Event Coreference Resolution Using Neural Network Classifiers

Arun Pandian
Carnegie Mellon University in Qatar
Doha, Qatar
apandian@andrew.cmu.edu

Lamana Mulaffer
Carnegie Mellon University in Qatar
Doha, Qatar
fmulaffe@andrew.cmu.edu

Kemal Oflazer
Carnegie Mellon University in Qatar
Doha, Qatar
ko@cs.cmu.edu

Amna AlZeyara
Carnegie Mellon University in Qatar
Doha, Qatar
azeyara@andrew.cmu.edu

Abstract

This paper presents a neural network classifier approach to detecting both within- and cross-document event coreference effectively using only event mention based features. Our approach does not (yet) rely on any event argument features such as semantic roles or spatiotemporal arguments. Experimental results on the ECB+ dataset show that our approach produces F_1 scores that significantly outperform the state-of-the-art methods for both within-document and cross-document event coreference resolution when we use B^3 and $CEAF_e$ evaluation measures, but gets worse F_1 score with the MUC measure. However, when we use the $CoNLL$ measure, which is the average of these three scores, our approach has slightly better F_1 for within-document event coreference resolution but is significantly better for cross-document event coreference resolution.

1 Introduction

Event coreference resolution is the task of identifying spans of text that refer to unique events and clustering or chaining them, resulting in one cluster/chain per unique event. This is an important part of an NLP system that performs topic detection (Allan et al., 1998), information extraction, question answering (Narayanan and Harabagiu, 2004), text summarization (Azzam et al., 1999) or any other system that is predicated on understanding natural language. Unlike entity coreference, which refers to clustering of nouns or pronouns that refer to the same entity, event coreference is a fundamentally harder problem. This is because the event as a semantic unit is structurally more complex to identify and resolve coreference for, as it has *event arguments* such as participants and spatio-temporal information that could be distributed across the text (Bejan and Harabagiu, 2010). Furthermore, different event mentions can refer to the same real world event and thus the context of the mentions and their arguments may also need to be considered (Yang et al., 2015). For instance, in the sentences

Lindsay Lohan ***checked into*** *New Promises Rehabilitation Facility on Sunday morning.*

and

News of this ***development*** *caused the media to line up outside the facility.*

"checked into" and "development" are coreferent. However, devoid of context, "checked into" and "development" are not semantically related. Additionally, events arguments (e.g., Lindsay Lohan, New Promises Rehabilitation Facility, Sunday morning) don't appear in the second sentence.

There are two types of event coreference resolution, within-document (WD) and cross-document (CD) event resolution. WD resolution is typically easier to solve as there is a higher chance of true coreference if there is similarity in the words used, contexts and event arguments. On the other hand, more evidence is needed to resolve coreference across documents as different documents are less likely to talk about the same events in the same way. Therefore, it is common practice to first solve WD coreference and then later use this information to solve CD coreference (Choubey and Huang, 2017). In this work, we try to solve both within-document and cross-document coreference without identifying event arguments and their semantic roles in the event as these are still difficult to extract with high accuracy (Bejan and Harabagiu, 2014).

We build two feed-forward neural nets for pairwise event coreference prediction, one for WD coreference and one for CD coreference that output the probability of coreference of the pair of events in question. Both classifiers are trained separately since we expect that the importance for the features for CD and WD coreference to differ (Choubey and Huang, 2017).

Once all pairwise event coreference predictions are complete, we construct a graph where each node is an event mention and each edge weight is the probability produced by the classifier representing a potential coreference relation between the nodes. We then find all the connected components (equivalent to finding the coreference clusters) in the graph after edges are filtered by a predetermined threshold. In the case of CD coreference, we find connected components in two phases - first by finding all the WD connected components and second by merging these WD components to build CD components.

Experimental results on ECB+ dataset show that our F_1 scores significantly outperform the state-of-the-art methods for both WD and CD event coreference resolution when we use B^3 (Bagga and Baldwin, 1998) and $CEAF_e$ (Luo, 2005) evaluation measures, but gets worse F_1 score with the MUC measure (Vilain et al., 1995). However, when we use the $CoNLL$ measure which is the average of these three scores (Pradhan et al., 2014), our approach has slightly better F_1 for WD resolution but is significantly better for CD resolution.

2 Related work

Different approaches, focusing on either WD or CD coreference chains, have been proposed for event coreference resolution. Works specific to WD event coreference include pairwise classifiers (Ahn, 2006; Chen et al., 2009), graph-based clustering (Chen and Ji, 2009) and information propagation (Liu et al., 2014). Works focusing purely on CD coreference include Cybulska and Vossen (2015b) who created pairwise classifiers using features indicating granularities of event slots, and in another work (Cybulska and Vossen, 2015a), use discourse analysis at the document level along with 'sentence' templates amongst documents that have possibly coreferent events. Several papers have studied event extraction and event coreference as a joint process (Araki and Mitamura, 2015; Lu and Ng, 2017).

Several studies have considered both WD and CD event coreference resolution tasks simultaneously. Such approaches (Lee et al., 2012; Bejan and Harabagiu, 2010; Bejan and Harabagiu, 2014) create a meta-document by concatenating topic-relevant documents and treat both WD and CD coreference resolution as identical tasks. Most recently, Yang et al. (2015) applied a two-level clustering model that first groups event mentions within a document and then groups WD clusters across documents in a joint inference process. Choubey et al. (2017) used an event coreference model that uses both pairwise CD and WD classifiers to build event clusters iteratively by switching between WD and CD coreference until the results converge. While our approach also uses the two pairwise classifiers, in our CD resolution model, we simply build WD components (clusters) followed by CD components (clusters) in one iteration.

3 Detecting pairwise event coreference

Both WD and CD event coreference resolution are implemented using feedforward neural nets with ReLU units that take two featurized events, their contexts and assorted semantic information, and determine if the events are coreferent. The WD NN is a single hidden layer neural net with 300 hidden units, while the CD NN has two hidden layers with 400 and 150 hidden units for layer 1 and layer 2 respectively. The size of the NNs were determined by experiments documented in Section 3.4. During each feedforward back-propagation cycle, the NN aims to minimize the negative log cross entropy of the softmax distribution. The final output layer of both NNs is a softmax that gives the probability of coreference between the pair of events.

3.1 Event features

We use two types of features for characterizing event mentions: *contextual features* and *relational features*. Contextual features are extracted from each sentence independently and relational features depend

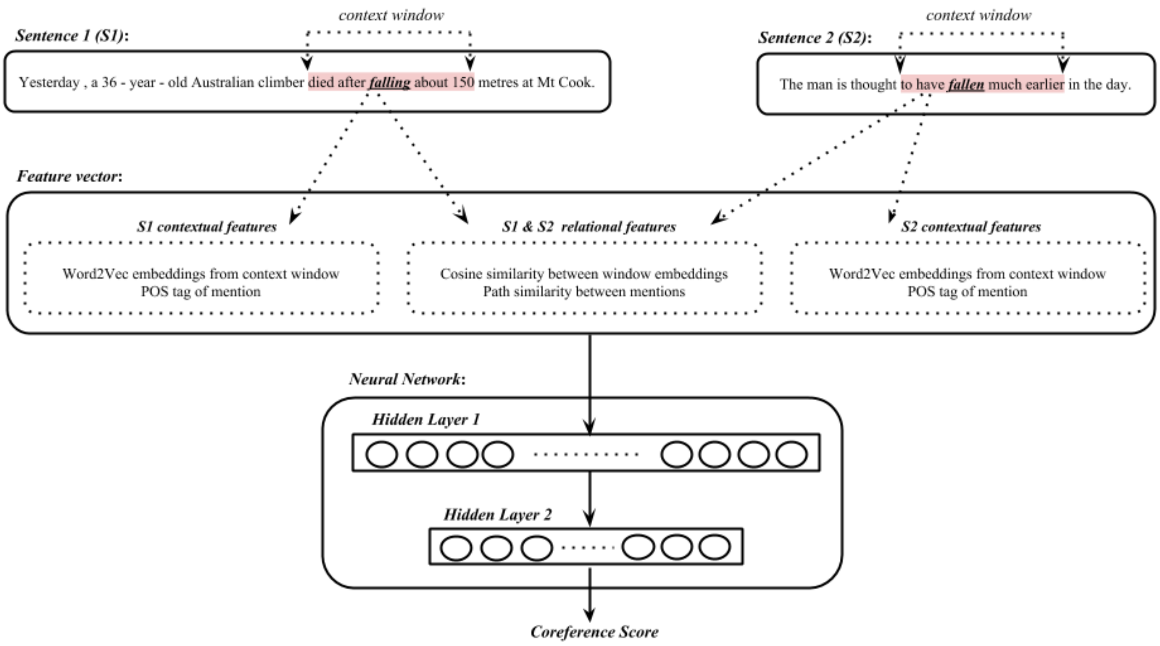

Figure 1: Feed-forward Neural Network structure for CD Resolution

on the relationships between the two sentences.

The contextual features are the main event word, the POS tag of the event word, and words in a pre-defined window around the event word. In representing the words in a sentence, we use word embeddings of size 400 from a word2vec model (Mikolov et al., 2013) that we built using the gensim implementation from the English Wikipedia corpus (Řehůřek and Sojka, 2010). The embeddings we included are that of the event mention as well as the two words on each side that appear in the word2vec model.[1] We generated the POS tags for the event mentions using the Natural Language ToolKit (NLTK) package (Bird et al., 2009) and represented them as one-hot vectors.

The relational features include the distance between the two sentences (for WD classification only), the cosine similarity of the embeddings of the event words and the WordNet similarities using variations of the main event words. The cosine similarity is calculated for the embedding of the event words and the value is quantized into 11 buckets (including one bucket for unknown similarities) and represented as one-hot vectors. As for the WordNet similarities, we consider three values using variations of the main event words. All the path similarities, senses and derivation related to WordNet are generated through TextBlob (Loria et al., 2014). We calculate the first value by considering all the senses of both event words. Different path similarities are calculated using two word combinations; one word from each set of senses. Ultimately, the maximum similarity is selected. We calculate the second value by using the path similarity between the hypernyms of the event words. Lastly, we calculate the third value by considering the path similarity between derivationally related verb forms of each event word provided by WordNet. Similar to the cosine similarity, all three values are quantized and represented as a one-hot vector. Figure 1 shows an overview of the CD resolution model, which takes as input a pair of sentences, featurizes them into neural network inputs comprising of contextual and relational features, passes them through the two layer NN and produces a coreference score, which is the probability of coreference between the two event mentions in the input sentences.

During the preliminary stage of feature selection, we started with only contextual features; the event word embeddings, context embeddings and POS tags. Then we tested with only relational features; distance between the sentences, word embeddings, cosine similarity and WordNet path similarity of the

[1]For multi-word events, we determined the head word using the approach by Honnibal and Johnson (2014) and used its embedding.

main event words. Finally, we tested with using both, contextual and relational features. The accuracy of using both contextual and relational features was higher than when using them separately as expected, but using either feature set individually gave similar results on the development set.

When we performed an error analysis, we discovered that despite an observable semantic relation between words, the WordNet similarity was low (especially for words that had the same hypernym). Therefore, we quantized the WordNet path similarity between the hypernyms and added it as a features as a one hot vector. Additionally, we found that the WordNet path similarity differentiated greatly between words of different syntactic categories. Therefore, we also generated the derivationally related verb form of each event word, and added the path similarity between them. Adding WordNet path similarity of hypernyms and the related verb form of each word greatly improved performance. Results further improved when non-coreferent pairs where sampled purely from all pairs whose mentions were either from the same sentence or from sentences that shared a coreference pair.

3.2 Changes to classifier from earlier work

In comparison to the latest system that uses pairwise classifiers (Choubey and Huang, 2017) which builds on (Chen et al., 2009; Ahn, 2006), our pairwise classifiers have some significant differences. We use precomputed word embeddings, as opposed to computing word embeddings during classifier training. We also do not featurize event arguments as part of our pairwise classifier (as mentioned above), as extracting event arguments and their relation to an event is difficult to identify accurately (Bejan and Harabagiu, 2010). Furthermore we use contextual features for both WD and CD coreference, while Choubey et al.,(2017) use contextual features only for CD coreference. The aforementioned system uses only cosine similarity and euclidean distance between their computed embeddings for their relational features, while we extract other relational features explained in 3.1, including WD path similarities.

3.3 Training the pairwise coreference classifiers

We train the pairwise classifiers using documents from topics 1-23 of the ECB+ corpus (Yang et al., 2015; Choubey and Huang, 2017). The statistics of the corpus are provided in Table 1. We extract the training data clusters, and generate all the coreferent event mention pairs from them. To ensure the pairwise classifiers become proficient at identifying non-coreferent event mentions even in similar contexts, we only include those non-coreferent pairs whose event mentions either belong to the same sentence or belong to sentences that also share a coreferent pair. We then sample from these non-coreferent pairs to ensure the number of coreferent and non-coreferent training pairs are the same. Although the number of non-coreferent pairs outweigh the number of coreferent pairs, we train our WD model using a 50-50 training split, that is equal number of coreferent and non-coreferent pairs, to avoid developing a bias for some statistical distribution of these pairs within the ECB+ corpus. However, we train our CD model using the actual training split, in order to provide more training samples as CD resolution is inherently a more difficult problem to solve than WD resolution. We train WD and CD pairwise classifiers separately since we expect the importance of neural net learned weights to differ between for the two cases (Choubey and Huang, 2017).

3.4 Intrinsic Evaluation

To ensure successful clustering, it was paramount to have a strong pairwise classifier. Therefore we performed intrinsic evaluations for this classifier. For this purpose, we use documents from topics (23-25) as the development set and topics (26-45) as the test set (Yang et al., 2015; Choubey and Huang, 2017). The development set was used to tune the parameters of the classifier as well as to find the probability threshold of the output that determines coreference.

As the ECB+ corpus is incompletely annotated in both event mentions and event coreference (Cybulska and Vossen, 2014a), running an event detection tool will not be necessarily instructive as some coreferences between events and actual events themselves are left unmarked in the database. Therefore we extract gold standard event mentions. Regardless, to be able to compare our results to previously published results (Yang et al., 2015; Choubey and Huang, 2017), we perform event detection using the same event detection tool used by these systems on the same test set. The event detection tool used is

	Train	**Dev**	**Test**	**Total**
#Documents	462	73	447	982
#Sentences	7,294	649	7,867	15,810
#Event Mentions	3,555	441	3,290	7,286
#CD Chains	687	47	586	1,220
#WD Chains	2,499	316	2,137	4,952
Avg. WD chain length	2.84	2.59	2.55	2.69
Avg. CD chain length	5.17	9.39	6.77	5.98

Table 1: ECB+ corpus statistics

	Coreference Threshold	#Coref Links	#Non-coref Links	TP	TN	P	R	F_1	Accuracy
WD-gold	0.5	1,799	1,799	1,302	1,565	84.77	72.37	78.08	79.68
WD-gold	0.95	1,799	1,799	1,097	1,697	91.49	60.98	73.18	77.65
CD-gold	0.5	24,315	24,315	16,968	21,124	84.17	69.78	76.30	78.33
CD-gold	1	24,315	24,315	9,817	23,689	94.01	40.37	56.48	68.90

Table 2: Intrinsic evaluation for pairwise classifier with balanced test sets

a CRF-based semi-Markov model that is trained using sentences from ECB+ to provide more accurate detection of events (Yang et al., 2015). Once we extract the event mentions, we only use those mentions also found in the gold standard as the ECB+ corpus is incompletely marked and it's not possible to determine if pairs generated from non-gold event mentions are actually non-coreferent.

Once the mentions are finalized, for both detected event mentions and gold standard event mentions, all possible coreferent pairs are generated. For the WD case, the non-coreferent pairs are all pairs within a document that are not marked coreferent, while for the CD case, they comprise all pairs within a sub-topic that are not marked coreferent.

As mentioned above, for the first set of experiments we sample non-coreference pairs equal to the number of coreference pairs and evaluate our pairwise models on them. For the second set of experiments we used all the non-coreferent pairs without any sampling. We then used our development set to determine the best coreference thresholds for event clustering. We realized that for the purpose of finding event clusters it was important to have high precision for the event coreference. The best coreference thresholds were 0.95 for WD coreference and 1.0 for CD coreference. Therefore we also reports results on the these numbers for both the balanced and unbalanced test sets. For the unbalanced test set, we report the results on both types of event mention pairs, one generated from the gold standard event mentions (called WD-gold and CD-gold) and one from the extracted event mentions (WD-detect and CD-detect).

The results of experiments on the balanced test set are reported in Table 2 and of the whole test set in Table 3. We see that the precision improves when the threshold increases in all cases. We also report the accuracy which represents the number of pairs classified correctly by our classifier. Additionally, the histogram in Figure 2 shows the coreference score distribution from the WD resolution model for coreferent and non-coreferent pairs of mentions. The CD resolution model has a similar coreference score distribution. As we can see in both Table 3 and Figure 2, our system is able to accurately detect and differentiate between coreferent pairs quite competently.

4 Clustering Events

From our pairwise classifier, our next step is to build clusters of event mentions such that all mentions in a cluster are considered coreferent to each other. In order to build such coreferent clusters, we model

	Coreference Threshold	#Coref Links	#Non-coref Links	TP	TN	P	R	F_1	Accuracy (Non-coref)	Accuracy (All)
WD-gold	0.95	1,799	12,701	1,097	11,984	60.47	60.98	60.72	94.35	90.21
WD-detect	0.95	1,212	11,735	762	11,353	66.61	62.87	64.69	96.74	93.57
CD-gold	1.0	24,315	144,515	9,669	143,002	86.47	39.77	54.48	98.95	90.43
CD-detect	1.0	16,329	133,991	7,110	133,048	88.29	43.54	58.32	99.3	93.24

Table 3: Intrinsic evaluation for pairwise classifier with actual (unbalanced) test sets

our problem as finding connected components in a weighted graph. We represent event mentions as nodes and coreference scores of event mention pairs (given by the pairwise classifier) as weights of the edges between those mention pairs. In the case of WD resolution, an edge between a pair of mentions (nodes) exists if they belong to the same document. In the case of CD resolution, an edge between a pair of mentions (nodes) exists if they belong to the same topic, since we know that there are no CD coreferences between event mentions from different topics.

For WD resolution, we filter out all edges with weights less than 0.95 and find all the connected components in the graph. These components are the WD coreferent clusters which we later evaluate against the WD gold standard clusters. For CD resolution, we first perform WD resolution on all WD edges. We then build CD components if there is a CD edge with a threshold of 1.0 between the WD components. These components are the CD coreferent clusters which we later evaluate against the CD gold standard clusters.

5 Evaluation

We perform all our experiments on the ECB+ news corpus (Cybulska and Vossen, 2014b). As described in 1, our test set consists of documents from topics 26-45.

We evaluated our system using three widely used coreference resolution metrics: MUC, B^3 and $CEAF_e$, computed using the most recent version (v8.01) of the official CoNLL Scorer (Pradhan et al., 2014). MUC (Vilain et al., 1995) measures how many gold (predicted) cluster merging operations are needed to recover each predicted (gold) cluster. B^3 (Bagga and Baldwin, 1998) measures the proportion of overlap between the predicted and gold clusters for each mention and computes the average scores. $CEAF_e$ (Luo, 2005) measures the best alignment of the gold-standard and predicted clusters. We also calculate the $CoNLL\ F_1$ score, which is the average of the F_1 scores across all three evaluation metrics.

5.1 Baseline Systems

We compare our system using three baseline systems.

1. LEMMA: This baseline groups event mentions into clusters based on their lemmatized head words. A cluster is formed if all its event mentions share the same lemmatized head word. This is often considered a strong baseline (Yang et al., 2015).

2. HDDCRP: This baseline is produced by the supervised Hierarchical Distance Dependent Chinese Restaurant Process model (Yang et al., 2015). The model clusters event mentions based on their relative distances, given by a learnable distance function. This is the most recent coreference system to be evaluated on the ECB+ corpus.

3. Iterative WD/CD Classifier: This baseline is produced by the iterative event coreference model that gradually builds event clusters by exploiting inter-dependencies within both WD and CD mentions until the clusters converge (Choubey and Huang, 2017).

5.2 Our System

We evaluate our system with two sets of slightly different testing data. The first set uses the gold standard event mentions marked in the ECB+ corpus (WD-gold and CD-gold); the second set uses events marked using the aforementioned event detection tool (WD-detect and CD-detect). We can also compare the results from the latter set to the last two baseline systems mentioned above.

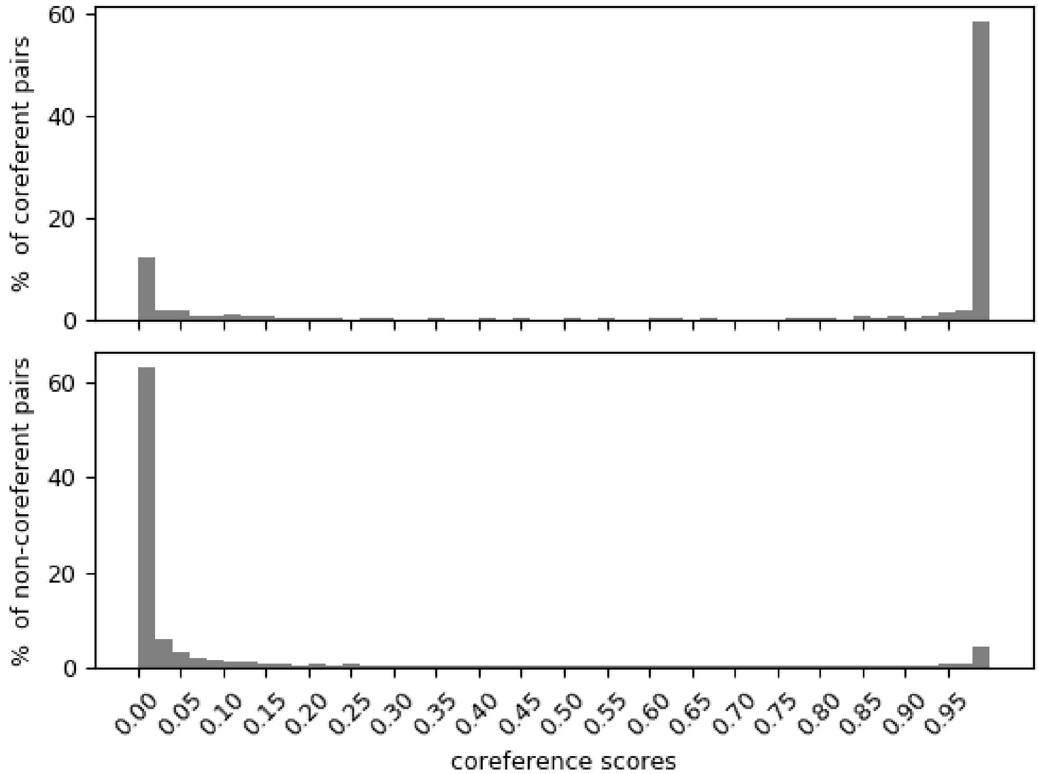

Figure 2: Coreference score distribution from WD-gold model. Scores of coreferent pairs (top) and scores of non-coreferent pairs (bottom)

5.3 Results

Table 4 shows the results obtained for WD coreference while Table 5 shows the results obtained for CD coreference. We notice that in the case of WD coreference, our system overall performs slightly better than the state-of-the-art (Choubey and Huang, 2017) for the $CoNLL\ F_1$ score. However for CD coreference our system performs more than 8% points better than the state-of-the-art in the same measure. Additionally our WD and CD systems perform better than the Lemma baseline for all measures except for MUC. There are instances where our WD and CD systems perform better than the WD-gold and CD-gold, this could be attributed to the fact that we select all predicted mentions as long as they are part of the gold standard, a limitation that is needed because of the incompletely marked ECB+ corpus. We also notice that our WD-gold and CD-gold systems perform, as expected, significantly better than the WD-detect and CD-detect systems. These numbers provide an indication of how our system performs with a fully annotated ECB+ corpus and an event detection tool that has 100% recall.

6 Discussion

6.1 Error Analysis

Since we construct coreferent clusters of event mentions in an agglomerative way, the primary disadvantage we face is error propagation. To mitigate this, we employ high thresholds to determine if two clusters (components) can be merged. However, there still are coreferent event pairs that are not considered coreferent and non-coreferent event pairs that are considered coreferent. We have performed an analysis of our system's final predictions on the development set to identify why we get these errors.

	WD Model									
	MUC			B^3			$CEAF_e$			$CoNLL$
	R	P	F_1	R	P	F_1	R	P	F_1	F_1
Baseline 1: Lemma, Yang et al., 2015	56.80	**80.90**	**66.70**	35.90	76.20	48.80	67.40	62.90	65.10	60.20
Baseline 2: Yang et al., 2015	41.70	74.30	53.40	67.30	85.60	75.40	**79.80**	65.10	71.70	66.83
Baseline 3: Choubey et al., 2017	**58.50**	67.30	62.60	**69.20**	76.00	72.40	67.90	76.10	71.80	68.93
WD-detect (this paper)	48.16	72.75	57.95	66.10	**90.53**	**76.41**	68.38	**80.02**	**73.74**	**69.37**
WD-gold (this paper)	67.68	71.57	69.57	85.54	87.99	86.75	80.63	79.70	80.16	78.83

Table 4: WD Coreference Results

	CD Model									
	MUC			B^3			$CEAF_e$			$CoNLL$
	R	P	F_1	R	P	F_1	R	P	F_1	F_1
Baseline 1: Lemma, Yang et al.,	39.50	73.90	51.40	58.10	78.20	66.70	58.90	36.50	46.20	54.80
Baseline 2: Yang et al., 2015	67.10	80.30	73.10	40.60	78.50	53.50	68.90	38.60	49.50	58.70
Baseline 3: Choubey et al., 2017	**67.50**	**80.40**	**73.40**	56.20	66.60	61.00	59.00	54.20	56.50	63.63
CD-detect (this paper)	53.94	62.09	57.73	**82.18**	**84.86**	**83.50**	**77.36**	**72.27**	**74.73**	**71.99**
CD-gold (this paper)	76.09	61.24	67.86	90.15	75.68	82.28	70.10	80.58	74.98	75.04

Table 5: CD Coreference Results

- **Missed coreference links:** This issue is common among event pairs that are not recognized by the word2vec model as being similar to each other, hence causing low relational similarity scores. For example, consider event pairs like (*raid*, *heist*), (*sprained*, *injury*), and (*nab*, *grabbing*). These pairs have low relational feature similarities causing the classifier to give low coreference scores. Another issue is that the current word2vec model is not able to successfully compare between multi-word events like *scooping up*, *making off*, *cleaning up* and *stay alive*. Since our model uses only event information in order to determine a coreference score, we also miss out on cases where event-entity coreference becomes crucial. Consider the following pair of sentences:

  > *Pierre Thomas was **placed** on injured reserve by the New Orleans Saints on Wednesday, meaning he won't play in the 2011 NFL playoffs.*

  and

  > *This means they will be missing two of their best players in the rushing game, and **it** could weaken the attack that the Saints have on offense even further.*

  In this case, having entity coreference information, and also event arguments and relations will help better resolve the coreference.

- **Incorrect coreference Links:** This issue is common among event pairs within the same sentence. Since our system uses contextual features, events in the same sentence share similar contextual features, resulting in our NNs giving high coreference scores to these event pairs. For example, consider the sentence paired with itself, where we are testing for a coreference within the same sentence.

  > *HP today announced that it has signed a definitive agreement to acquire EYP Mission Critical Facilities Inc., a consulting company specializing in strategic technology **planning**, design, and operations support for large-scale data centers.*

  and

  > *HP today announced that it has signed a definitive agreement to acquire EYP Mission Critical Facilities Inc., a consulting company specializing in strategic technology planning, **design**, and operations support for large-scale data centers.*

Here the contextual similarity between the two events *planning* and *design* are high, being very close to each other.

7 Conclusion and Future Work

Our results show that the pairwise models and the method with which we find coreferent clusters of event mentions is adept at identifying event coreference even without relying on event arguments. These results make manifest that accurate event detection significantly helps in improving event coreference resolution, as evidenced by the disparity between our system results when event mentions detected are gold standard and otherwise.

We would like to explore if event arguments and their relations to the event mention (whether semantic or syntactic) can be extracted in a reasonable way without error which would propagate upwards. We would also want to explore using the GloVe vector model embeddings (Pennington et al., 2014) in our NNs to obtain better word sense disambiguation, which would help in reducing the number of coreferent event pairs that are not recognized currently. We would also like to do joint entity and event coreference to improve the overall event coreference resolution. Additionally, we currently employ an even distribution of coreferent and non-coreferent event pairs for training purposes. However, since there is a much larger number of non-coreferent event pairs in general, we would also like to explore how our system performs with a higher number of non-coreferent event pairs for training.

References

[Ahn2006] David Ahn. 2006. The stages of event extraction. In *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*, pages 1–8. Association for Computational Linguistics.

[Allan et al.1998] James Allan, Jaime Carbonell, George Doddington, and Jonathan Yamron. 1998. Topic detection and tracking pilot study final report. In *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*.

[Araki and Mitamura2015] Jun Araki and Teruko Mitamura. 2015. Joint event trigger identification and event coreference resolution with structured perceptron. In *Proceedings of EMNLP*, pages 2074–2080.

[Azzam et al.1999] Saliha Azzam, Kevin Humphreys, and Robert Gaizauskas. 1999. Using coreference chains for text summarization. In *Proceedings of the Workshop on Coreference and its Applications*, pages 77–84. Association for Computational Linguistics.

[Bagga and Baldwin1998] Amit Bagga and Breck Baldwin. 1998. Algorithms for scoring coreference chains. In *The First LREC Workshop on Linguistic Coreference*, pages 563–566.

[Bejan and Harabagiu2010] Cosmin Adrian Bejan and Sanda Harabagiu. 2010. Unsupervised event coreference resolution with rich linguistic features. In *Proceedings of ACL*, pages 1412–1422. Association for Computational Linguistics.

[Bejan and Harabagiu2014] Cosmin Adrian Bejan and Sanda Harabagiu. 2014. Unsupervised event coreference resolution. *Computational Linguistics*, 40(2):311–347.

[Bird et al.2009] Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python — Analyzing Text with the Natural Language Toolkit*. O'Reilly Media.

[Chen and Ji2009] Zheng Chen and Heng Ji. 2009. Graph-based event coreference resolution. In *Proceedings of the 2009 Workshop on Graph-based Methods for Natural Language Processing*, pages 54–57. Association for Computational Linguistics.

[Chen et al.2009] Zheng Chen, Heng Ji, and Robert Haralick. 2009. A pairwise event coreference model, feature impact and evaluation for event coreference resolution. In *Proceedings of the Workshop on Events in Emerging Text Types*, pages 17–22. Association for Computational Linguistics.

[Choubey and Huang2017] Prafulla Kumar Choubey and Ruihong Huang. 2017. Event coreference resolution by iteratively unfolding inter-dependencies among events. In *Proceedings of EMNLP*, pages 2117–2123.

[Cybulska and Vossen2014a] Agata Cybulska and Piek Vossen. 2014a. Guidelines for ECB+ annotation of events and their coreference. Technical report, NWR-2014-1, VU University Amsterdam.

[Cybulska and Vossen2014b] Agata Cybulska and Piek Vossen. 2014b. Using a sledgehammer to crack a nut? Lexical diversity and event coreference resolution. In *LREC*, pages 4545–4552.

[Cybulska and Vossen2015a] Agata Cybulska and Piek Vossen. 2015a. "Bag of Events" approach to event coreference resolution. supervised classification of event templates. *International Journal of Computational Linguistics and Applications*, 6(2):11–27.

[Cybulska and Vossen2015b] Agata Cybulska and Piek Vossen. 2015b. Translating granularity of event slots into features for event coreference resolution. In *Proceedings of the The 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 1–10.

[Lee et al.2012] Heeyoung Lee, Marta Recasens, Angel Chang, Mihai Surdeanu, and Dan Jurafsky. 2012. Joint entity and event coreference resolution across documents. In *Proceedings of EMNLP-CONLL*, pages 489–500. Association for Computational Linguistics.

[Liu et al.2014] Zhengzhong Liu, Jun Araki, Eduard H Hovy, and Teruko Mitamura. 2014. Supervised within-document event coreference using information propagation. In *LREC*, pages 4539–4544.

[Loria et al.2014] Steven Loria, P Keen, M Honnibal, R Yankovsky, D Karesh, and E Dempsey. 2014. Textblob: simplified text processing. *Secondary TextBlob: Simplified Text Processing*.

[Lu and Ng2017] Jing Lu and Vincent Ng. 2017. Joint learning for event coreference resolution. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 90–101, Vancouver, Canada, July. Association for Computational Linguistics.

[Luo2005] Xiaoqiang Luo. 2005. On coreference resolution performance metrics. In *Proceedings of HLT-EMNLP*, pages 25–32. Association for Computational Linguistics.

[Mikolov et al.2013] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *Proceedings of Workshop at ICLR*.

[Narayanan and Harabagiu2004] Srini Narayanan and Sanda Harabagiu. 2004. Question answering based on semantic structures. In *Proceedings of COLING*, pages 693–701. International Committee on Computational Linguistics.

[Pennington et al.2014] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of EMNLP*, pages 1532–1543, Doha, Qatar, October. Association for Computational Linguistics.

[Pradhan et al.2014] Sameer Pradhan, Xiaoqiang Luo, Marta Recasens, Eduard Hovy, Vincent Ng, and Michael Strube. 2014. Scoring coreference partitions of predicted mentions: A reference implementation. In *Proceedings of ACL*, pages 30–35. Association for Computational Linguistics.

[Řehůřek and Sojka2010] Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May. ELRA.

[Vilain et al.1995] Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. 1995. A model-theoretic coreference scoring scheme. In *Proceedings of MUC*, pages 45–52.

[Yang et al.2015] Bishan Yang, Claire Cardie, and Peter Frazier. 2015. A hierarchical distance-dependent Bayesian model for event coreference resolution. *Transactions of the ACL*, 3:517–528.