Michelle Yuan, Patrick Xia, Chandler May, Benjamin Van Durme, and **Jordan Boyd-Graber**. **Adapting Coreference Resolution Models through Active Learning**. *Association for Computational Linguistics*, 2022, 9 pages.

```
@inproceedings{Yuan:Xia:May:Durme:Boyd-Graber-2022,
Author = {Michelle Yuan and Patrick Xia and Chandler May and Benjamin Van Durme and Jordan Boyd-Graber},
Title = {Adapting Coreference Resolution Models through Active Learning},
Booktitle = {Association for Computational Linguistics},
Year = {2022},
Location = {Dublin},
Url = {http://cs.umd.edu/~jbg//docs/2022_acl_alcoref.pdf},
}
```

Links:

- Code [https://github.com/forest-snow/incremental-coref]

Contact Jordan Boyd-Graber (jbg@boydgraber.org) for questions about this paper.

Adapting Coreference Resolution Models through Active Learning

Michelle Yuan† **Patrick Xia**‡ **Chandler May**‡
Benjamin Van Durme‡ **Jordan Boyd-Graber**†
Human Language Technology Center of Excellence
†University of Maryland ‡Johns Hopkins University
myuan@cs.umd.edu paxia@cs.jhu.edu jbg@umiacs.umd.edu

Abstract

Neural coreference resolution models trained on one dataset may not transfer to new, low-resource domains. Active learning mitigates this problem by sampling a small subset of data for annotators to label. While active learning is well-defined for classification tasks, its application to coreference resolution is neither well-defined nor fully understood. This paper explores how to actively label coreference, examining sources of model uncertainty and document reading costs. We compare uncertainty sampling strategies and their advantages through thorough error analysis. In both synthetic and human experiments, labeling spans within the same document is more effective than annotating spans across documents. The findings contribute to a more realistic development of coreference resolution models.

1 Introduction

Linguistic expressions are coreferent if they refer to the same entity. The computational task of discovering coreferent mentions is coreference resolution (CR). Neural models (Lee et al., 2018; Joshi et al., 2020) are SOTA on ONTONOTES 5.0 (Pradhan et al., 2013) but cannot immediately generalize to other datasets. Generalization is difficult because domains differ in content, writing style, and annotation guidelines. To overcome these challenges, models need copiously labeled, in-domain data (Bamman et al., 2020).

Despite expensive labeling costs, adapting CR is crucial for applications like uncovering information about proteins in biomedicine (Kim et al., 2012) and distinguishing entities in legal documents (Gupta et al., 2018). Ideally, we would like to quickly and cheaply adapt the model without repeatedly relying on an excessive amount of annotations to retrain the model. To reduce labeling cost, we investigate active learning (Settles, 2009) for CR. Active learning aims to reduce annotation costs by intelligently selecting examples to label. Prior approaches use active learning to improve the model within the same domain (Gasperin, 2009; Sachan et al., 2015) without considering adapting to new data distributions. For domain adaptation in CR, Zhao and Ng (2014) motivate the use of active learning to select out-of-distribution examples. A word like "the bonds" refers to municipal bonds in ONTONOTES but links to "chemical bonds" in another domain (Figure 1). If users annotate the antecedents of "the bonds" and other ambiguous entity mentions, then these labels help adapt a model trained on ONTONOTES to new domains.

Active learning for CR adaptation is well-motivated, but the implementation is neither straightforward nor well-studied. First, CR is a span detection and clustering task, so selecting which spans to label is more complicated than choosing independent examples for text classification. Second, CR labeling involves closely reading the documents. Labeling more spans within the same context is more efficient. However, labeling more spans across different documents increases data diversity and may improve model transfer. How should we balance these competing objectives?

Our paper extends prior work in active learning for CR to the problem of coreference model transfer (Xia and Van Durme, 2021):

1. We generalize the *clustered entropy* sampling strategy (Li et al., 2020) to include uncertainty in mention detection. We analyze the effect of each strategy on coreference model transfer.
2. We investigate the trade-off between labeling and reading through simulations and a real-time user study. Limiting annotations to the same document increases labeling throughput and decreases volatility in model training.

Taken together, these contributions offer a blueprint for faster creation of CR models across domains.[1]

[1] https://github.com/forest-snow/incremental-coref

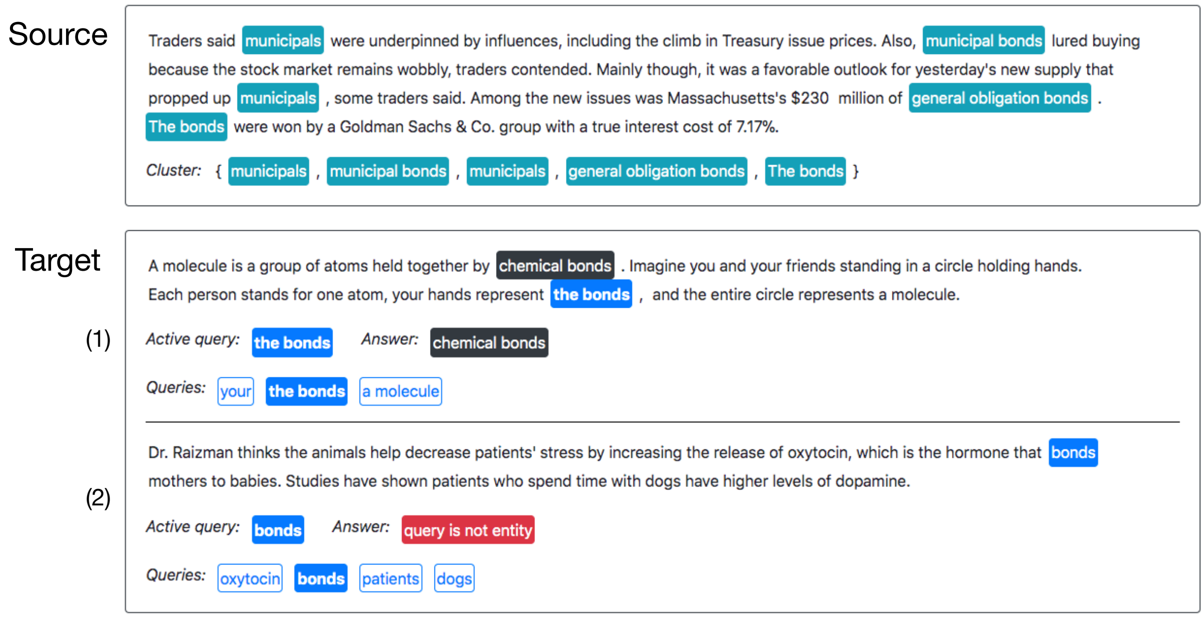


Figure 1: CR models are trained on **source** domain ONTONOTES, which contains data like news articles. The **source** document links "the bonds" to "municipal bonds". In a **target** domain like PRECO (Chen et al., 2018), "the bonds" may no longer have the same meaning. It can refer to "chemical bonds" (Document 1) or not be considered an entity (Document 2). A solution is to continue training the **source** model on more spans from the **target** domain. Active learning helps select ambiguous spans, like "the bonds", for the user to label on this interface (Section 4.2).

2 Problem: Adapting Coreference

Lee et al. (2018) introduce C2F-COREF, a neural model that outperforms prior rule-based systems. It assigns an antecedent y to mention span x. The set $\mathcal{Y}(x)$ of possible antecedent spans include a dummy antecedent ϵ and all spans preceding x. If span x has no antecedent, then x should be assigned to ϵ. Given entity mention x, the model learns a distribution over its candidate antecedents in $\mathcal{Y}(x)$,

$$P(Y=y) = \frac{\exp\{s(x,y)\}}{\sum_{y' \in \mathcal{Y}(x)} \exp\{s(x,y')\}}. \quad (1)$$

The scores $s(x,y)$ are computed by the model's pairwise scorer (Appendix A.1).

CR models like C2F-COREF are typically trained on ONTONOTES. Recent work in CR improves upon C2F-COREF and has SOTA results on ONTONOTES (Wu et al., 2020; Joshi et al., 2020). However, annotation guidelines and the underlying text differ across domains. As a result, these CR models cannot immediately transfer to other datasets. For different domains, spans could hold different meanings or link to different entities. Xia and Van Durme (2021) show the benefits of *continued training* where a model trained on ONTONOTES is further trained on the target dataset. For several target domains, continued training from ONTONOTES is stronger than training the model from scratch, especially when the training dataset is small.

Their experiments use an incremental variant of C2F-COREF called ICOREF (Xia et al., 2020). While C2F-COREF requires $\Theta(n)$ memory to simultaneously access all spans in the document and infer a span's antecedent, ICOREF only needs constant memory to predict a span's entity cluster. Despite using less space, ICOREF retains the same accuracy as C2F-COREF. Rather than assigning x to antecedent y, ICOREF assigns x to cluster c where c is from a set of observed entity clusters $\mathcal{C}$,

$$P(C=c) = \frac{\exp\{s(x,c)\}}{\sum_{c' \in \mathcal{C}} \exp\{s(x,c')\}}. \quad (2)$$

As the algorithm processes spans in the document, each span is either placed in a cluster from $\mathcal{C}$ or added to a new cluster. To learn the distribution over clusters (Equation 2), the algorithm first creates a cluster representation that is an aggregate of span representations over spans that currently exist in the cluster. With cluster and span representations, individual spans and entity clusters are mapped into a shared space. Then, we can compute $s(x,c)$ using the same pairwise scorer as before.

Xia and Van Durme (2021) show that continued training is useful for domain adaptation but assume that labeled data already exist in the target domain. However, model transfer is more critical when annotations are scarce. Thus, the question becomes: how can we adapt CR models without requiring a large, labeled dataset? Our paper investigates active learning as a potential solution. Through active learning, we reduce labeling costs by sampling and annotating a small subset of ambiguous spans.

3 Method: Active Learning

Neural models achieve high accuracy for ONTONOTES but cannot quickly adapt to new datasets because of shifts in domain or annotation standards (Poot and van Cranenburgh, 2020). To transfer to new domains, models need substantial in-domain, labeled data. In low-resource situations, CR is infeasible for real-time applications. To reduce the labeling burden, active learning may target spans that most confuse the model. Active learning for domain adaptation (Rai et al., 2010) typically proceeds as follows: begin with a model trained on source data, sample and label k spans from documents in the target domain based on a strategy, and train the model on labeled data.

This labeling setup may appear straightforward to apply to CR, but there are some tricky details. The first complication is that—unlike text classification—CR is a *clustering* task. Early approaches in active learning for CR use *pairwise annotations* (Miller et al., 2012; Sachan et al., 2015). Pairs of spans are sampled and the annotator labels whether each pair is coreferent. The downside to pairwise annotations is that it requires many labels. To label the antecedent of entity mention x, x must be compared to every candidate span in the document. Li et al. (2020) propose a new scheme called *discrete annotations*. Instead of sampling pairs of spans, the active learning strategy samples individual spans. Then, the annotator only has to find and label first antecedent of x in the document, which bypasses the multiple pairwise comparisons. Thus, we use discrete annotations to minimize labeling.

To further improve active learning for CR, we consider the following issues. First, the CR model has different scores for mention detection and linking, but prior active learning methods only considers linking. Second, labeling CR requires time to read the document context. Therefore, we explore important aspects of active learning for adapting CR: model uncertainty (Section 3.1), and the balance between reading and labeling (Section 3.2).

3.1 Uncertainty Sampling

A well-known active learning strategy is uncertainty sampling. A common measure of uncertainty is the entropy in the distribution of the model's predictions for a given example (Lewis and Gale, 1994). Labeling uncertain examples improves accuracy for tasks like text classification (Settles, 2009). For CR, models have multiple components, and computing uncertainty is not as straightforward. Is uncertainty over where mentions are located more important than linking spans? Or the other way around? Thus, we investigate different sources of CR model uncertainty.

3.1.1 Clustered Entropy

To sample spans for learning CR, Li et al. (2020) propose a strategy called *clustered entropy*. This metric scores the uncertainty in the entity cluster assignment of a mention span x. If x has *high* clustered entropy, then it should be labeled to help the model learn its antecedents. Computing clustered entropy requires the probability that x is assigned to an entity cluster. Li et al. (2020) use C2F-COREF, which only gives probability of x being assigned to antecedent y. So, they define $P(C = c)$ as the sum of antecedent probabilities $P(Y = y)$,

$$P(C = c) = \sum_{y \in C \cap \mathcal{Y}(x)} P(Y = y). \quad (3)$$

Then, they define clustered entropy as,

$$\mathrm{H}(x) = -\sum_{c \in \mathcal{C}} P(C = c) \log P(C = c). \quad (4)$$

The computation of clustered entropy in Equation 4 poses two issues. First, summing the probabilities may not accurately represent the model's probability of linking x to c. There are other ways to aggregate the probabilities (e.g. taking the maximum). C2F-COREF never computes cluster probabilities to make predictions, so it is not obvious how $P(C = c)$ should be computed for clustered entropy. Second, Equation 4 does not consider mention detection. For ONTONOTES, this is not an issue because singletons (clusters of size 1) are not annotated and mention detection score is implicitly included in $P(Y = y)$. For other datasets containing singletons, the model should disambiguate singleton clusters from non-mention spans.

To resolve these issues, we make the following changes. First, we use ICOREF to obtain cluster probabilities. ICOREF is a mention clustering model so it already has probabilities over entity clusters (Equation 2). Second, we explore other forms of maximum entropy sampling. Neural CR models have scorers for mention detection and clustering. Both scores should be considered to sample spans that confuse the model. Thus, we propose more strategies to target uncertainty in mention detection.

3.1.2 Generalizing Entropy in Coreference

To generalize entropy sampling, we first formalize mention detection and clustering. Given span x, assume X is the random variable encoding whether x is an entity mention (1) or not (0). In Section 2, we assume that the cluster distribution $P(C)$ is independent of X: $P(C) = P(C \mid X)$.[2] In other words, Equation 2 is actually computing $P(C = c \mid X = 1)$. We sample top-k spans with the following strategies.

ment-ent Highest mention detection entropy:

$$\begin{aligned} \mathrm{H}_{\text{MENT}}(x) &= \mathrm{H}(X) \\ &= -\sum_{i=0}^{1} P(X=i) \log P(X=i). \end{aligned} \tag{5}$$

The probability $P(X)$ is computed from normalized mention scores s_m (Equation 10). **Ment-ent** may sample spans that challenge mention detection (e.g. class-ambiguous words like "park"). The annotator can clarify whether spans are entity mentions to improve mention detection.

clust-ent Highest mention clustering entropy:

$$\begin{aligned} \mathrm{H}_{\text{CLUST}}(x) &= \mathrm{H}(C \mid X=1) \\ &= -\sum_{c \in \mathcal{C}} P(C=c \mid X=1) \log \\ &\quad P(C=c \mid X=1). \end{aligned} \tag{6}$$

Clust-ent looks at clustering scores without explicitly addressing mention detection. Like in ONTONOTES, all spans are assumed to be entity mentions. The likelihood $P(C = c \mid X = 1)$ is given by ICOREF (Equation 2).

cond-ent Highest conditional entropy:

$$\begin{aligned} \mathrm{H}_{\text{COND}}(x) &= \mathrm{H}(C \mid X) \\ &= \sum_{i=0}^{1} P(X=i)\mathrm{H}(C \mid X=i) \\ &= P(X=1)\mathrm{H}(C \mid X=1) \\ &= P(X=1)\mathrm{H}_{\text{CLUST}}(x). \end{aligned} \tag{7}$$

[2]A side effect of ONTONOTES models lacking singletons.

We reach the last equation because there is no uncertainty in clustering x if x is not an entity mention and $\mathrm{H}(C \mid X = 0) = 0$. **Cond-ent** takes the uncertainty of mention detection into account. So, we may sample more pronouns because they are obviously mentions but difficult to cluster.

joint-ent Highest joint entropy:

$$\begin{aligned} \mathrm{H}_{\text{JOINT}}(x) &= \mathrm{H}(X, C) = \mathrm{H}(X) + \mathrm{H}(C \mid X) \\ &= \mathrm{H}_{\text{MENT}}(x) + \mathrm{H}_{\text{COND}}(x). \end{aligned} \tag{8}$$

Joint-ent may sample spans that are difficult to detect as entity mentions *and* too confusing to cluster. This sampling strategy most closely aligns with the uncertainty of the training objective. It may also fix any imbalance between mention detection and linking (Wu and Gardner, 2021).

3.2 Trade-off between Reading and Labeling

For CR, the annotator reads the document context to label the antecedent of a mention span. Annotating and reading spans from different documents may slow down labeling, but restricting sampling to the same document may cause redundant labeling (Miller et al., 2012). To better understand this trade-off, we explore different configurations with k, the number of annotated spans, and m, the maximum number of documents being read. Given source model h_0 already fine-tuned on ONTONOTES, we adapt h_0 to a target domain through active learning (Algorithm 1):

Scoring To sample k spans from unlabeled data $\mathcal{U}$ of the target domain, we score spans with an active learning strategy S. Assume S scores each span through an *acquisition model* (Lowell et al., 2019). For the acquisition model, we use h_{t-1}, the model fine-tuned from the last cycle. The acquisition score quantifies the span's importance given S and the acquisition model.

Reading Typically, active learning samples k spans with the highest acquisition scores. To constrain m, the number of documents read, we find the documents of the m spans with highest acquisition scores and only sample spans from those documents. Then, the k sampled spans will belong to at most m documents. If m is set to "unconstrained", then we simply sample the k highest-scoring spans, irrespective of the document boundaries.

Our approach resembles Miller et al. (2012) where they sample spans based on highest uncer-

Algorithm 1 Active Learning for Coreference

Require: Source model h_0, Unlabeled data $\mathcal{U}$, Active learning strategy S, No. of cycles T, No. of labeled spans k, Max. no. of read docs m

Labeled data $\mathcal{L} = \{\}$
for cycles $t = 1, \ldots, T$ **do**
  $a_x \leftarrow$ Score span $x \in \mathcal{U}$ by $S(h_{t-1}, x)$
  $\mathcal{Q} \leftarrow$ Sort ($\downarrow$) $x \in \mathcal{U}$ by scores a_x
  $\mathcal{Q}_m \leftarrow$ Top-m spans in $\mathcal{Q}$
  $\mathcal{D} \leftarrow \{\boldsymbol{d}_x \mid x \in \mathcal{Q}_m\}$ where $\boldsymbol{d}_x$ is doc of x
  $\widetilde{\mathcal{Q}} \leftarrow$ Filter $\mathcal{Q}$ s.t. spans belong to $\boldsymbol{d} \in \mathcal{D}$
  $\widetilde{\mathcal{Q}}_k \leftarrow$ Top-k spans in $\widetilde{\mathcal{Q}}$
  $\mathcal{L}_k \leftarrow$ Label antecedents for $\widetilde{\mathcal{Q}}_k$
  $\mathcal{L} \leftarrow \mathcal{L} \cup \mathcal{L}_k$
  $h_t \leftarrow$ Continue train h_0 on $\mathcal{L}$
return h_T

tainty and continue sampling from the same document until uncertainty falls below a threshold. Then, they sample the most uncertain span from a new document. We modify their method because the uncertainty threshold will vary for different datasets and models. Instead, we use the number of documents read to control context switching.

Labeling An oracle (e.g., human annotator or gold data) labels the antecedents of sampled spans with discrete annotations (Section 3).

Continued Training We combine data labeled from current and past cycles. We train the source model h_0 (which is already trained on OntoNotes) on the labeled target data. We do not continue training a model from a past active learning cycle because it may be biased from only training on scarce target data (Ash and Adams, 2020).

4 Active Learning for CR through Simulations and Humans

We run experiments to understand two important factors of active learning for CR: sources of model uncertainty (Section 3.1) and balancing reading against labeling (Sections 3.2). First, we simulate active learning on PreCo to compare sampling strategies based on various forms of uncertainty (Section 4.1). Then, we set up a user study to investigate how humans perform when labeling spans from fewer or more documents from PreCo (Section 4.2). Specifically, we analyze their annotation time and throughput. Finally, we run large-scale simulations on PreCo and QBCoref (Section 4.3). We explore different combinations of sampling strategies and labeling configurations.

Models In all experiments, the source model is the best checkpoint of ICoref model trained on OntoNotes (Xia et al., 2020) with SpanBERT-large-cased (Joshi et al., 2020) encoder. For continued training on the target dataset, we optimize with a fixed parameter configuration (Appendix A.2). We evaluate models on Avg F_1, the averaged F_1 scores of MUC (Vilain et al., 1995), B^3 (Bagga and Baldwin, 1998), and $\text{CEAF}_{\phi 4}$ (Luo, 2005). For all synthetic experiments, we simulate active learning with gold data substituting as an annotator. However, gold mention boundaries are not used when sampling data. The model scores spans that are likely to be entity mentions for inference, so we limit the active learning candidates to this pool of high-scoring spans. For each active learning simulation, we repeat five runs with different random seed initializations.

Baselines We compare the proposed sampling strategies (Section 3.1.2) along with **li-clust-ent**, which is clustered entropy from Li et al. (2020) (Equation 4). Active learning is frustratingly less effective than random sampling in many settings (Lowell et al., 2019), so we include two random baselines in our simulation. **Random** samples from all spans in the documents. **Random-ment**, as well as other strategies, samples only from the pool of likely (high-scoring) spans. Thus, **random-ment** should be a stronger baseline than **random**.

Datasets OntoNotes 5.0 is the most common dataset for training and evaluating CR (Pradhan et al., 2013). The dataset contains news articles and telephone conversations. Only non-singletons are annotated. Our experiments transfer a model trained on OntoNotes to two target datasets: PreCo and QBCoref. PreCo is a large corpus of grade-school reading comprehension texts (Chen et al., 2018). Unlike OntoNotes, PreCo has annotated singletons. There are 37K training, 500 validation, and 500 test documents. Because the training set is so large, Chen et al. (2018) only analyze subsets of 2.5K documents. Likewise, we reduce the training set to a subset of 2.5K documents, comparable to the size of OntoNotes.

The QBCoref dataset (Guha et al., 2015) contains trivia questions from Quizbowl tournaments that are densely packed with entities from academic topics. Like PreCo, singletons are annotated. Unpreco_simple_f1.pdf

Figure 2: Test AVG F_1 on PRECO for each strategy. On each cycle, fifty spans from one document are sampled and labeled. We repeat each simulation five times. **Ment-ent**, **clust-ent**, and **joint-ent** are most effective while **random** hurts the model the most.

preco_samples.pdf

Figure 3: Cumulative counts of entities, non-entities, pronouns, and singletons sampled for each strategy over first four cycles of the PRECO simulation. **Random** mostly samples non-entities. **Li-clust-ent** and **cond-ent** sample many entity mentions but avoid singletons.

like other datasets, the syntax is idiosyncratic and world knowledge is needed to solve coreference. Examples are pronouns before the first mention of named entities and oblique references like "this polity" for "the Hanseatic League". These complicated structures rarely occur in everyday text but serve as challenging examples for CR. There are 240 training, 80 validation, and 80 test documents.

4.1 Simulation: Uncertainty Sampling

To compare different sampling strategies, we first run experiments on PRECO. We sample fifty spans from one document for each cycle. By the end of a simulation run, 300 spans are sampled from six documents. For this configuration, uncertainty sampling strategies generally reach higher accuracy than the random baselines (Figure 2), but **cond-ent** and **li-clust-ent** are worse than **random-ment**.

4.1.1 Distribution of Sampled Span Types

To understand the type of spans being sampled, we count entity mentions, non-entities, pronouns, and singletons that are sampled by each strategy (Figure 3). **Random** samples very few entities, while other strategies sample more entity mentions. **Clust-ent** and **cond-ent** sample more entity mentions and pronouns because the sampling objective prioritizes mentions that are difficult to link. **Clust-ent**, **joint-ent**, and **ment-ent** sample more singleton mentions. These strategies also show higher AVG F_1 (Figure 2). For transferring from ONTONOTES to PRECO, annotating singletons is useful because only non-singleton mentions are labeled in ONTONOTES. We notice **ment-ent** sampling pronouns, which should obviously be entity mentions, only in the first cycle. Many pronouns in ONTONOTES are singletons, so the mention detector has trouble distinguishing them initially in PRECO.

4.1.2 Error Analysis

Kummerfeld and Klein (2013) enumerate the ways CR models can go wrong: *missing entity*, *extra entity*, *missing mention*, *extra mention*, *divided entity*, and *conflated entity*. *Missing entity* means a gold entity cluster is missing. *Missing mention* means a mention span for a gold entity cluster is missing. The same definitions apply for *extra entity* and *extra mention*. *Divided entity* occurs when the model splits a gold entity cluster into multiple ones. *Conflated entity* happens when the model merges gold entity clusters. For each strategy, we analyze the errors of its final model from the simulation's last cycle (Figure 4). We compare against the **source** model that is only trained on ONTONOTES.

The **source** model makes many *missing entity* and *missing mention* errors. It does not detect several entity spans in PRECO, like locations ("Long Island") or ones spanning multiple words ("his kind

Figure 4: For each sampling strategy, we analyze the model from the last cycle of its PreCo simulation. We compare the number of errors across common error types in CR. The **source** OntoNotes model severely suffers from *missing entities* and *missing mentions*. **Ment-ent** helps most with reducing these errors.

acts of providing everything that I needed”). These spans are detected by uncertainty sampling strategies and **rand-ment**. **Ment-ent** is most effective at reducing “missing” errors. It detects gold entity clusters like “constant communication” and “the best educated guess about the storm”. By training on spans that confuse the mention detector, the model adapts to the new domain by understanding what constitutes as an entity mention.

Surprisingly, **li-clust-ent** makes at least twice as many *extra entity* and *extra mention* errors than any other strategy. For the sentence, “Living in a large building with only 10 bedrooms”, the gold data identifies two entities: “a large building with only 10 bedrooms” and “10 bedrooms”. In both OntoNotes and PreCo, the guidelines only allow the longest noun phrase to be annotated. Yet, the **li-clust-ent** model predicts additional mentions, “a large building” and “only 10 bedrooms”. We find that **li-clust-ent** tends to sample nested spans (Table 4). Due to the summed entropy computation, nested spans share similar values for clustered entropy as they share similar antecedent-linking probabilities. This causes the *extra entity* and *extra mention* errors because the model predicts there are additional entity mentions within a mention span.

Finally, we see a stark difference between **random-ment** and **random**. Out of all the sampling strategies, **random** is least effective at preventing *missing entity* and *missing mention* errors. We are more likely to sample non-entities if we randomly sample from all spans in the document (Appendix A.7). By limiting the sampling pool to only spans that are likely to be entity mentions, we sample more spans that are useful to label for CR. Thus, the mention detector from neural models should be deployed during active learning.

4.2 User Study: Reading and Labeling

We hold a user study to observe the trade-off between reading and labeling. Three annotators, with minimal NLP knowledge, label spans sampled from PreCo. We use **ment-ent** to sample spans because the strategy shows highest Avg F_1 (Figure 2). First, the users read instructions (Appendix A.6) and practice labeling for ten minutes. Then, they complete two sessions: **FewDocs** and **ManyDocs**. In each session, they label as much as possible for at least twenty-five minutes. In **FewDocs**, they read fewer documents and label roughly seven spans per document. In **ManyDocs**, they read more documents and label about one span per document.

For labeling coreference, we develop a user interface that is open-sourced (Figure 8). To label the antecedent of the highlighted span, the user clicks on a contiguous span of tokens. The interface suggests overlapping candidates based on the spans that are retained by the CR model.

In the user study, participants label at least twice as much in **FewDocs** compared to **ManyDocs** (Figure 5). By labeling more spans in **FewDocs**, the mean Avg F_1 score is also slightly higher. Our findings show that the number of read documents should be constrained to increase labeling throughput. Difference in number of labeled spans between **FewDocs** and **ManyDocs** is more pronounced when two annotators volunteer to continue labeling after required duration (Appendix A.6).

4.3 Simulation: Uncertainty Sampling and Reading-Labeling Trade-off

We finally run simulations to explore *both* sources of model uncertainty and the trade-off between reading and labeling. The earlier experiments have individually looked at each aspect. Now, we analyze the interaction between both factors to understand which combination works best for adapting CR to new domains. We run simulations on PreCo and QBCoref that trade-off the number of documents read m with the number of annotated

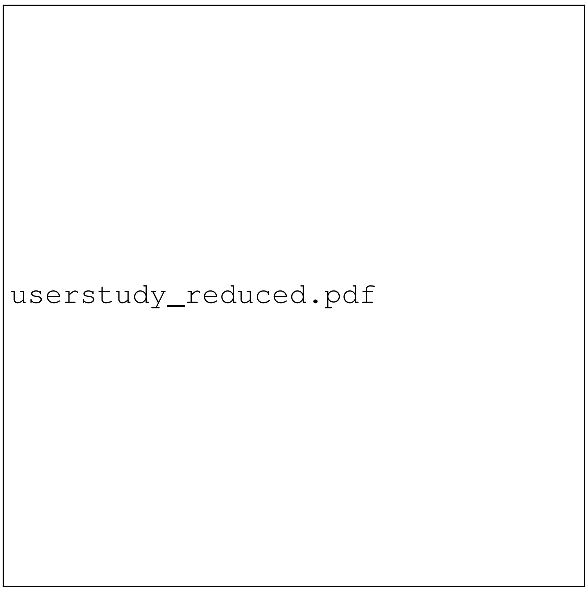
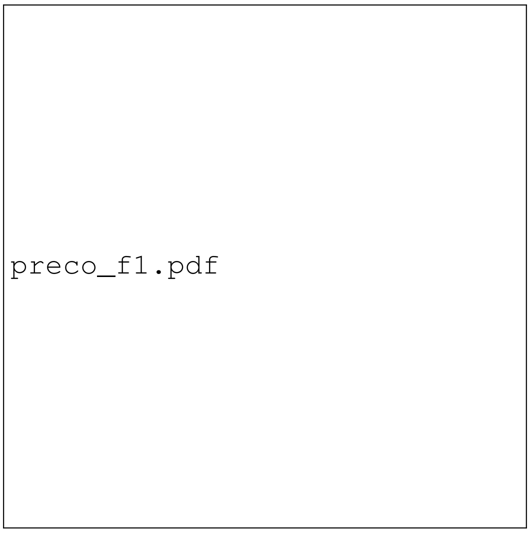


Figure 5: The number of spans labeled within twenty-five minutes. Each color indicates one of three users and the linetype designates the session. Black dots mark the first span labeled in a different document. The mean AVG F_1 across users for each session is on the right. By restricting the number of read documents in **FewDocs**, users label at least twice as many spans and the model slightly improves in AVG F_1.



(a) PRECO

(b) QBCOREF

Figure 6: Test AVG F_1 on PRECO and QBCOREF of each strategy throughout simulations. Each row varies in m, the maximum number of documents read per cycle. Each column varies in k, the number of annotated spans per cycle. For m of one or five, **ment-ent** shows highest AVG F_1 for PRECO and other uncertainty sampling strategies are best for QBCOREF. When m is unconstrained, many strategies show unstable training.

spans k (Figure 6). We vary m between one, five, and an unconstrained number of documents. For PRECO, we set k to twenty and fifty. For QBCOREF, we set k to twenty and forty. These results are also presented in numerical form (Appendix A.5).

PRECO For PRECO, the test AVG F_1 of ICOREF trained on the full training dataset is 0.860. When m is constrained to one or five, AVG F_1 can reach around 0.707 from training the model on only 300 spans sampled by **ment-ent**. As m increases, fewer spans are sampled per document and all sampling strategies deteriorate. After training on sparsely annotated documents, the model tends to predict singletons rather than cluster coreferent spans. Like in the user study, we see benefits when labeling more spans within a document. Interestingly, **li-clust-ent** performs better when document reading is not constrained to one document. The issue with **li-clust-ent** is that it samples nested mention spans (Section 4.1.2). Duplicate sampling is less severe if spans can be sampled across more documents. Another strategy that suffers from duplicate sampling is **cond-ent** because it mainly samples pronouns. For some documents, the pronouns all link to the same entity cluster. As a result, the model trains on a less diverse set of entity mentions and **cond-ent** drops in AVG F_1 as the simulation continues.

QBCOREF For QBCOREF, the test AVG F_1 of ICOREF trained on the full training dataset is 0.795. When we constrain m to one or five, **li-clust-ent**, **clust-ent**, **cond-ent**, and **joint-ent** have high AVG F_1. Clustering entity mentions in QBCOREF questions is difficult, so these strategies help target ambiguous mentions (Table 5). **Ment-ent** is less useful because demonstratives are abundant in QBCOREF and make mention detection easier. **Li-clust-ent** still samples nested entity mentions, but annotations for these spans help clarify interwoven entities in Quizbowl questions. Unlike PreCo, **li-clust-ent** does not sample duplicate entities because nested entity mentions belong to different clusters and need to be distinguished.

Overall, the most helpful strategy depends on the domain. For domains like PreCo that contain long documents with many singletons, **ment-ent** is useful. For domains like QBCoref where resolving coreference is difficult, we need to target linking uncertainty. Regardless of the dataset, **random** performs worst. **Random-ment** has much higher Avg F_1, which shows the importance of the mention detector in active learning. Future work should determine the appropriate strategy for a given domain and annotation setup.

5 Related Work

Gasperin (2009) present the first work on active learning for CR yet observe negative results: active learning is not more effective than random sampling. Miller et al. (2012) explore different settings for labeling CR. First, they label the most uncertain pairs of spans in the corpus. Second, they label all pairs in the most uncertain documents. The first approach beats random sampling but requires the annotator to infeasibly read many documents. The second approach is more realistic but loses to random sampling. Zhao and Ng (2014) argue that active learning helps domain adaptation of CR. Sachan et al. (2015) treat pairwise annotations as optimization constraints. Li et al. (2020) replace pairwise annotations with discrete annotations and experiment active learning with neural models.

Active learning has been exhaustively studied for text classification (Lewis and Gale, 1994; Zhu et al., 2008; Zhang et al., 2017). Text classification is a much simpler task, so researchers investigate strategies beyond uncertainty sampling. Yuan et al. (2020) use language model surprisal to cluster documents and then sample representative points for each cluster. Margatina et al. (2021) search for constrastive examples, which are documents that are similar in the feature space yet differ in predictive likelihood. Active learning is also applied to tasks like machine translation (Liu et al., 2018), visual question answering (Karamcheti et al., 2021), and entity alignment (Liu et al., 2021).

Rather than solely running simulations, other papers have also ran user studies or developed user-friendly interfaces. Wei et al. (2019) hold a user study for active learning to observe the time to annotate clinical named entities. Lee et al. (2020) develop active learning for language learning that adjusts labeling difficulty based on user skills. Klie et al. (2020) create a human-in-the-loop pipeline to improve entity linking for low-resource domains.

6 Conclusion

Neural CR models desparately depend on large, labeled data. We use active learning to transfer a model trained on OntoNotes, the "de facto" dataset, to new domains. Active learning for CR is difficult because the problem does not only concern sampling examples. We must consider different aspects, like sources of model uncertainty and cost of reading documents. Our work explores these factors through exhaustive simulations. Additionally, we develop a user interface to run a user study from which we observe human annotation time and throughput. In both simulations and the user study, CR improves from continued training on spans sampled from the same document rather than different contexts. Surprisingly, sampling by entropy in mention detection, rather than linking, is most helpful for domains like PreCo. This opposes the assumption that the uncertainty strategy must be directly tied to the training objective. Future work may extend our contributions to multilingual transfer or multi-component tasks, like open-domain QA.

7 Ethical Considerations

This paper involves a user study to observe the trade-off between reading and labeling costs for annotating coreference. The study has been approved by IRB to collect data about human behavior. Any personal information will be anonymized prior to paper submission or publication. All participants are fully aware of the labeling task and the information that will be collected from them. They are appropriately compensated for their labeling efforts.

Acknowledgements

We thank Ani Nenkova, Jonathan Kummerfeld, Matthew Shu, Chen Zhao, and the anonymous reviewers for their insightful feedback. We thank the user study participants for supporting this work through annotating data. Michelle Yuan and Jordan Boyd-Graber are supported in part by Adobe Inc. Any opinions, findings, conclusions, or recommendations expressed here are those of the authors and do not necessarily reflect the view of the sponsors.

References

Jordan T. Ash and Ryan P Adams. 2020. On warm-starting neural network training. In *Proceedings of Advances in Neural Information Processing Systems*.

Amit Bagga and Breck Baldwin. 1998. Algorithms for scoring coreference chains. In *The first international conference on language resources and evaluation workshop on linguistics coreference*.

David Bamman, Olivia Lewke, and Anya Mansoor. 2020. An annotated dataset of coreference in English literature. In *Proceedings of the Language Resources and Evaluation Conference*.

Hong Chen, Zhenhua Fan, Hao Lu, Alan Yuille, and Shu Rong. 2018. PreCo: A large-scale dataset in preschool vocabulary for coreference resolution. In *Proceedings of Empirical Methods in Natural Language Processing*.

Caroline Gasperin. 2009. Active learning for anaphora resolution. In *Proceedings of the NAACL HLT 2009 Workshop on Active Learning for Natural Language Processing*.

Anupam Guha, Mohit Iyyer, Danny Bouman, and Jordan Boyd-Graber. 2015. Removing the training wheels: A coreference dataset that entertains humans and challenges computers. In *Conference of the North American Chapter of the Association for Computational Linguistics*.

Ajay Gupta, Devendra Verma, Sachin Pawar, Sangameshwar Patil, Swapnil Hingmire, Girish K Palshikar, and Pushpak Bhattacharyya. 2018. Identifying participant mentions and resolving their coreferences in legal court judgements. In *International Conference on Text, Speech, and Dialogue*.

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.

Siddharth Karamcheti, Ranjay Krishna, Li Fei-Fei, and Christopher Manning. 2021. Mind your outliers! investigating the negative impact of outliers on active learning for visual question answering. In *Proceedings of the Association for Computational Linguistics*.

Jin-Dong Kim, Ngan Nguyen, Yue Wang, Jun'ichi Tsujii, Toshihisa Takagi, and Akinori Yonezawa. 2012. The genia event and protein coreference tasks of the BioNLP shared task 2011. In *BMC bioinformatics*.

Jan-Christoph Klie, Richard Eckart de Castilho, and Iryna Gurevych. 2020. From zero to hero: Human-in-the-loop entity linking in low resource domains. In *Proceedings of the Association for Computational Linguistics*.

Jonathan K. Kummerfeld and Dan Klein. 2013. Error-driven analysis of challenges in coreference resolution. In *Proceedings of Empirical Methods in Natural Language Processing*.

Ji-Ung Lee, Christian M. Meyer, and Iryna Gurevych. 2020. Empowering Active Learning to Jointly Optimize System and User Demands. In *Proceedings of the Association for Computational Linguistics*.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proceedings of Empirical Methods in Natural Language Processing*.

Kenton Lee, Luheng He, and Luke Zettlemoyer. 2018. Higher-order coreference resolution with coarse-to-fine inference. In *Conference of the North American Chapter of the Association for Computational Linguistics*.

David D. Lewis and William A. Gale. 1994. A sequential algorithm for training text classifiers. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*.

Belinda Z Li, Gabriel Stanovsky, and Luke Zettlemoyer. 2020. Active learning for coreference resolution using discrete annotation. In *Proceedings of the Association for Computational Linguistics*.

Bing Liu, Harrisen Scells, Guido Zuccon, Wen Hua, and Genghong Zhao. 2021. ActiveEA: Active learning for neural entity alignment. In *Proceedings of Empirical Methods in Natural Language Processing*.

Ming Liu, Wray Buntine, and Gholamreza Haffari. 2018. Learning to actively learn neural machine translation. In *Conference on Computational Natural Language Learning*.

David Lowell, Zachary C. Lipton, and Byron C. Wallace. 2019. Practical obstacles to deploying active learning. In *Proceedings of Empirical Methods in Natural Language Processing*.

Xiaoqiang Luo. 2005. On coreference resolution performance metrics. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*.

Katerina Margatina, Giorgos Vernikos, Loïc Barrault, and Nikolaos Aletras. 2021. Active learning by acquiring contrastive examples. In *Proceedings of Empirical Methods in Natural Language Processing*.

Timothy Miller, Dmitriy Dligach, and Guergana Savova. 2012. Active learning for coreference resolution. In *BioNLP: Proceedings of the 2012 Workshop on Biomedical Natural Language Processing*.

Corbèn Poot and Andreas van Cranenburgh. 2020. A benchmark of rule-based and neural coreference resolution in Dutch novels and news. In *Proceedings of the Third Workshop on Computational Models of Reference, Anaphora and Coreference*.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Björkelund, Olga Uryupina, Yuchen Zhang, and Zhi Zhong. 2013. Towards robust linguistic analysis using OntoNotes. In *Conference on Computational Natural Language Learning*.

Piyush Rai, Avishek Saha, Hal Daumé III, and Suresh Venkatasubramanian. 2010. Domain adaptation meets active learning. In *Conference of the North American Chapter of the Association for Computational Linguistics*.

Mrinmaya Sachan, Eduard Hovy, and Eric P Xing. 2015. An active learning approach to coreference resolution. In *International Joint Conference on Artificial Intelligence*.

Burr Settles. 2009. Active learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences.

Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. 1995. A model-theoretic coreference scoring scheme. In *Sixth Message Understanding Conference (MUC-6): Proceedings of a Conference Held in Columbia, Maryland*.

Qiang Wei, Yukun Chen, Mandana Salimi, Joshua C Denny, Qiaozhu Mei, Thomas A Lasko, Qingxia Chen, Stephen Wu, Amy Franklin, Trevor Cohen, et al. 2019. Cost-aware active learning for named entity recognition in clinical text. *Journal of the American Medical Informatics Association*, 26:1314–1322.

Wei Wu, Fei Wang, Arianna Yuan, Fei Wu, and Jiwei Li. 2020. CorefQA: Coreference resolution as query-based span prediction. In *Proceedings of the Association for Computational Linguistics*.

Zhaofeng Wu and Matt Gardner. 2021. Understanding mention detector-linker interaction for neural coreference resolution. In *Proceedings of the Fourth Workshop on Computational Models of Reference, Anaphora and Coreference*.

Patrick Xia, João Sedoc, and Benjamin Van Durme. 2020. Incremental neural coreference resolution in constant memory. In *Proceedings of Empirical Methods in Natural Language Processing*.

Patrick Xia and Benjamin Van Durme. 2021. Moving on from OntoNotes: Coreference resolution model transfer. In *Proceedings of Empirical Methods in Natural Language Processing*.

Michelle Yuan, Hsuan-Tien Lin, and Jordan Boyd-Graber. 2020. Cold-start active learning through self-supervised language modeling. In *Proceedings of Empirical Methods in Natural Language Processing*.

Ye Zhang, Matthew Lease, and Byron C. Wallace. 2017. Active discriminative text representation learning. In *Association for the Advancement of Artificial Intelligence*.

Shanheng Zhao and Hwee Tou Ng. 2014. Domain adaptation with active learning for coreference resolution. In *Proceedings of the 5th International Workshop on Health Text Mining and Information Analysis (Louhi)*.

Jingbo Zhu, Huizhen Wang, Tianshun Yao, and Benjamin K. Tsou. 2008. Active learning with sampling by uncertainty and density for word sense disambiguation and text classification. In *Proceedings of International Conference on Computational Linguistics*.

A Appendix

A.1 Coreference Resolution Models

C2F-COREF In C2F-COREF, a pairwise scorer computes $s(x, y)$ to learn antecedent distribution $P(Y)$ (Equation 1). The model's pairwise scorer judges whether span x and span y are coreferent based on their antecedent score s_a and individual mention scores s_m,

$$s(x, y) = \begin{cases} 0 & y = \epsilon \\ s_m(x) + s_m(y) + s_a(x, y) & y \neq \epsilon \end{cases}, \tag{9}$$

Suppose $\boldsymbol{g_x}$ and $\boldsymbol{g_y}$ are the span representations of x and y, respectively. Mention scores and antecedent scores are then computed with feedforward networks $FFNN_m$ and $FFNN_c$,

$$s_m(x) = FFNN_m(\boldsymbol{g_x}) \tag{10}$$
$$s_a(x, y) = FFNN_a(\boldsymbol{g_x}, \boldsymbol{g_y}, \phi(x, y)). \tag{11}$$

The input $\phi(x, y)$ includes features like the distance between spans. The unary mention score s_m can be viewed as the likelihood that the span is an entity mention. For computational purposes, the C2F-COREF model only retains top-k spans with the highest unary mention scores. Lee et al. (2018) provide more details about the pairwise scorer and span pruning.

Incremental Clustering We elaborate upon the clustering algorithm of ICOREF here. As the algorithm processes spans in the document, each span is either placed in a cluster from $\mathcal{C}$ or added to a new cluster. To learn the distribution over clusters (Equation 2), the algorithm first creates a cluster representation $\boldsymbol{g_c}$ that is an aggregate of span representation that is an aggregate of span representations over spans that currently exist in the cluster. (Equation 12). With cluster and span representations, individual spans and entity clusters are mapped into a shared space. Then, we can compute $s(x, c)$ using the same pairwise scorer as Lee et al. (2018). Suppose that model predicts c^* as most likely cluster: $c^* = \arg\max_{c \in \mathcal{C}} s(x, c)$. Now, the algorithm makes one of two decisions:

1. If $s(x, c^*) > 0$, then x is assigned to c^* and update $\boldsymbol{g_{c^*}}$ such that

$$\boldsymbol{g_{c^*}} = s_e(c^*, x)\boldsymbol{g_{c^*}} + (1 - s_e(c^*, x))\boldsymbol{g_x}, \tag{12}$$

where s_e is a learned weight.

Strategy	PRECO	QBCOREF
random	2	< 1
random-ment	4	< 1
ment-ent	5	< 1
li-clust-ent	12	< 1
clust-ent	12	1
cond-ent	14	1
joint-ent	16	1

Table 1: The time (minutes) to sample a batch of fifty spans from five documents from either PRECO or QBCOREF for a given active learning strategy. On large datasets like PRECO, we see that **li-clust-ent**, **clust-ent**, **cond-ent**, and **joint-ent** are slower because the strategy needs to incrementally cluster each span and then compute clustering entropy.

2. If $s(x, c^*) \leq 0$, then a new entity cluster $c_x = \{x\}$ is added to $\mathcal{C}$.

The algorithm repeats for each span in the document.

Like C2F-COREF, the ICOREF model only retains top-k spans with highest unary mention score. All of our active learning baselines (Section 4), except **random**, sample spans from this top-k pool of spans.

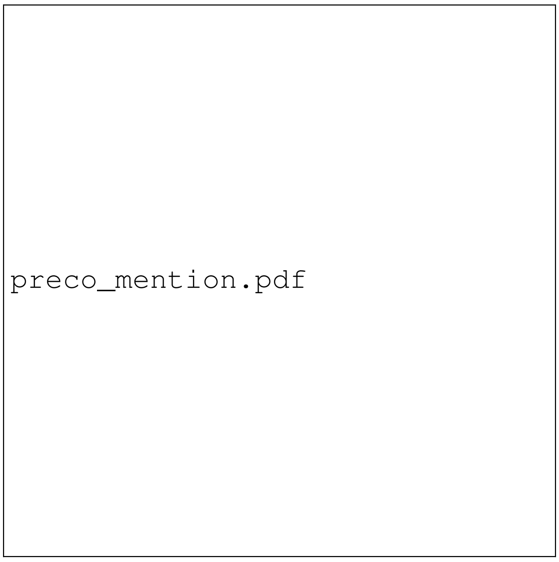
A.2 Training Configuration

The SPANBERT-LARGE-CASED encoder has 334M parameters and ICOREF has 373M parameters in total. For model fine-tuning, we train for a maximum of fifty epochs and implement early stopping with a patience of ten epochs. We set top span pruning to 0.4, dropout to 0.4, gradient clipping to 10.0, and learning rate to 1e-4 for Adam optimizer. The hyperparameter configuration is based on results from prior work (Lee et al., 2017; Xia et al., 2020).

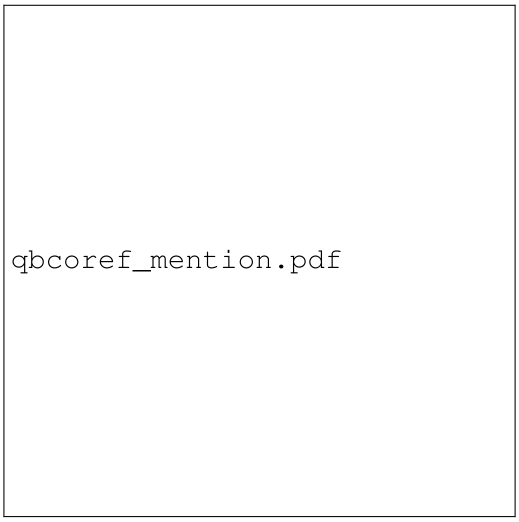
All experiments in the paper are ran on NVIDIA Tesla V100 GPU and 2.2 GHz Intel Xeon Silver 4114 CPU processor.

A.3 Simulation Time

We compare the time to sample fifty spans between different active learning strategies for PRECO and QBCOREF (Table 1). For PRECO, **clust-ent**, **cond-ent**, and **joint-ent** are slower because they need to run documents through ICOREF and get span-cluster likelihood. On the other hand, **ment-ent** only needs unary scores s_m, which is much faster to compute. Thus, for both datasets, running **ment-ent** takes about the same time as **random-ment**.



(a) PRECO



(b) QBCOREF

Figure 7: Comparing mention detection accuracy on test set for different active learning strategies across reading/labeling configurations. The plots are formatted in the same way as Figure 6. Generally, mention detection improves most from **ment-ent** sampling.

For QBCOREF, fine-tuning ICOREF on fifty spans takes three minutes and fine-tuning on full training set takes thirty-four minutes. For PRECO, fine-tuning ICOREF on fifty spans takes nine minutes and fine-tuning on full training set takes five hours and 22 minutes.

A.4 Mention Detection Accuracy

For the annotation simulation in Section 4, we also record mention detection accuracy. As **ment-ent** targets ambiguity in mention detection, it is the most effective strategy for improving mention detection (Figure 7). The strategy is unaffected by labeling setup parameters, like the number of spans labeled per cycle or the number of documents read per cycle. For strategies like **cond-ent** and **joint-ent**, mention detection accuracy is stagnant or decreases as more spans are sampled (Figure 7a). Due to deteriorating mention detection, the AVG F_1 of models also drop.

A.5 Numerical Results

The results for AVG F_1 and mention detection accuracy are presented as graphs throughout the paper. To concretely understand the differences between the methods, we provide results in numerical form (Tables 2,3). We show results from the PRECO and QBCOREF simulations where twenty spans are labeled each cycle and the number of documents read is either one or an unconstrained amount. The values in the tables show the mean and variance of AVG F_1 and mention detection accuracy over five different runs.

A.6 User Study

Instructions to Participants We give the following instructions to user study participants:

> You will be shown several sentences from a document. We have highlighted a mention (a word or phrase) of an entity (a person, place, or thing). This entity mention may be a pronoun (such as "she" or "their") or something else.
>
> We need your help to find an earlier mention of the same entity, whether in the same sentence or in an earlier sentence. The mention does not have to be the immediately previous one.
>
> If the span is not an entity mention or does not have an antecedent, please make note of it on the interface.

User Interface We design a user interface for annotators to label coreference (Figure 8). The user interface takes the sampled spans from active learning as input. Afterward, it will present the document and highlight the sampled spans in the document. The user the proceeds to go through the list of "Queries". For the "Active query", they need to either: find its antecedent, mark there is "no previous mention", or indicate that "query is not an entity". The interface will suggest some overlapping candidates to help narrow down the user's search. The candidates are spans that the CR

Total No. of Labeled Spans	m	Strategy	Avg F_1	Mention Accuracy
100	1	clust-ent	0.64 ± 0.02	0.71 ± 0.03
		cond-ent	0.57 ± 0.02	0.66 ± 0.02
		joint-ent	0.64 ± 0.03	0.76 ± 0.02
		ment-ent	$\mathbf{0.70 \pm 0.01}$	$\mathbf{0.80 \pm 0.00}$
		random	0.43 ± 0.09	0.49 ± 0.11
		random-ment	0.65 ± 0.04	0.78 ± 0.02
		li-clust-ent	0.56 ± 0.02	0.65 ± 0.03
	unconstrained	clust-ent	0.62 ± 0.03	0.70 ± 0.03
		cond-ent	0.43 ± 0.09	0.67 ± 0.04
		joint-ent	0.55 ± 0.06	0.71 ± 0.05
		ment-ent	0.65 ± 0.03	0.76 ± 0.03
		random	0.48 ± 0.07	0.54 ± 0.07
		random-ment	$\mathbf{0.69 \pm 0.01}$	$\mathbf{0.80 \pm 0.01}$
		li-clust-ent	0.62 ± 0.01	0.73 ± 0.01
200	1	clust-ent	0.68 ± 0.01	0.77 ± 0.01
		cond-ent	0.62 ± 0.02	0.70 ± 0.03
		joint-ent	0.68 ± 0.03	0.80 ± 0.02
		ment-ent	$\mathbf{0.71 \pm 0.01}$	$\mathbf{0.82 \pm 0.00}$
		random	0.48 ± 0.18	0.55 ± 0.21
		random-ment	0.65 ± 0.05	0.77 ± 0.07
		li-clust-ent	0.57 ± 0.05	0.67 ± 0.04
	unconstrained	clust-ent	0.65 ± 0.02	0.73 ± 0.03
		cond-ent	0.36 ± 0.08	0.63 ± 0.07
		joint-ent	0.40 ± 0.12	0.67 ± 0.12
		ment-ent	0.67 ± 0.03	$\mathbf{0.81 \pm 0.01}$
		random	0.49 ± 0.08	0.61 ± 0.07
		random-ment	$\mathbf{0.69 \pm 0.01}$	$\mathbf{0.81 \pm 0.00}$
		li-clust-ent	0.65 ± 0.03	0.75 ± 0.03
300	1	clust-ent	0.68 ± 0.02	0.78 ± 0.01
		cond-ent	0.61 ± 0.03	0.70 ± 0.04
		joint-ent	$\mathbf{0.69 \pm 0.02}$	0.81 ± 0.01
		ment-ent	$\mathbf{0.69 \pm 0.02}$	$\mathbf{0.82 \pm 0.00}$
		random	0.50 ± 0.09	0.58 ± 0.10
		random-ment	0.61 ± 0.10	0.81 ± 0.01
		li-clust-ent	0.63 ± 0.05	0.73 ± 0.05
	unconstrained	clust-ent	0.51 ± 0.12	0.70 ± 0.04
		cond-ent	0.33 ± 0.07	0.57 ± 0.04
		joint-ent	0.41 ± 0.05	0.69 ± 0.04
		ment-ent	0.54 ± 0.07	0.80 ± 0.02
		random	0.40 ± 0.04	0.60 ± 0.13
		random-ment	0.65 ± 0.05	$\mathbf{0.80 \pm 0.04}$
		li-clust-ent	$\mathbf{0.67 \pm 0.02}$	0.78 ± 0.01

Table 2: Results of PreCo simulation in numerical form, accompanying the graphs in Figures 6a and 7a. The table shows Avg F_1 and mention detection accuracy of experiments where twenty spans are sampled and labeled each cycle. Results are shown for m, the maximum number of documents read, equal to one and also unconstrained.

Total No. of Labeled Spans	m	Strategy	AVG F_1	Mention Accuracy
100	1	clust-ent	0.47 ± 0.06	0.62 ± 0.06
		cond-ent	0.47 ± 0.03	0.61 ± 0.03
		joint-ent	$\mathbf{0.50 \pm 0.03}$	$\mathbf{0.65 \pm 0.02}$
		ment-ent	$\mathbf{0.50 \pm 0.01}$	0.66 ± 0.03
		random	0.40 ± 0.07	0.53 ± 0.07
		random-ment	0.44 ± 0.06	0.63 ± 0.04
		li-clust-ent	0.45 ± 0.02	0.59 ± 0.03
	unconstrained	clust-ent	0.41 ± 0.05	0.59 ± 0.07
		cond-ent	0.39 ± 0.10	0.57 ± 0.05
		joint-ent	0.50 ± 0.01	0.66 ± 0.02
		ment-ent	$\mathbf{0.51 \pm 0.02}$	$\mathbf{0.69 \pm 0.01}$
		random	0.36 ± 0.08	0.48 ± 0.10
		random-ment	0.48 ± 0.02	0.65 ± 0.01
		li-clust-ent	0.47 ± 0.01	0.62 ± 0.02
200	1	clust-ent	0.52 ± 0.01	0.67 ± 0.01
		cond-ent	0.52 ± 0.02	0.66 ± 0.02
		joint-ent	$\mathbf{0.53 \pm 0.03}$	0.70 ± 0.03
		ment-ent	0.51 ± 0.02	$\mathbf{0.71 \pm 0.02}$
		random	0.40 ± 0.06	0.53 ± 0.08
		random-ment	0.48 ± 0.05	0.68 ± 0.01
		li-clust-ent	0.49 ± 0.01	0.64 ± 0.02
	unconstrained	clust-ent	0.45 ± 0.04	0.64 ± 0.06
		cond-ent	0.39 ± 0.06	0.55 ± 0.06
		joint-ent	0.48 ± 0.05	$\mathbf{0.70 \pm 0.03}$
		ment-ent	0.49 ± 0.08	0.68 ± 0.13
		random	0.34 ± 0.08	0.50 ± 0.11
		random-ment	0.49 ± 0.04	$\mathbf{0.70 \pm 0.01}$
		li-clust-ent	$\mathbf{0.50 \pm 0.03}$	0.68 ± 0.02
300	1	clust-ent	0.54 ± 0.02	0.70 ± 0.02
		cond-ent	$\mathbf{0.55 \pm 0.02}$	0.70 ± 0.02
		joint-ent	$\mathbf{0.55 \pm 0.02}$	0.74 ± 0.01
		ment-ent	0.53 ± 0.02	$\mathbf{0.75 \pm 0.02}$
		random	0.42 ± 0.05	0.55 ± 0.06
		random-ment	0.49 ± 0.03	0.69 ± 0.03
		li-clust-ent	0.53 ± 0.04	0.71 ± 0.02
	unconstrained	clust-ent	0.46 ± 0.04	0.67 ± 0.06
		cond-ent	0.42 ± 0.07	0.58 ± 0.12
		joint-ent	0.43 ± 0.11	0.68 ± 0.08
		ment-ent	0.50 ± 0.06	0.74 ± 0.04
		random	0.34 ± 0.18	0.45 ± 0.23
		random-ment	0.47 ± 0.08	$\mathbf{0.75 \pm 0.02}$
		li-clust-ent	$\mathbf{0.52 \pm 0.03}$	0.71 ± 0.01

Table 3: Results of QBCOREF simulation in numerical form, accompanying the graphs in Figures 6b and 7b. The table shows AVG F_1 and mention detection accuracy of experiments where twenty spans are sampled and labeled each cycle. Results are shown for m, the maximum number of documents read, equal to one and also unconstrained.

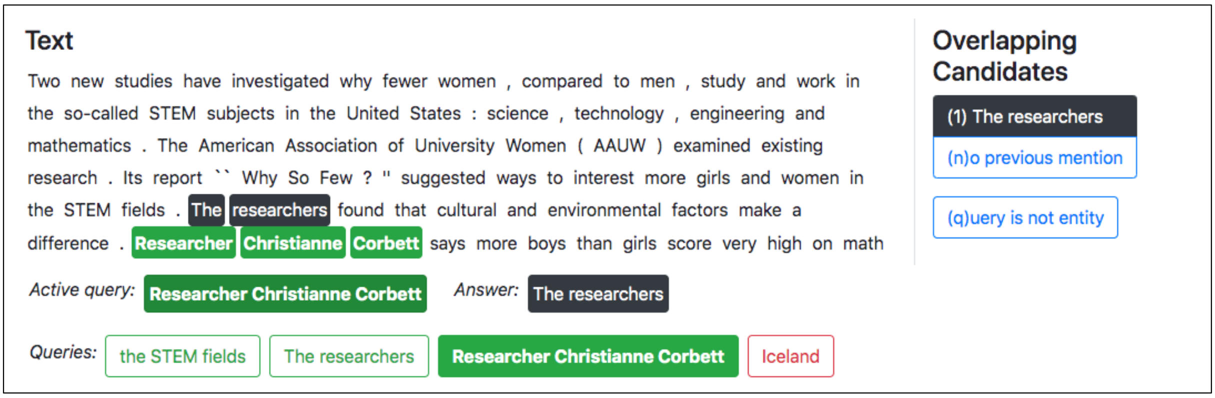


Figure 8: On the user interface, the sampled span is highlighted and the user must select an antecedent. If no antecedents exist or the span is not an entity mention, then the user will click the corresponding buttons.

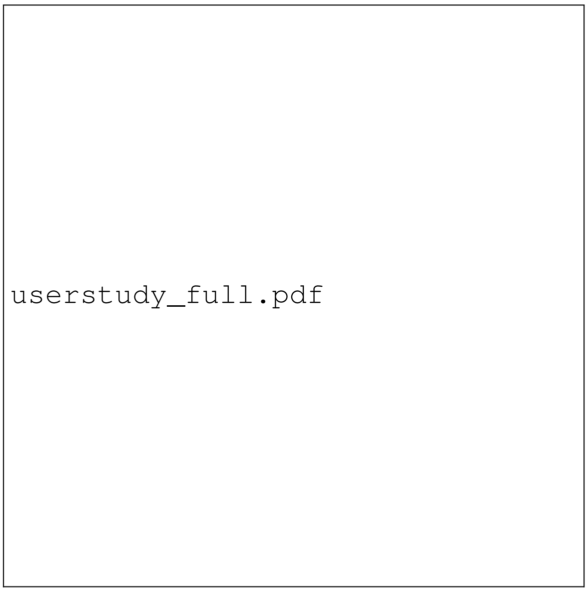


Figure 9: Full annotation times of participants (distinguished by color) during the user study. Over a longer period of time, the difference in number of labeled spans between the two sessions is much more pronounced. Within fourty-five minutes, the red user can label a hundred spans in the **FewDocs** session but only labels about thirty spans in the **ManyDocs** session.

model scores as likely entity mentions. Users may use keyboard shortcuts to minimize labeling time. The code for the user interface is released along with the code for the simulations.

Extending Annotation Time User study participants are asked to annotate at least twenty-five minutes (Section 4.2). During the study, two participants continue to label after the minimum duration. Figure 9 shows full results from the user study. Over a longer duration, the differences between the **FewDocs** and **ManyDocs** sessions are clearer.

A.7 Examples of Sampled Spans

We provide examples of spans that are sampled from the experiments. For these examples, we look at the simulation where document reading is constrained to one document and twenty spans are sampled per cycle. We compare the spans sampled by each strategy for both PreCo (Table 4) and QBCoref (Table 5). Across domains, the strategies behave similarly, but we notice some differences in **ment-ent** and **joint-ent**. In PreCo, those strategies tend to sample a mix of spans that are and are not entity mentions (Section 4.1.1). In QBCoref, they sample more entity mentions. This could be due to more entity mentions present in a Quizbowl question, which makes it more likely to sample something that should belong to an entity cluster.

For other strategies, we notice some issues. As mentioned in Section 4.1.2, **li-clust-ent** tends to sample nested entity mentions, which may become redundant for annotators to label. In fact, Avg F_1 for **li-clust-ent** tends to be lower if document reading is constrained to one document. **Cond-ent** suffers from redundant labeling because pronouns are repeatedly sampled and they tend to link to the same entity cluster.

Strategy	Sampled Spans	Comments
random	Later, I got out of the back door secretly and gave the food to the old man, whose [name I had discovered]$_1$ was Taff. I had never seen anything else as lovely as the smile of satisfaction [on]$_2$ Taff's face when he ate the food. From then on, my visits to [the old house had]$_3$ a purpose, and I enjoyed every minute of the rest of my stay.	*Sampled spans are typically not entity mentions.*
random-ment	When opening the door, his face was full of smiles and he hugged [his two children and gave [his wife]$_2$ a kiss]$_1$. Afterwards, he walked with me to the car. We passed the tree. I was so curious that I asked [him]$_3$ about what I had [seen]$_4$ earlier.	*Diverse set of span types is sampled, including spans that are not entity mentions and ones that do link to entities.*
li-clust-ent	Although [he and [his young men]$_2$]$_1$ had taken no part in the killings, he knew that [the white men]$_3$ would blame [all of [the Indians]$_5$]$_4$.	*Many sampled spans are nested entity mentions.*
ment-ent	This summer, Republicans have been [meeting]$_1$ "behind closed doors" on a Medicare proposal scheduled to be released [later this month, only a few weeks before Congress votes]$_2$ on it, thereby avoiding independent analysis of the costs, mobilization by opponents and other inconvenient aspects of a long national debate. Two years ago, the Republicans rang alarms about the [Clinton]$_3$ plan's emphasis on [managed care]$_4$	*Sampled spans are both entity mentions and non-entities. The spans are difficult for mention detection like "meeting" but may also be hard for clustering like "Clinton".*
clust-ent	After that, [Mary]$_1$ buys some school things, too. Here [mother]$_2$ buys a lot of food, like bread, cakes, meat and fish. [They]$_3$ get home very late.	*Different types of entity mentions are sampled.*
cond-ent	It is a chance to thank everyone who has contributed to shaping [you]$_1$ during the high school years; it is a chance to appreciate all those who have been instrumental in [your]$_2$ education. Take a moment to express gratitude to all those who have shared the experiences of [your]$_3$ high school years.	*More pronouns are sampled because they are obviously entity mentions and hard to cluster. However, repeated sampling of the same entity occurs.*
joint-ent	[This]$_1$ is an eternal regret handed down from generation to generation and [you]$_2$ are only one of those who languish for (...) followers. [Love]$_3$ is telephone, but it is difficult to seize [the center time for dialing]$_4$, and you will let the opportunity slip if your call is either too early or too late.	*Many entity mentions are sampled but some are difficult for mention detector to detect.*

Table 4: The example spans from PRECO documents that are sampled with each active learning strategy.

Strategy	Sampled Spans	Comments
random	The discovery of a tube behind a [fuse box alarms Linda, and the image of stock[ings]$_2$ disturbs the main]$_2$ character due to his guilt over [an encounter with a woman and his son Biff in [Boston]$_4$]$_3$.	*Choice of sampled spans are very random and do not seem to improve learning coreference.*
random-ment	The speaker of one of [this author's works]$_1$ invites the reader to [take]$_2$ a little sun, a little honey, as commanded by [Persephone's]$_3$ bees.	*Diverse set of span types is sampled, including spans that are not entity mentions and ones that do link to entities.*
li-clust-ent	For 10 points, name [this [Moliere]$_2$ play about [Argan who is constantly concerned with [his]$_4$ health]$_3$]$_1$.	*Many sampled spans are nested entity mentions.*
ment-ent	He then sees [Ignorance and Want]$_1$ emerge from [a cloak]$_2$. Earlier, he sees [a door-knocker]$_3$ [transform]$_4$ into [a human figure, which drags a belt made of chains and locks]$_5$.	*Compared to* PRECO, *more entity mentions are sampled but most sampled spans are still difficult to detect.*
clust-ent	[[Its]$_2$ protagonist]$_1$ hires Croton to rescue a different character after listening to a giant - LRB - * - RRB - Christian named Urban [discuss]$_3$ a meeting at Ostranium.	*Compared to* PRECO, *a few sampled spans are not entity mentions.*
cond-ent	While [this work]$_1$ acknowledges the soundness of the arguments that use the example of the ancients, [[its]$_3$ author]$_2$ refuses to reply to [them]$_4$, adding that we are constructing no system here [we]$_5$ are a historian, not a critic.	*More pronouns are sampled because they are obviously entity mentions and hard to cluster. Unlike* PRECO, *repeated sampling occurs less often.*
joint-ent	This man falls in love with [the maid with [lime colored panties]$_2$]$_1$ and dates [Luciana]$_3$.	*Compared to* PRECO, *more entity mentions are sampled.*

Table 5: The example spans from QBCOREF documents that are sampled with each active learning strategy.