Anupam Guha, Mohit Iyyer, Danny Bouman, and **Jordan Boyd-Graber**. **Removing the Training Wheels: A Coreference Dataset that Entertains Humans and Challenges Computers**. *North American Association for Computational Linguistics*, 2015, 11 pages.


```
    @inproceedings{Guha:Iyyer:Bouman:Boyd-Graber-2015,
Author = {Anupam Guha and Mohit Iyyer and Danny Bouman and Jordan Boyd-Graber},
Url = {docs/2015_naacl_qb_coref.pdf},
Booktitle = {North American Association for Computational Linguistics},
Location = {Denver, Colorado},
Year = {2015},
Title = {Removing the Training Wheels: A Coreference Dataset that Entertains Humans and Challenges Compute
}
```


Links:

- Code/Data [http://www.cs.umd.edu/~aguha/qbcoreference]
- Slides [http://cs.colorado.edu/~jbg/docs/2015_naacl_qb_coref_pres.pdf]
- Video [http://techtalks.tv/talks/removing-the-training-wheels-a-coreference-dataset-that-entertains-huma
  61525/]
- LaTeX [https://github.com/Pinafore/publications/tree/master/2015_naacl_qb_coref]

Removing the Training Wheels: A Coreference Dataset that Entertains Humans and Challenges Computers

Anupam Guha,[1] **Mohit Iyyer,**[1] **Danny Bouman,**[1] **Jordan Boyd-Graber**[2]
[1]University of Maryland, Department of Computer Science and UMIACS
[2]University of Colorado, Department of Computer Science
aguha@cs.umd.edu, miyyer@umiacs.umd.edu, dannybb@gmail.com,
Jordan.Boyd.Graber@colorado.edu

Abstract

Coreference is a core NLP problem. However, newswire data, the primary source of existing coreference data, lack the richness necessary to truly solve coreference. We present a new domain with denser references—quiz bowl questions—that is challenging and enjoyable to humans, and we use the quiz bowl community to develop a new coreference dataset, together with an annotation framework that can tag any text data with coreferences and named entities. We also successfully integrate active learning into this annotation pipeline to collect documents maximally useful to coreference models. State-of-the-art coreference systems underperform a simple classifier on our new dataset, motivating non-newswire data for future coreference research.

1 Introduction

Coreference resolution—adding annotations to an input text where multiple strings refer to the same entity—is a fundamental problem in computational linguistics. It is challenging because it requires the application of syntactic, semantic, and world knowledge (Ng, 2010).

For example, in the sentence *Monsieur Poirot assured Hastings that he ought to have faith in him*, the strings *Monsieur Poirot* and *him* refer to the same person, while *Hastings* and *he* refer to a different character.

There are a panoply of sophisticated coreference systems, both data-driven (Fernandes et al., 2012; Durrett and Klein, 2013; Durrett and Klein, 2014; Björkelund and Kuhn, 2014) and rule-based (Pradhan et al., 2011; Lee et al., 2011). Recent CoNLL shared tasks provide the opportunity to make a fair comparison between these systems. However, because all of these shared tasks contain strictly newswire data,[1] it is unclear how existing systems perform on more diverse data.

We argue in Section 2 that to truly solve coreference resolution, the research community needs high-quality datasets that contain many challenging cases such as nested coreferences and coreferences that can only be resolved using external knowledge. In contrast, newswire is deliberately written to contain few coreferences, and those coreferences should be easy for the reader to resolve. Thus, systems that are trained on such data commonly fail to detect coreferences in more expressive, non-newswire text.

Given newswire's imperfect range of coreference examples, can we do better? In Section 3 we present a specialized dataset that specifically tests a *human's* coreference resolution ability. This dataset comes from a community of trivia fans who also serve as enthusiastic annotators (Section 4). These data have denser coreference mentions than newswire text and present hitherto unexplored questions of what is coreferent and what is not. We also incorporate active learning into the annotation process. The result is a small but highly dense dataset of 400 documents with 9,471 mentions.

[1]We use "newswire" as an umbrella term that encompasses all forms of edited news-related data, including news articles, blogs, newsgroups, and transcripts of broadcast news.

We demonstrate in Section 5 that our dataset is significantly different from newswire based on results from the effective, widely-used Berkeley system (Durrett and Klein, 2013). These results motivate us to develop a very simple end-to-end coreference resolution system consisting of a CRF-based mention detector and a pairwise classifier. Our system outperforms the Berkeley system when both have been trained on our new dataset. This result motivates further exploration into complex coreference types absent in newswire data, which we discuss at length in Section 7.

2 Newswire's Limitations for Coreference

Newswire text is widely used as training data for coreference resolution systems. The standard datasets used in the MUC (MUC-6, 1995; MUC-7, 1997), ACE (Doddington et al., 2004), and CoNLL shared tasks (Pradhan et al., 2011) contain only such text. In this section we argue why this monoculture, despite its many past successes, offer diminishing results for advancing the coreference subfield.

First, newswire text has sparse references, and those that it has are mainly identity coreferences and appositives. In the CoNLL 2011 shared task (Pradhan et al., 2007) based on OntoNotes 4.0 (Hovy et al., 2006),[2] there are 2.1 mentions per sentence; in the next section we present a dataset with 3.7 mentions per sentence.[3] In newswire text, most nominal entities (not including pronouns) are singletons; in other words, they do not corefer to anything. OntoNotes 4.0 development data contains 25.4K singleton nominal entities (Durrett and Klein, 2013), compared to only 7.6K entities which corefer to something (anaphora). On the other hand, most pronominals are anaphoric, which makes them easy to resolve as pronouns are single token entities. While it is easy to obtain a lot of newswire data, the amount of coreferent-heavy mention clusters in such text is not correspondingly high.

Second, coreference resolution in news text is trivial for humans because it rarely requires world knowledge or semantic understanding. Systems trained on news media data for a related problem—entity extraction—falter on non-journalistic texts (Poibeau and Kosseim, 2001). This discrepancy in performance can be attributed to the stylistic conventions of journalism. Journalists are instructed to limit the number of entities mentioned in a sentence, and there are strict rules for referring to individuals (Boyd et al., 2008). Furthermore, writers cannot assume that their readers are familiar with all participants in the story, which requires that each entity is explicitly introduced in the text (Goldstein and Press, 2004). These constraints make for easy reading and, as a side effect, easy coreference resolution. Unlike this simplified "journalistic" coreference, everyday coreference relies heavily on inferring the identities of people and entities in language, which requires substantial world knowledge.

While news media contains examples of coreference, the primary goal of a journalist is to convey information, not to challenge the reader's coreference resolution faculty. Our goal is to evaluate coreference systems on data that taxes even human coreference.

3 Quiz Bowl: A Game of Human Coreference

One example of such data comes from a game called *quiz bowl*. Quiz bowl is a trivia game where questions are structured as a series of sentences, all of which indirectly refer to the answer. Each question has multiple clusters of mutually-coreferent terms, and one of those clusters is coreferent with the answer. Figure 1 shows an example of a quiz bowl question where all answer coreferences have been marked.

A player's job is to determine[4] the entity ref-

[2] As our representative for "newswire" data, the English portion of the Ontonotes 4.0 contains professionally-delivered weblogs and newsgroups (15%), newswire (46%), broadcast news (15%), and broadcast conversation (15%).

[3] Neither of these figures include singleton mentions, as OntoNotes does not have gold tagged singletons. Our dataset has an even higher density when singletons are included.

[4] In actual competition, it is a race to see which team can identify the coreference faster, but we ignore that aspect here.

NW	Later, [they]$_1$ all met with [President Jacques Chirac]$_2$. [Mr. Chirac]$_2$ said an important first step had been taken to calm tensions.
NW	Around the time of the [Macau]$_1$ handover, questions that were hot in [the Western media]$_2$ were "what is Macaense"? And what is native [Macau]$_1$ culture?
NW	[MCA]$_1$ said that [it]$_1$ expects [the proposed transaction]$_2$ to be completed no later than November 10th.
QB	As a child, [this character]$_1$ reads [[his]$_1$ uncle]$_2$ [the column]$_3$ [*That Body of Yours*]$_3$ every Sunday.
QB	At one point, [these characters]$_1$ climb into barrels aboard a ship bound for England. Later, [one of [these characters]$_1$]$_2$ stabs [the Player]$_3$ with a fake knife.
QB	[One poet from [this country]$_2$]$_1$ invented the haiku, while [another]$_3$ wrote the [*Tale of Genji*]$_4$. Identify [this homeland]$_2$ of [Basho]$_1$ and [Lady Murasaki]$_3$.

Table 1: Three newswire sentences and three quiz bowl sentences with annotated coreferences and singleton mentions. These examples show that quiz bowl sentences contain more complicated types of coreferences that may even require world knowledge to resolve.

[The Canadian rock band by [this name]] has released such albums as Take A Deep Breath, Young Wild and Free, and Love Machine and had a 1986 Top Ten single with Can't Wait For the Night. [The song by [this name]] is [the first track on Queen's Sheer Heart Attack]. [The novel by [this name]] concerns Fred Hale, who returns to town to hand out cards for a newspaper competition and is murdered by the teenage gang member Pinkie Brown, who abuses [the title substance]. [The novel] was adapted into [a 1947 film starring Richard Attenborough]; [this] was released in the US as Young Scarface. FTP, identify [the shared name of, most notably, [a novel by Graham Greene]].

Figure 1: An example quiz bowl question about the novel *Brighton Rock*. Every mention referring to the answer of the question has been marked; note the variety of mentions that refer to the same entity.

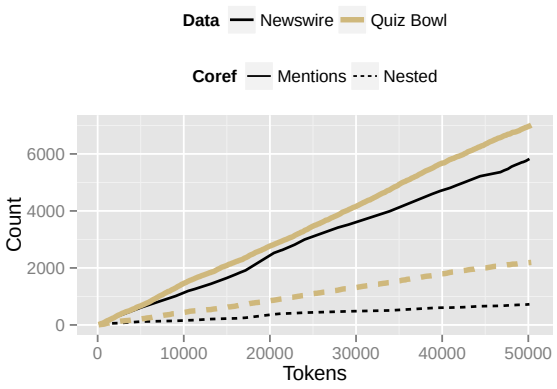

Figure 2: Density of quiz bowl vs. CoNLL coreference both for raw and nested mentions.

erenced by the question. Each sentence contains progressively more informative references and more well-known clues. For example, a question on Sherlock Holmes might refer to him as "he", "this character", "this housemate of Dr. Watson", and finally "this detective and resident of 221B Baker Street". While quiz bowl has been viewed as a classification task (Iyyer et al., 2014), previous work has ignored the fundamental task of coreference. Nevertheless, quiz bowl data are dense and diverse in coreference examples. For example, nested mentions, which are difficult for both humans and machines, are very rare in the newswire text of OntoNotes—0.25 mentions per sentence—while quiz bowl contains 1.16 mentions per sentence (Figure 2). Examples of nested mentions can be seen in in Table 1. Since quiz bowl is a game, it makes the task of solving coreference interesting and *challenging* for an annotator. In the next section, we use the intrinsic fun of this task to create a new annotated coreference dataset.

4 Intelligent Annotation

Here we describe our annotation process. Each document is a single quiz bowl question containing an average of 5.2 sentences. While quiz bowl

covers all areas of academic knowledge, we focus on questions about literature from Boyd-Graber et al. (2012), as annotation standards are more straightforward.

Our webapp (Figure 3) allows users to annotate a question by highlighting a phrase using their mouse and then pressing a number corresponding to the coreference group to which it belongs. Each group is highlighted with a single color in the interface. The webapp displays a single question at a time, and for some questions, users can compare their answers against gold annotations by the authors. We provide annotators the ability to see if their tags match the gold labels for a few documents as we need to provide a mechanism to help them learn the annotation guidelines as the annotators are crowdsourced volunteers. This improves inter-annotator agreement.

The webapp was advertised to quiz bowl players before a national tournament and attracted passionate, competent annotators preparing for the tournament. A leaderboard was implemented to encourage competitiveness, and prizes were given to the top five annotators.

Users are instructed to annotate all authors, characters, works, and the answer to the question (even if the answer is not one of the previously specified types of entities). We consider a coreference to be the maximal span that can be replaced by a pronoun.[5] As an example, in the phrase *this folk sermon by James Weldon Johnson*, the entire phrase is marked, not just *sermon* or *this folk sermon*. Users are asked to consider appositives as separate coreferences to the same entity. Thus, *The Japanese poet Basho* has two phrases to be marked, *The Japanese poet* and *Basho*, which both refer to the same group.[6] Users annotated prepositional phrases attached to a noun to capture entire noun phrases.

Titular mentions are mentions that refer to entities with similar names or the same name as a title, e.g., "The titular doctor" refers to the person "Dr. Zhivago" while talking about the book with the same name. For our purposes, all titular mentions refer to the same coreference group. We also encountered a few mentions that refer to multiple groups; for example, in the sentence *Romeo met Juliet at a fancy ball, and they get married the next day*, the word *they* refers to both *Romeo* and *Juliet*. Currently, our webapp cannot handle such mentions.

Number of ...	**Quiz bowl**	**OntoNotes**
documents[7]	400	1,667
sentences	1,890	44,687
tokens	50,347	955,317
mentions	9,471	94,155
singletons[8]	2,461	0
anaphora	7,010	94,155
nested ment.	2,194	11,454

Table 2: Statistics of both our quiz bowl dataset and the OntoNotes training data from the CoNLL 2011 shared task.

To illustrate how popular the webapp proved to be among the quiz bowl community, we had 615 documents tagged by seventy-six users within a month. The top five annotators, who between them tagged 342 documents out of 651, have an agreement rate of 87% with a set of twenty author-annotated questions used to measure tagging accuracy.

We only consider documents that have either been tagged by four or more users with a predetermined degree of similarity and verified by one or more author (150 documents), or documents tagged by the authors in committee (250 documents). Thus, our gold dataset has 400 documents.

Both our quiz bowl dataset and the OntoNotes dataset are summarized in Table 2. If coreference resolution is done by pairwise classification, our dataset has a total of 116,125 possible mention pairs. On average it takes about fifteen minutes to tag a document because often the annotator will not know which mentions co-refer

[5] We phrased the instruction in this way to allow our educated but linguistically unsavvy annotators to approximate a noun phrase.

[6] The datasets, full annotation guide, and code can be found at `http://www.cs.umd.edu/~aguha/qbcoreference`.

[7] This number is for the OntoNotes training split only.

[8] OntoNotes is not annotated for singletons.

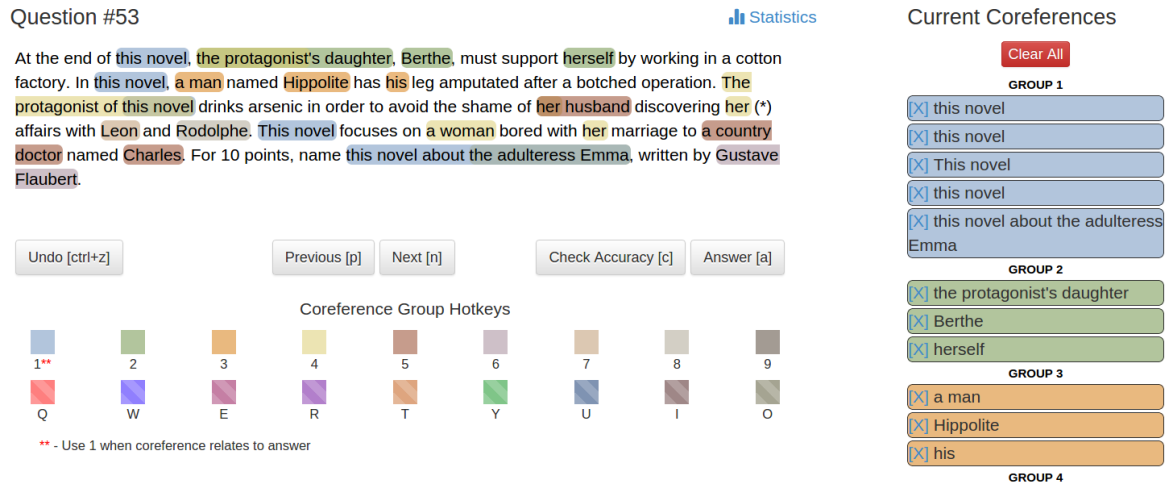

Figure 3: The webapp to collect annotations. The user highlights a phrase and then assigns it to a group (by number). Showing a summary list of coreferences on the right significantly speeds up user annotations.

to what group without using external knowledge. OntoNotes is 18.97 larger than our dataset in terms of tokens but only 13.4 times larger in terms of mentions.[9] Next, we describe a technique that allows our webapp to choose which documents to display for annotation.

4.1 Active Learning

Active learning is a technique that alternates between training and annotation by selecting instances or documents that are maximally useful for a classifier (Settles, 2010). Because of the large sample space and amount of diversity present in the data, active learning helps us build our coreference dataset. To be more concrete, the original corpus contains over 7,000 literature questions, and we want to tag only the useful ones. Since it can take a quarter hour to tag a single document and we want at least four annotators to agree on every document that we include in the final dataset, annotating all 7,000 questions is infeasible.

We follow Miller et al. (2012), who use active learning for document-level coreference rather than at the mention level. Starting from a seed set of a hundred documents and an evaluation set of fifty documents[10] we sample 250 more documents from our set of 7,000 quiz bowl questions. We use the Berkeley coreference system (described in the next section) for the training phase. In Figure 4 we show the effectiveness of our iteration procedure. Unlike the result shown by Miller et al. (2012), we find that for our dataset voting sampling beats random sampling, which supports the findings of Laws et al. (2012).

Voting sampling works by dividing the seed set into multiple parts and using each to train a model. Then, from the rest of the dataset we select the document that has the most variance in results after predicting using all of the models. Once that document gets tagged, we add it to the seed set, retrain, and repeat the procedure. This process is impractical with instance-level active learning methods, as there are 116,125 mention pairs (instances) for just 400 documents. Even with document-level sampling, the procedure of training on all documents in the seed set and then testing every document in the sample space is a slow task. Batch learning can speed up this process at the cost of increased document redundancy; we choose not to use it because we want a diverse collection of annotated documents. Active learning's advantage is that new documents are more likely to contain diverse

[9] These numbers do not include singletons as OntoNotes does not have them tagged, while ours does.

[10] These were documents tagged by the quiz bowl community, so we didn't have to make them wait for the active learning process to retrain candidate models.

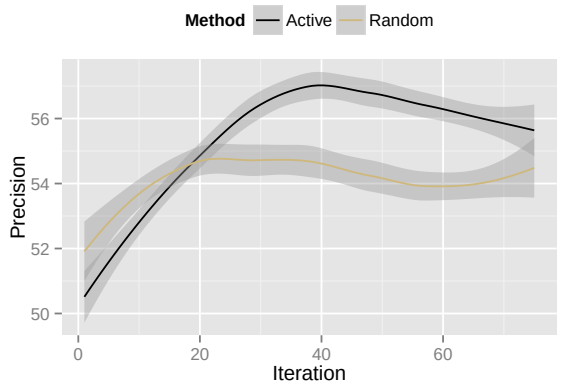

Figure 4: Voting sampling active learning works better than randomly sampling for annotation.

(and thus interesting) combinations of entities and references, which annotators noticed during the annotation process. Documents selected by the active learning process were dissimilar to previously-selected questions in both content and structure.

5 Experimental Comparison of Coreference Systems

We evaluate the widely used Berkeley coreference system (Durrett and Klein, 2013) on our dataset to show that models trained on newswire data cannot effectively resolve coreference in quiz bowl data. Training and evaluating the Berkeley system on quiz bowl data also results in poor performance. [11] This result motivates us to build an end-to-end coreference resolution system that includes a data-driven mention detector (as opposed to Berkeley's rule-based one) and a simple pairwise classifier. Using our mentions and only six feature types, we are able to outperform the Berkeley system on our data. Finally, we explore the linguistic phenomena that make quiz bowl coreference so hard and draw insights from our analysis that may help to guide the next generation of coreference systems.

[11] We use default options, including hyperparameters tuned on OntoNotes

5.1 Evaluating the Berkeley System on Quiz Bowl Data

We use two publicly-available pretrained models supplied with the Berkeley coreference system, *Surface* and *Final*, which are trained on the entire OntoNotes dataset. The difference between the two models is that *Final* includes semantic features. We report results with both models to see if the extra semantic features in *Final* are expressive enough to capture quiz bowl's inherently difficult coreferences. We also train the Berkeley system on quiz bowl data and compare the performance of these models to the pretrained newswire ones in Table 3. Our results are obtained by running a five-fold cross-validation on our dataset. The results show that newswire is a poor source of data for learning how to resolve quiz bowl coreferences and prompted us to see how well a pairwise classifier does in comparison. To build an end-to-end coreference system using this classifier, we first need to know which parts of the text are "mentions", or spans of a text that refer to real world entities. In the next section we talk about our mention detection system.

5.2 A Simple Mention Detector

Detecting mentions is done differently by different coreference systems. The Berkeley system does rule-based mention detection to detect every NP span, every pronoun, and every named entity, which leads to many spurious mentions. This process is based on an earlier work of Kummerfeld et al. (2011), which assumes that every maximal projection of a noun or a pronoun is a mention and uses rules to weed out spurious mentions. Instead of using such a rule-based mention detector, our system detects mentions via sequence labeling, as detecting mentions is essentially a problem of detecting start and stop points in spans of text. We solve this sequence tagging problem using the MALLET (McCallum, 2002) implementation of conditional random fields (Lafferty et al., 2001). Since our data contain nested mentions, the sequence labels are BIO markers (Ratinov and Roth, 2009). The features we use, which are similar to those used in Kummerfeld et al. (2011), are:

		MUC		
System	Train	P	R	F_1
Surface	OntoN	47.22	27.97	35.13
Final	OntoN	50.79	30.77	38.32
Surface	QB	**60.44**	31.31	41.2
Final	QB	60.21	**33.41**	**42.35**

Table 3: The top half of the table represents Berkeley models trained on OntoNotes 4.0 data, while the bottom half shows models trained on quiz bowl data. The MUC F_1-score of the Berkeley system on OntoNotes text is 66.4, which when compared to these results prove that quiz bowl coreference is significantly different than OntoNotes coreference.

- the token itself
- the part of speech
- the named entity type
- a dependency relation concatenated with the parent token[12]

Using these simple features, we obtain surprisingly good results. When comparing our detected mentions to gold standard mentions on the quiz bowl dataset using exact matches, we obtain 76.1% precision, 69.6% recall, and 72.7% F_1 measure. Now that we have high-quality mentions, we can feed each pair of mentions into a pairwise mention classifier.

5.3 A Simple Coref Classifier

We follow previous pairwise coreference systems (Ng and Cardie, 2002; Uryupina, 2006; Versley et al., 2008) in extracting a set of lexical, syntactic, and semantic features from two mentions to determine whether they are coreferent. For example, if *Sylvia Plath*, *he*, and *she* are all of the mentions that occur in a document, our classifier gives predictions for the pairs *he—Sylvia Plath*, *she—Sylvia Plath*, and *he—she*.

Given two mentions in a document, m_1 and m_2, we generate the following features and feed them to a logistic regression classifier:

- binary indicators for all tokens contained in m_1 and m_2 concatenated with their parts-of-speech
- same as above except for an n-word window before and after m_1 and m_2
- how many tokens separate m_1 and m_2
- how many sentences separate m_1 and m_2
- the cosine similarity of `word2vec` (Mikolov et al., 2013) vector representations of m_1 and m_2; we obtain these vectors by averaging the word embeddings for all words in each mention. We use publicly-available 300-dimensional embeddings that have been pretrained on 100B tokens from Google News.
- same as above except with publicly-available 300-dimensional `GloVe` (Pennington et al., 2014) vector embeddings trained on 840B tokens from the Common Crawl

The first four features are standard in coreference literature and similar to some of the surface features used by the Berkeley system, while the word embedding similarity scores increase our F-measure by about 5 points on the quiz bowl data. Since they have been trained on huge corpora, the word embeddings allow us to infuse world knowledge into our model; for instance, the vector for *Russian* is more similar to *Dostoevsky* than *Hemingway*.

Figure 5 shows that our logistic regression model (LR) outperforms the Berkeley system on numerous metrics when trained and evaluated on the quiz bowl dataset. We use precision, recall, and F_1, metrics applied to MUC, BCUB, and CEAFE measures used for comparing coreference systems.[13] We find that our LR model outperforms Berkeley by a wide margin when both are trained on the mentions found by our mention detector (CRF). For four metrics, the CRF mentions actually improve over training on the gold mentions.

Why does the LR model outperform Berkeley

[12]These features were obtained using the Stanford dependency parser (De Marneffe et al., 2006).

[13]The MUC (Vilain et al., 1995) score is the minimum number of links between mentions to be inserted or deleted when mapping the output to a gold standard key set. BCUB (Bagga and Baldwin, 1998) computes the precision and recall for all mentions separately and then combines them to get the final precision and recall of the output. CEAFE (Luo, 2005) is an improvement on BCUB and does not use entities multiple times to compute scores.

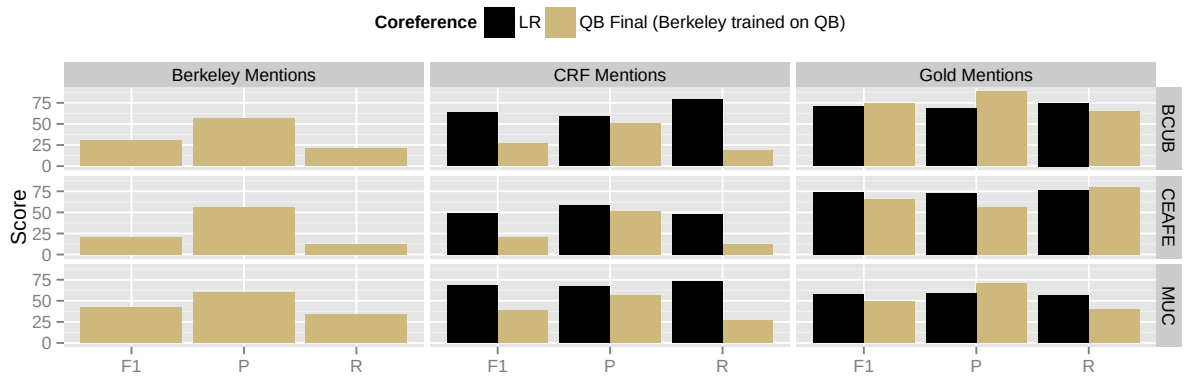

Figure 5: All models are trained and evaluated on quiz bowl data via five fold cross validation on F_1, precision, and recall. Berkeley/CRF/Gold refers to the mention detection used, LR refers to our logistic regression model and *QB Final* refers to the Berkeley model trained on quiz bowl data. Our model outperforms the Berkeley model on every metric when using our detected CRF mentions. When given gold mentions, LR outperforms Berkeley *QB Final* in five of nine metrics.

when both are trained on our quiz bowl dataset? We hypothesize that some of Berkeley's features, while helpful for sparse OntoNotes coreferences, do not offer the same utility in the denser quiz bowl domain. Compared to newswire text, our dataset contains a much larger percentage of complex coreference types that require world knowledge to resolve. Since the Berkeley system lacks semantic features, it is unlikely to correctly resolve these instances, whereas the pretrained word embedding features give our LR model a better chance of handling them correctly. Another difference between the two models is that the Berkeley system ranks mentions as opposed to doing pairwise classification like our LR model, and the mention ranking features may be optimized for newswire text.

5.4 Why Quiz Bowl Coreference is Challenging

While models trained on newswire falter on these data, is this simply a domain adaptation issue or something deeper? In the rest of this section, we examine specific examples to understand why quiz bowl coreference is so difficult. We begin with examples that *Final* gets wrong.

> *This writer* depicted a group of samurai's battle against an imperial. For ten points, name *this Japanese writer of A Personal Matter and The Silent Cry.*

While *Final* identifies most of pronouns associated with Kenzaburo Oe (the answer), it cannot recognize that the theme of the entire paragraph is building to the final reference, "this Japanese writer", despite the many Japanese-related ideas in the text of the question (e.g., Samurai and emperor). *Final* also cannot reason effectively about coreferences that are tied together by similar modifiers as in the below example:

> That *title character* plots to secure a "beautiful death" for Lovberg by burning his manuscript and giving him a pistol. For 10 points, name this play in which *the titular wife of George Tesman* commits suicide.

While a reader can connect "titular" and "title" to the same character, Hedda Gabler, the Berkeley system fails to make this inference. These data are a challenge for all systems, as they require extensive world knowledge. For example, in the following sentence, a model must know that the story referenced in the first sentence is about a dragon and that dragons can fly.

> The protagonist of *one of this man's works* erects a sign claiming that that story's title figure will fly to heaven from a pond. Identify this author of *Dragon: the Old Potter's Tale*

Humans solve cases like these using a vast amount of external knowledge, but existing models lack information about worlds (both real and imaginary) and thus cannot confidently mark these coreferences. We discuss coreference work that incorporates external resources such as Wikipedia in the next section; our aim is to provide a dataset that benefits more from this type of information than newswire does.

6 Related Work

We describe relevant data-driven coreference research in this section, all of which train and evaluate on only newswire text. Despite efforts to build better rule-based (Luo et al., 2004) or hybrid statistical systems (Haghighi and Klein, 2010), data-driven systems currently dominate the field. The 2012 CoNLL shared task led to improved data-driven systems for coreference resolution that finally outperformed both the Stanford system (Lee et al., 2011) and the IMS system (Björkelund and Farkas, 2012), the latter of which was the best available publicly-available English coreference system at the time. The recently-released Berkeley coreference system (Durrett and Klein, 2013) is especially striking: it performs well with only a sparse set of carefully-chosen features. Semantic knowledge sources—especially WordNet (Miller, 1995) and Wikipedia—have been used in coreference engines (Ponzetto and Strube, 2006). A system by Ratinov and Roth (2012) demonstrates good performance by using Wikipedia knowledge to strengthen a multi-pass rule based system. In a more recent work, Durrett and Klein (2014) outperform previous systems by building a joint model that matches mentions to Wikipedia entities while doing named entity resolution and coreference resolution simultaneously. We take a different approach by approximating semantic and world knowledge through our word embedding features. Our simple classifier yields a binary decision for each mention pair, a method that had been very popular before the last five years (Soon et al., 2001; Bengtson and Roth, 2008; Stoyanov et al., 2010). Recently, better results have been obtained with mention-ranking systems (Luo et al., 2004; Haghighi and Klein, 2010; Durrett and Klein, 2013; Björkelund and Kuhn, 2014). However, on quiz bowl data, our experiments show that binary classifiers can outperform mention-ranking approaches.

7 Embracing Harder Coreference

This paper introduces a new, naturally-occuring coreference dataset that is easy to annotate but difficult for computers to solve. We show that active learning allows us to create a dataset that is rich in different types of coreference. We develop an end-to-end coreference system using very simple mention detection and pairwise classification models that outperforms traditional systems on our dataset. The next challenge is to incorporate the necessary world knowledge to solve these harder coreference problems. Systems should be able to distinguish who is likely to marry whom, identify the titles of books from roundabout descriptions, and intuit family relationships from raw text. These are coreference challenges not found in newswire but that do exist in the real world. Unlike other AI-complete problems like machine translation, coreference in challenging datasets is easy to both annotate and evaluate. This paper provides the necessary building blocks to create and evaluate those systems.

8 Acknowledgments

We thank the anonymous reviewers for their insightful comments. We also thank Dr. Hal Daumé III and the members of the "feetthinking" research group for their advice and assistance. We also thank Dr. Yuening Hu and Mossaab Bagdouri for their help in reviewing the draft of this paper. This work was supported by NSF Grant IIS-1320538. Boyd-Graber is also supported by NSF Grants CCF-1018625 and NCSE-1422492. Any opinions, findings, results, or recommendations expressed here are of the authors and do not necessarily reflect the view of the sponsor.

References

Amit Bagga and Breck Baldwin. 1998. Algorithms for scoring coreference chains. In *International Language Resources and Evaluation*. Citeseer.

Eric Bengtson and Dan Roth. 2008. Understanding the value of features for coreference resolution. In *Proceedings of Emperical Methods in Natural Language Processing*. Association for Computational Linguistics.

Anders Björkelund and Richárd Farkas. 2012. Data-driven multilingual coreference resolution using resolver stacking. In *Conference on Computational Natural Language Learning*.

Anders Björkelund and Jonas Kuhn. 2014. Learning structured perceptrons for coreference resolution with latent antecedents and non-local features. In *Proceedings of the Association for Computational Linguistics*.

A. Boyd, P. Stewart, and R. Alexander. 2008. *Broadcast Journalism: Techniques of Radio and Television News*. Taylor & Francis.

Jordan Boyd-Graber, Brianna Satinoff, He He, and Hal Daume III. 2012. Besting the quiz master: Crowdsourcing incremental classification games. In *Proceedings of Emperical Methods in Natural Language Processing*.

Marie-Catherine De Marneffe, Bill MacCartney, Christopher D Manning, et al. 2006. Generating typed dependency parses from phrase structure parses. In *International Language Resources and Evaluation*.

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 *International Language Resources and Evaluation*.

Greg Durrett and Dan Klein. 2013. Easy victories and uphill battles in coreference resolution. In *Proceedings of Emperical Methods in Natural Language Processing*.

Greg Durrett and Dan Klein. 2014. A joint model for entity analysis: Coreference, typing, and linking. *Transactions of the Association for Computational Linguistics*.

Eraldo Rezende Fernandes, Cícero Nogueira Dos Santos, and Ruy Luiz Milidiú. 2012. Latent structure perceptron with feature induction for unrestricted coreference resolution. In *Proceedings of Emperical Methods in Natural Language Processing*.

N. Goldstein and A. Press. 2004. *The Associated Press Stylebook and Briefing on Media Law*. Associated Press Stylebook and Briefing on Media Law. Basic Books.

Aria Haghighi and Dan Klein. 2010. Coreference resolution in a modular, entity-centered model. In *Conference of the North American Chapter of the Association for Computational Linguistics*.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. Ontonotes: the 90% solution. In *Conference of the North American Chapter of the Association for Computational Linguistics*.

Mohit Iyyer, Jordan Boyd-Graber, Leonardo Claudino, Richard Socher, and Hal Daumé III. 2014. A neural network for factoid question answering over paragraphs. In *Proceedings of Emperical Methods in Natural Language Processing*.

Jonathan K Kummerfeld, Mohit Bansal, David Burkett, and Dan Klein. 2011. Mention detection: heuristics for the ontonotes annotations. In *Conference on Computational Natural Language Learning*.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Florian Laws, Florian Heimerl, and Hinrich Schütze. 2012. Active learning for coreference resolution. In *Conference of the North American Chapter of the Association for Computational Linguistics*.

Heeyoung Lee, Yves Peirsman, Angel Chang, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. 2011. Stanford's multi-pass sieve coreference resolution system at the conll-2011 shared task. In *Conference on Computational Natural Language Learning*.

Xiaoqiang Luo, Abe Ittycheriah, Hongyan Jing, Nanda Kambhatla, and Salim Roukos. 2004. A mention-synchronous coreference resolution algorithm based on the bell tree. In *Proceedings of the Association for Computational Linguistics*.

Xiaoqiang Luo. 2005. On coreference resolution performance metrics. In *Proceedings of Emperical Methods in Natural Language Processing*.

Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Timothy A Miller, Dmitriy Dligach, and Guergana K Savova. 2012. Active learning for coreference resolution. In *Proceedings of the 2012 Workshop on*

Biomedical Natural Language Processing. Proceedings of the Association for Computational Linguistics.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

MUC-6. 1995. Coreference task definition (v2.3, 8 sep 95). In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*.

MUC-7. 1997. Coreference task definition (v3.0, 13 jun 97). In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*.

Vincent Ng and Claire Cardie. 2002. Improving machine learning approaches to coreference resolution. In *Proceedings of the Association for Computational Linguistics*.

Vincent Ng. 2010. Supervised noun phrase coreference research: The first fifteen years. In *Proceedings of the Association for Computational Linguistics*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of Emperical Methods in Natural Language Processing*.

Thierry Poibeau and Leila Kosseim. 2001. Proper name extraction from non-journalistic texts. *Language and computers*, 37(1):144–157.

Simone Paolo Ponzetto and Michael Strube. 2006. Exploiting semantic role labeling, WordNet and Wikipedia for coreference resolution. In *Conference of the North American Chapter of the Association for Computational Linguistics*.

Sameer S Pradhan, Eduard Hovy, Mitch Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2007. Ontonotes: A unified relational semantic representation. *International Journal of Semantic Computing*, 1(04).

Sameer Pradhan, Lance Ramshaw, Mitchell Marcus, Martha Palmer, Ralph Weischedel, and Nianwen Xue. 2011. CoNLL-2011 shared task: Modeling unrestricted coreference in Ontonotes. In *Conference on Computational Natural Language Learning*.

Lev Ratinov and Dan Roth. 2009. Design challenges and misconceptions in named entity recognition. In *Conference on Computational Natural Language Learning*.

Lev Ratinov and Dan Roth. 2012. Learning-based multi-sieve co-reference resolution with knowledge. In *Proceedings of Emperical Methods in Natural Language Processing*.

Burr Settles. 2010. Active learning literature survey. *University of Wisconsin, Madison*, 52:55–66.

Wee Meng Soon, Hwee Tou Ng, and Daniel Chung Yong Lim. 2001. A machine learning approach to coreference resolution of noun phrases. *Computational linguistics*, 27(4).

Veselin Stoyanov, Claire Cardie, Nathan Gilbert, Ellen Riloff, David Buttler, and David Hysom. 2010. Coreference resolution with reconcile. In *Proceedings of the Association for Computational Linguistics*.

Olga Uryupina. 2006. Coreference resolution with and without linguistic knowledge. In *International Language Resources and Evaluation*.

Yannick Versley, Simone Paolo Ponzetto, Massimo Poesio, Vladimir Eidelman, Alan Jern, Jason Smith, Xiaofeng Yang, and Alessandro Moschitti. 2008. Bart: A modular toolkit for coreference resolution. In *Proceedings of the Association for Computational Linguistics*.

Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. 1995. A model-theoretic coreference scoring scheme. In *Proceedings of the conference on Message understanding*, pages 45–52.