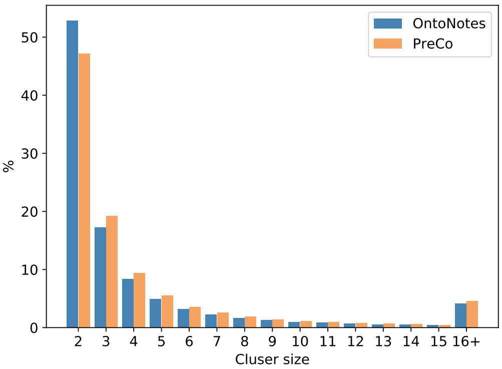


Figure 4: Distribution of cluster sizes within non-singleton clusters. We ignore singleton clusters in this figure so that it is easier to compare between OntoNotes and PreCo.

[<His father> and he] get off the car. [They] find the old man lying near the taxi. The banana skin is near him. The old man looks at [them] and says, "Teach [**your**] child to throw the banana skin to the right place!"
He gave his last few coins to [a beggar], but then he saw <another one>, and forgot that he did not have any money. He asked <the man> if <he> would like to have lunch with him, and [**the beggar**] accepted, so they went into a small restaurant and had a good meal.
[Holmes] and <Dr. Watson> went on a camping trip. After a good meal and a bottle of wine, they lay down in a tent for the night and went to sleep. Some hours later, Holmes woke up and pushed [**his friend**].

Table 5: Error cases of EE2E-Coref on PreCo. Each bold mention is incorrectly referred to the entity in []s. The mentions of its gold entity are in <>s.

We demonstrate three typical error cases made by EE2E-Coref on PreCo in Table 5. Coreference resolution in these cases requires good understanding of multiple sentences, which is an open problem in NLP. A capable entity representation for "them", "another one" or "Dr. Watson" may help to resolve these error cases. We also compare the performance of EE2E-Coref with human performance to estimate the room for improvement on PreCo. As described in Section 3.4, human annotators get low recall mostly due to negligence. So we use the AB-merge annotation to estimate human's ability on coreference resolution. The gap of performance between model and human is 6.4 F1 score, from 81.5 to 87.9. The actual gap is larger, since AB-merge still has some missed coreference annotations due to negligence. This shows that the dataset is challenging and encourages future research. The error cases show the challenges as well.

Note that PreCo is not a general purpose dataset. Our motivation of designing PreCo is to make it easier to improve coreference resolution algorithms, e.g., to make error analysis easier. It is not a goal of PreCo to generalize well on corpus from other domains. Furthermore, we find that there are a certain amount of annotation errors in the development and test sets. We suggest that researchers working on PreCo should be careful about these errors, especially after a model gets F1 score beyond 90.0.

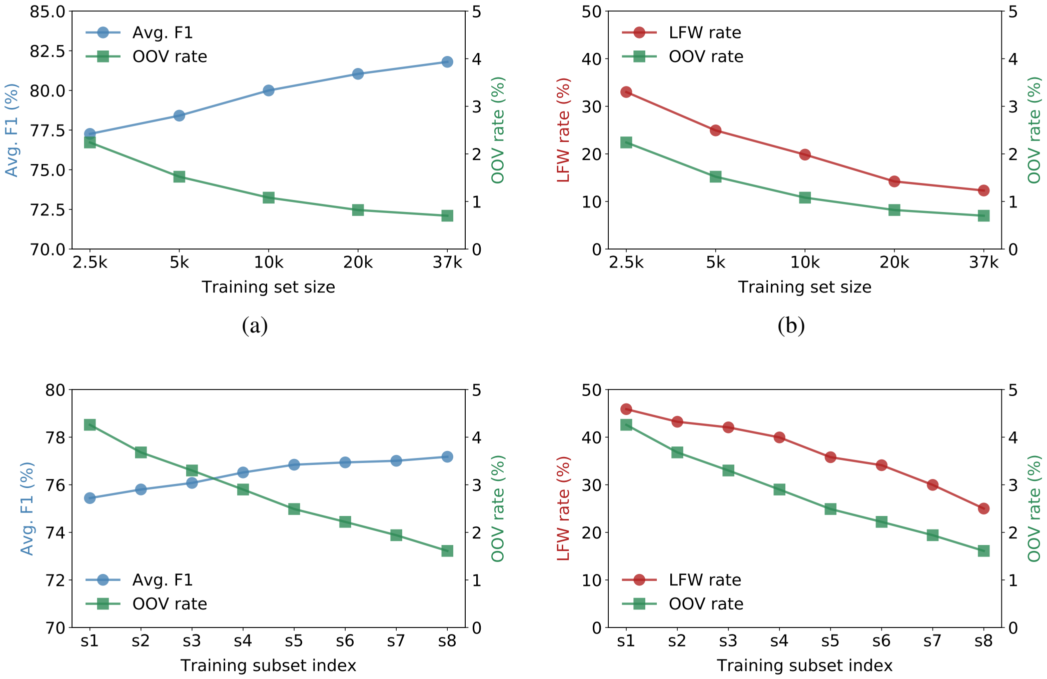
4.2 Impact of Training-test Overlap

Training-test overlap makes significant impact on error analysis. Consider an error case of coreference resolution, if there are low-frequency words in the related mentions, then it will be hard to tell whether the algorithm can succeed if the words has not been seen enough times in the training set. We call an error case LFW if there are low-frequency words[3] in its related mentions[4]. Therefore, the lower LFW rate a training set contains, the more precisely it may expose the drawbacks of the algorithm.

To study the impact of training-test overlap, actually, the training-dev overlap, we pick different subsets from the training data and evaluate the models trained on them. At first, we control overlap by picking different sizes of the training data randomly. Figure 5(a) shows that, as the training data size grows, the OOV rate, which is the overlap indicator, decreases and the F1 score of EE2E-Coref increases significantly. Figure 5(b) shows that when training set size increases, the OOV rate and the LFW rate drop together. Then, to remove the impact of data size, we pick training sets which have a fixed size but different overlaps with the development set vocabulary. The OOV rates and F1 scores of these subsets are shown in Figure 5(c). This experiment verifies the positive cor-

[3] In our experiments, a word is defined as low-frequency if it appears in the training set less than 10 times.

[4] There are 3 kinds of error cases of coreference resolution: false-new, false-link and wrong-link. In our experiments, the related mentions include: the current mention in all 3 kinds of cases, the nearest gold antecedent in false-new and wrong-link and the false referred antecedent in false-link and wrong-link.



(a)

(b)

(c)

(d)

Figure 5: Impact of training-dev overlap. (a) and (b) show the impact of training set sizes. (c) and (d) show the impact of the training-dev OOV rate, when the training sets have the same size of 2.5K documents. The 8 subsets, s1-s8, consist of documents ranked by their overlaps with the development set vocabulary.

relation between training-dev overlap and coreference resolution performance suggested by (Sadat Moosavi and Strube, 2017). Figure 5(d) shows that for training sets with the same size, the OOV rate and the LFW rate also drop together.

We observe that the training set of 2.5K documents in Figure 5(a) has a higher model performance than all the training sets in Figure 5(c). This is not expected. One hypothesis is that the lower performance in Figure 5(c) is due to the smaller diversity of these training sets, which are selected to have certain training-dev OOV rates.

The training-dev LFW rate of OntoNotes is 34.8%. As a comparison, the number for PreCo is 12.3%. A subset of PreCo with a similar token number to OntoNotes has a LFW rate of 33.0%. This indicates that research of coreference algorithms on PreCo will be much more efficient than on OntoNotes. Even if we can ignore the LFW error cases, there are others related to low-frequency word senses, phrases and sentence structures, which are hard to filter out. They will also obscure the error analysis. It is reasonable to believe that training-dev overlap impacts the rate of these error cases in a similar way to impact LFW rate.

4.3 Mention Detection

Since most coreference systems consist of a mention detection module and a mention clustering module, an important question is: with a perfect mention detection module, what is the model performance on coreference resolution? The answer would help us understand the bottlenecks of the entire system, by quantifying the impact of the mention detection module on the final F1 score. (Lee et al., 2017) gave an answer by taking ground truth non-singleton mentions as the input of the coreference resolver for both training and evaluation, assuming that the perfect mention detector can also make perfect anaphoricity decisions, e.g., to decide whether a mention should be linked to an antecedent. But this assumption can be violated since mention detectors usually take local information but anaphoricity decisions usually need more context, nearly as much as entity identification. The anaphoricity decisions should be made in the mention clustering module.

Mention	OntoNotes	PreCo
detected	66.7	77.3
*all	N/A	81.6
*non-singleton	85.2	89.2

Table 6: Coreference resolution performances on development set under different mention detection qualities. A prefixed * denotes ground truth. The model trained on OntoNotes is E2E-Coref (Lee et al., 2017) while the one trained on PreCo is EE2E-Coref. The PreCo training set contains the same 2.5K documents as in Table 3.

We argue that a better way to answer the question is to take all ground truth mentions (including singletons) for coreference. This operation is not feasible in OntoNotes since it does not have annotations for singleton mentions. We do this on PreCo and the results are shown in Table 6. There is an obvious difference between the F1 scores achieved with all gold mentions and non-singleton gold mentions. Therefore, the room for improvement by better mention detection is not as enormous as suggested in (Lee et al., 2017). The major challenge remained in coreference resolution is mention clustering.

5 Conclusion

In this paper, we propose a large-scale coreference resolution dataset to overcome the limitations of existing ones. Our dataset, PreCo, features higher training-test overlap, about 10 times larger scale than previous datasets, and singleton mention annotations. By evaluating a state-of-the-art coreference resolver, we show that there is a wide gap between the model and human performance, which demonstrated challenges of the dataset. We verified the expectation that PreCo's higher training-test overlap helps research on coreference resolution. For the first time, we quantified the impact of mention detector to the entire system, thanks to our singleton mention annotations. We make the dataset public, and hope it will stimulate further research on coreference resolution.

References

Steven Bird, Edward Loper, and Ewan Klein. 2009. *Natural Language Processing with Python.* O'Reilly Media Inc.

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

Kevin Clark and Christopher D. Manning. 2016b. Improving coreference resolution by learning entity-level distributed representations. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics.*

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE.

George R Doddington, Alexis Mitchell, Mark A Przybocki, Lance A Ramshaw, Stephanie Strassel, and Ralph M Weischedel. 2004. The automatic content extraction (ace) program-tasks, data, and evaluation. In *LREC*, volume 2, page 1.

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.

Abbas Ghaddar and Phillippe Langlais. 2016. Wikicoref: An english coreference-annotated corpus of wikipedia articles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).

Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. 2015. The goldilocks principle: Reading children's books with explicit memory representations. *arXiv preprint arXiv:1511.02301.*

Lynette Hirschman and Nancy Chinchor. 1997. Muc-7 coreference task definition. In *Proceedings of MUC-7.* Applications International Corporation.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. 2017. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 1988–1997. IEEE.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683.*

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. *CoRR*, abs/1707.07045.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. 2013. Playing atari with deep reinforcement learning. NIPS Deep Learning Workshop 2013.

OntoNotes Guidelines. 2007. Ontonotes english coreference guidelines.

M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. 2018. Deep contextualized word representations. *ArXiv e-prints*.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In *Proceedings of the Sixteenth Conference on Computational Natural Language Learning (CoNLL 2012)*, Jeju, Korea.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Nafise Sadat Moosavi and Michael Strube. 2017. Lexical features in coreference resolution: To be used with caution.

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*, MUC6 '95, pages 45–52, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ralph Weischedel, Eduard Hovy, Mitchell Marcus, Martha Palmer, Robert Belvin, Sameer Pradhan, Lance Ramshaw, and Nianwen Xue. 2011. Ontonotes: A large training corpus for enhanced processing. *Handbook of Natural Language Processing and Machine Translation. Springer*.

Wikipedia. 2018. Vocabulary development. [Online; accessed 18-April-2018].

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. Association for Computational Linguistics.