Precision Event Coreference Resolution Using Neural Network Classifiers

Arun Pandian[1], Lamana Mulaffer[1], Kemal Oflazer[1], and Amna AlZeyara[2]

[1] Carnegie Mellon University in Qatar {apandian,lmulaffe,ko}@cmu.edu
[2] Qatar University amna.azr@gmail.com

Abstract. This paper presents a neural network classifier approach to detecting precise within-document(WD) and cross-document(CD) event coreference clusters effectively using only event mention based features. Our approach does not rely on any event argument features such as semantic roles or spatio-temporal arguments and uses no sophisticated clustering approach. Experimental results on the ECB+ dataset show that our simple approach outperforms state-of-the-art methods for both within-document and cross-document event coreference resolution while producing clusters of high precision, which is useful for several downstream tasks.

Keywords: Deep Learning · Event Coreference · Semantics.

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 [2], information extraction, question answering [25], text summarization [4] 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 [6]. 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 [30]. 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, some coreference systems solve WD coreference first and then later use this information to solve CD coreference [30, 11].

We try to solve both within-document and cross-document coreference without explicitly identifying event arguments and their semantic roles in the event as these are still difficult to extract with high accuracy [7]. We build two feed-forward neural nets for pairwise event coreference prediction with a small feature set, one for WD coreference and one for CD coreference that output the probability of coreference of the pair of events in question. We prioritize high precision over recall in the pairwise classifiers. Both CD and WD classifiers are trained separately since we expect that the importance for the features for CD and WD coreference to differ [11]. 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 using a (high) threshold value, determined through experiments with our development set to favor precise clustering. Precise clustering might be more useful in several downstream tasks like text summarization and question answering systems.

Experimental results on ECB+ dataset show that our simple system outperforms the state-of-the-art methods for both WD and CD event coreference resolution when we use the widely used CoNLL[26] measure which is the average of the CEAF-E [23], the MUC [29] and the B^3 [5] measures.

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 [1, 10], graph-based clustering [9] and information propagation [19]. Works focusing purely on CD coreference include Cybulska and Vossen [15] who created pairwise classifiers using features indicating granularities of event slots, and in another work [14], 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 [3, 21].

Several studies have considered both WD and CD event coreference resolution tasks simultaneously. Such approaches [17, 6, 7] create a meta-document by concatenating topic-relevant documents and treat both WD and CD coreference resolution as identical tasks. Both Yang et al. [30] and Choubey et al. [11] use the same ECB+ corpus as ourselves. Yang et al. [30] apply 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. [11] use an event coreference model that uses both pairwise CD and WD classifiers to build event clusters iteratively by switching between WD and CD coreference resolution, using additional information that is available about the clusters as they are being merged, until the results converge. Kenyon-Dean et al. [16], much recently build a general framework for clustering that uses supervised representation learning for the event mention embeddings using clustering oriented regularization terms. Although they also the ECB+ corpus, their results are not directly comparable to ours or the two published systems mentioned above, as they use a different criteria in selecting their testing data.

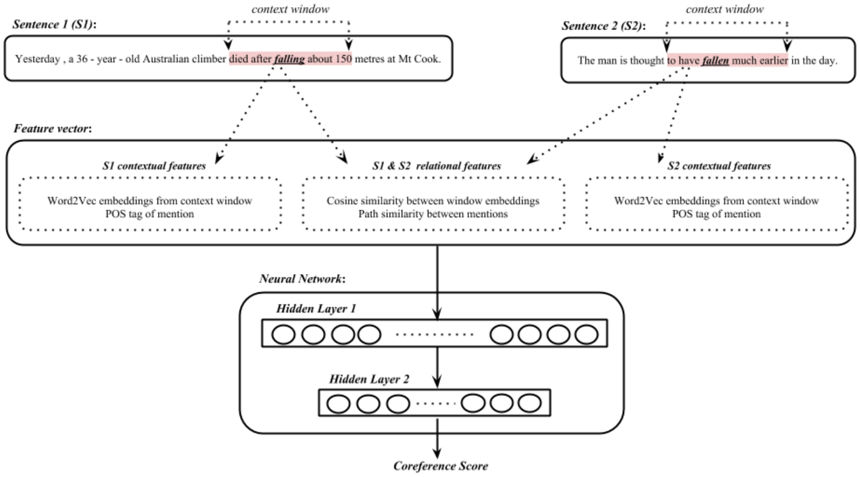
context window
Sentence 1 (S1):
Yesterday , a 36 - year - old Australian climber died after falling about 150 metres at Mt Cook.
context window
Sentence 2 (S2):
The man is thought to have fallen much earlier in the day.
Feature vector:
S1 contextual features
Word2Vec embeddings from context window
POS tag of mention
S1 & S2 relational features
Cosine similarity between window embeddings
Path similarity between mentions
S2 contextual features
Word2Vec embeddings from context window
POS tag of mention
Neural Network:
Hidden Layer 1
Hidden Layer 2
Coreference Score

Fig. 1: Feed-forward Neural Network structure for CD resolution

3 Detecting pairwise event coreference

The first step of both WD and CD event coreference resolution are implemented using feedforward neural nets with ReLU units that take two featurized events, their contexts, syntactic and semantic information, and determine if the events are coreferent. The basic architecture is as shown in 1. 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 number of hidden layers and units of the NNs were determined by experiments described in Section 3.4. The final output layer of both NNs is a softmax that gives the probability of coreference between the pair of events. Each feed-forward back-propagation cycle of the NN aims to minimize the negative log cross entropy of the softmax distribution. 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 features* 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.

3.1 Event features

We selected our features through extensive ablation studies and error analysis discussed in 3.4 and 5.4 to provide high precision for pairwise event coreference. We use 6 features of two types of features for characterizing event mentions: three *contextual features* and three *relational features*. Contextual features are extracted from each sentence independently and relational features depend on the relationships between the two sentences.

The contextual features are the embeddings of the event word(s), the POS tag of the event word, and embedding of words in a pre-defined window around the event word. We use pre-computed word embeddings of size 400 from a word2vec model [24] that we built using the gensim implementation from the English Wikipedia corpus [27]. 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. For event mentions with multiple words, we average the embeddings. We generated the POS tags for the event mentions using the Natural Language ToolKit (NLTK) package [8] and represented them as one-hot vectors.

The relational features are the the cosine similarity of the (average) embeddings of the event word(s), the cosine similarity of the lemmatized head word of the event[3] and the WordNet similarities using variations of the main event words. The cosine similarities are quantized into 11 buckets (including one bucket for unknown similarities) and represented as one-hot vectors.

The WordNet similarities we calculate are the maximum path similarity of all the senses of both event words, the maximum path similarity between hypernyms of both event words and the path similarity between derivationally related verb forms of both event words. The path similarities, senses and derivation related to WordNet are generated through TextBlob [20]. When we analysed WordNet features, 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 feature 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. The WordNet similarities are also quantized and represented as a one-hot vector.

3.2 Changes to classifier from earlier work

In comparison to the latest system that uses pairwise classifiers [11] which builds on [10, 1], our pairwise classifiers have some significant differences. A major difference between our work and previous work is that we try to build a classifier with high pairwise precision, which can then be used to generate accurate clusters. This affects the feature select we select, while keeping the pairwise model as simple as possible. We use pre-computed word embeddings, as opposed to computing word embeddings during

[3] We determined the head word using the approach by Honnibal and Johnson [20] and used its embedding.

	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	486	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

classifier training to maintain the simplicity of the model and also so that the embeddings for words not seen during training would not lose efficacy during test time as the trained embeddings might have moved in the N-dimensional space. We also do not explicitly identify 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 [6]. Furthermore we use contextual features for both WD and CD coreference, while Choubey et al.,[11] use contextual features only for CD coreference. Their 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 WordNet path similarities.

3.3 Training the pairwise coreference classifiers

We train the pairwise classifiers using documents from topics 1-23 of the ECB+ corpus [30, 11]. 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 generated from the training data, we train our WD model using a 50-50 training split, i.e. 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 [11]. To confirm this, we also used the WD classifier during CD classification and vice-versa, however the results obtained on the development set were much lower than that obtained when the neural nets were trained separately.

3.4 Intrinsic Evaluation

To ensure successful clustering, it was paramount to have a strong pairwise classifier(and in our case, one that produces precise pairwise results). Therefore we performed intrinsic(i.e. pairwise) evaluations for this classifier using documents from topics (23-25) as the development set and topics (26-45) as the test set [30, 11]. The counting of topic numbers vary slightly between this paper and previous work [30, 11] because we include topic numbers with no documents while they do not. Additionally, we have verified that the training, development and test splits are the same despite the difference in topic counting numbers, by ensuring that we obtain the same corpus statistics as them.

As the ECB+ corpus is incompletely annotated in both event mentions and event coreference [12], 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 [30, 11], 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 a CRF-based semi-Markov model that is trained using sentences from ECB+ to provide more accurate detection of events [30]. 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, we generate all possible coreferent pairs are generated for both detected event mentions and gold standard event mentions. 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.

We report the results on the final test set for 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 purpose of this intrinsic evaluation is two fold: to find a feature set for our pairwise models that prioritizes high clustering precision and to tune the hyperparameters of the classifier: size and number of hidden layers of the neural network along with a coreference threshold. To this end we did several experiments on our development sets.

To find a feature set that maximizes clustering precision, we did ablation studies to determine our ideal feature set. The results for WD are in Table 3. For description of each feature, refer to subsection3.1. As it can be observed, each feature we added improves the average precision of the three clustering metrics while increasing or maintaining the average recall of the clustering metrics. The only exception is average mention similarity which slightly reduces average clustering precision while improving average clustering recall. Nevertheless, we felt comfortable keeping this feature since error analysis revealed that adding this feature enabled resolving interesting cases (as described in subsection 5.4). It has to be noted that for the WD classifier when we add the POS tags, we lemmatize the headword before we calculate headword similarity. We feel lemmatization will ensure higher similiarity scores for different conjugations of verbs and the POS tags can act as the safety that ensures too much information is not lost in the process. This works well in practice, as is shown in 5.4. For CD coreference, lemmatization even with POS tags led to loss in the pairwise results(and subsequently clustering), which makes sense. For CD coreference, loss in information is tough to overcome as evidence for coreference is already low even in case of true coreferences.

To determine the pairwise coreference threshold, we ran experiments an example of which is shown in 2. The value chosen to maximize clustering metric's precision scores for the WD pairwise classifier was 0.85 and for the CD pairwise classifier this threshold was 1.0 (i.e. the probability of coferenece is high enough to overflow to 1). The high threshold value for CD coreference is necessary to mantain precision as higher confidence is needed to resolve CD coreference, which is a much harder problem than WD coreference(for reasons mentioned in Section 1). As we can see from the table, our system has high precision in detecting coreference and performs well on the large amounts of incoreferent pairs(especially striking in the WD case which is trained on a balanced training set to avoid distribution bias).

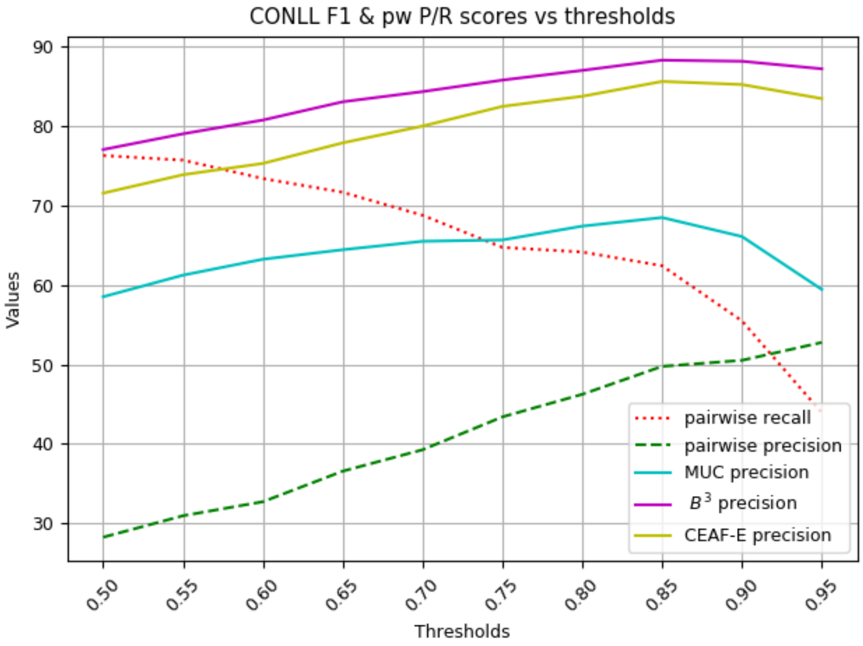

Fig. 2: Determining WD classifier threshold to maximize precision

	Coreference	#Coref	#Non-coref	TP	TN	R	P	F_1	Accuracy	Accuracy
	Threshold	Links	Links						(Non-Coref)	(All)
WD-gold	0.85	1,799	12,701	947	12466	52.64	80.12	63.54	98.14	92.50
WD-detect	0.85	1,212	7927	663	7785	54.70	82.36	65.74	98.20	92.43
CD-gold	1.00	24,315	144,515	10488	142817	42.97	86.02	57.31	98.88	90.78
CD-detect	1.00	16,329	91333	7,555	90278	46.27	87.75	60.59	98.84	90.87

Table 2: Intrinsic evaluation for pairwise classifier with the test sets using final model

	WD Model									
	MUC			**B^3**			**CEAF-E**			**CoNLL**
	R	P	F_1	R	P	F_1	R	P	F_1	F_1
Embeddings(mention + context)	39.20	33.56	36.16	81	72.22	76.36	67.36	73.15	70.14	60.89
+Headword cos. sim.	44.00	68.75	53.65	82.1	92.59	87.03	87.72	77.64	82.37	74.35
+Avg. mention cos. sim.	65.6	60.74	63.07	88.18	84.01	86.04	79.20	82.87	80.99	76.7
+WordNet path similarities	74.4	64.58	69.14	90.68	83.73	87.06	80.40	86.71	83.44	79.88
+POS tags	69.59	67.44	68.50	89.09	87.55	88.31	84.54	86.74	85.63	80.81

Table 3: WD ablation Results

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 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 our WD threshold 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 satisfying the CD threshold 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 [13]. As described in Table 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 [26]. MUC [29] measures how many links need to changed to get the correct clustering. B^3 [5] measures the overlap between the predicted and gold clusters for each mention and computes the average scores(hence can overcount same chain). CEAF-E [23] measures the 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 published in previous work.

1. LEMMA: This baseline groups event mentions into clusters if they have the same lemmatized head word. This is often considered a strong baseline [30].
2. HDDCRP: This baseline is the supervised Hierarchical Distance Dependent Chinese Restaurant Process [30] that is evaluated on the same ECB+ dataset. This model uses distances between event mentions generated, using a feature-rich learnable distance function, as Bayesian priors for single pass non-parametric Bayesian clustering.
3. Iterative WD/CD Classifier: This baseline is the iterative event coreference model that gradually builds event clusters by exploiting inter-dependencies within both WD and CD mentions until the clusters converge [11].

5.2 Our System

We evaluate our system with two sets of 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 as they use the same testing data.

5.3 Results

Table 4 shows the results obtained for WD coreference while Table 5 shows the results obtained for CD coreference. Critically, in all clustering measures, our precision is very high achieving our aim of high quality clustering. We notice that in the case of WD coreference, our system overall performs slightly better than the state-of-the-art [11] for the CoNLL F_1 score. However for CD coreference our system performs about 5% points better than the state-of-the-art in the same measure. Our MUC recall scores for both our Detect systems drop sharply compared to our Gold systems. Since MUC recall is the number of common links between the reference clusters and the system generated clusters, divided by the number of links in the reference clusters (a very non-descriminate measure to the quality of the links missing/present). Our simple system is not sophisticated enough to prioritize making links explicitly between the detected mentions; hence such a fall in performance can be expected for MUC recall. Additionally, the pairwise recall of our gold system is low to begin with, since our pairwise system

	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)	43.95	**84.79**	57.90	64.44	**95.86**	**77.07**	72.79	**78.14**	**75.37**	**70.11**
WD-gold (this paper)	60.94	84.56	70.83	82.92	95.11	88.6	87.99	77.7	82.53	80.65

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. (2015)	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)	41.94	**82.58**	55.63	**64.96**	**95.38**	**77.28**	**73.41**	**76.96**	**75.14**	**69.35**
CD-gold (this paper)	61.2	75.00	67.40	84.08	90.52	87.18	84.13	76.94	80.38	78.32

Table 5: CD Coreference Results

is tuned for higher precision. We also notice that our WD-gold and CD-gold systems perform, as expected, significantly better than the WD-detect and CD-detect systems, illustrating the need for better mention detection systems.

The strength of our results are explained primarily by the strength of our pairwise classifier. Here are some observations about the strength of the pairwise classifier even though its relatively very simple:

- The fact that we prioritize precision in the pairwise classification stage through selection of features and threshold mean that we also get better quality clusters as evidence by our high clustering precision scores
- When we train our pairwise classifier, we use generate non-coreferent pairs by selecting for each mention m which is alteast coreferent with one other mention, a non-coreferent mention from the same sentences as itself (the two mentions making a non-coreferent pair), or a non-coreferent mention from a sentence which contains a mention coreferent to m. This is to ensure that the classifier learns to discriminate between the mention pairs well, leading to higher precision in pairwise coreference resolution.
- Unlike in previous work, we do not use event arguments explicitly and avoid the error propagation from that source. In fact when we did use event arguments, we noticed that our performance dropped on the development set.
- The inclusion of WordNet (not used in previous work) seems to help in a lot of non-trivial cases. For "*Ram-raiders **ploughed** their vehicle into an upmarket jewellery boutique near. . .*" and "*The police source . . . said the men **drove** a large four - by four car into. . .*", WordNet indicates that there is a sense of 'drove' that is related

to application of force which is a sense that helps in coreference with ‘ploughed’. Wordnet also helps solve the coreference between rammed in the sentence “*Four men **rammed** their car into an upmarket jewellery store*” and ‘drove’ in the second sentence above.

5.4 Error Analysis

Since we construct coreferent clusters of event mentions in an agglomerative way, the primary disadvantage we face is error propagation from pair-wise coreference. To mitigate this, we employ high thresholds to determine if two clusters (components) can be merged. However, there still are several false positives as can be seen from Table 2. Additionally despite the high thresholds, there are also some smaller proportion of false negatives. We have performed an analysis of our system’s final predictions on the development set to identify why we get these errors. We will first describe two major problems and then describe our ablation based error analysis in more detail.

– **Incorrect coreference Links:** One recurring issue is that the ECB+ corpus is incompletely marked. This leads to several instances where the system correctly (by our judgment) detects coreference but is not marked as such in the corpus. For example, consider the following pair of events: “*Robbers crash 4x4 into store , **grabbing** jewelry and watches , before setting car ablaze.*” and “*Four men rammed their car into an upmarket jewellery store in central Paris on Monday, **smashing** the shop window . . .* ”. Not having event arguments also plays a role leads to inability to distinguish between events, for instance, for the same two sentences above, our system marked the events **grabbing** and **smashing** as co-referent, but knowledge of the arguments of those events (if they could be accurately identified) will help differentiate between them (jewelry and watches are grabbed, shop window is smashed) even though are contextual cues for event co-reference. We also noticed that having information about event-event relations (such as sequencing, causation, sub-event) will greatly help in avoiding both false positives (as well as false negatives). For instance, consider the following sentences: “*The **heist** near the upscale Place Vendome is the latest to hit France after a spate of high - profile robberies in the southern resort of Cannes .*” and “*The heist near the upscale Place Vendome is the latest to hit France after a spate of high - profile **robberies** in the southern resort of Cannes .*”. The detection of the after relationship would have ensured the prediction of lack of co-reference, but our system incorrectly marks these sentences as coreferent.
– **Missed coreference links:** Another issue is that the word2vec model is not able to not always successfully compare between multi-word events like *scooping up*, *making off*, *cleaning up* and *stay alive* and other idiomatic phrases, since we use pre-computed word2vec model model embeddings.
– **Scare training data for pronominal event coreference** The ECB+ corpus also does not have substantial amounts of events referenced to as pronouns (it, itself, etc.) and hence is not good at resolving events to pronominals. Consider the following pair of events: “*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 addition to the above general problems we did some ablation based error analysis to figure out what each feature we add contributes using our development set. For this purpose we analysed development set pairs(event mention pairs) that move in the confusion matrix(e.g. FN to TP) when a new feature is added(i.e. between ablations). The development set has 173 coreferent pairs and 1432 non-coreferent pairs. We started with the first row in table 3 and proceeded downwards. Starting from the embeddings((average event mention embeddings and context embeddings) as the first feature is intuitive, since it's considered the single most important feature in terms of performance[22].

– **Adding headword cosine similarity**: Of the 251 pairs that move in the confusion matrix, 178 cases move from FP to TN. This massively increases pairwise precision. Were before the model was unable to resolve non-coreference between actions like "climb" and "fall" that occur in similiar contexts, our new model correctly resolves to non-coreference between "...died while **climbing** to a..." and "...when he **fell** in the..." (event mentions in bold). 33 cases move from FN to TP and most of these movements is cases where the event mention embeddings are very similiar. For e.g., event mention pairs like: "...Pierre Thomas **placed** on injured..." and "...Orleans Saints **placed** Pierre Thomas..." is now marked as coreference with high confidence.
– **Adding average mention similiarity**: Of the 112 pairs that move in the confusion matrix, 73 cases move from TN to FP but 39 cases go from FN to TP. The former is mainly due to the fact that adding average mention similiarty for one word mentions just replicates the event mention embedding itself. Consider the following pair: "...to their **deaths** in New Zealand ..." and "...Duncan Rait **died** after slipping...". Here replicating the event embeddings that are similiar pushes the system towards coreference even though the former refers to the death of several people.
  Of the cases that go from FN to TP, the main factor seems to be illustrated by cases like the pair "...climber who **fell** on Friday ..." and "...witnessed the **fall** and had ..." which are coreferent, the replication of the event mention gives confidence to the system that in fact the pair is coreferent. In other words, one of the effects of adding this feature is to add weight to the average mention similiarity. Additionally, averaging the mention does help in correctly resolving multiword events. For instance, consider the pair: "...[placed Pierre] Thomas on **injured reserve** Wednesday because..." and "...placed on **injured reserve**..." is now correctly resolved. Overall pairwise precision takes a small hit at the cost of recall.
– **Adding WordNet features**: Of the 105 pairs that move in the confusion matrix, 21 cases move from FN to TP, 33 cases from FP to TN , 10 cases from TP to FN, 41 cases from TN to FP. Here again, pairwise precision increases. Adding semantic information boosts the discriminanting capacity of the classifier. Now our classifier is able to distinguish between pairs like: "Ram-raiders **ploughed** their vehicle into an upmarket jewellery boutique near..." and The police source ...said the men **drove** a large four - by four car into...' is now correctly marked as coreferent as mentioned in subsection 5.3.

On the other hand Since we do not perform word sense disambiguation, all senses of a mention are considered when semantic features are generated, which might not always be helpful. For instance, consider the pair: ”Climber dead after Aoraki Mount Cook **fall**” and ”The man, from Hampton East, **fell** in the Tasman Glacier area”. This is a case that moves from TP to FN and it's attributable to the WordNet senses in which ”fall” and ”fell” are different.

- **Adding POS tags** (and lemmatizing head word when calculating headword cosine similarity): Of the 53 pairs that move in the confusion matrix, 30 cases move from FP to TN, 17 cases from TN to FP and 6 cases from TP to FN. Pairwise precision increases here as well. Adding POS tags helps the classifier identify plurality/singularity and differentiate between event mentions based on them. For instance, consider the following pair: ”France jewels **thefts** : Robbers ram 4x4 into Paris shop” and ”French police are investigating a daring jewellery **robbery** in Paris . . . ”. This pair is now correctly identified as non-coreferent where as it was not previously. One reason could be that **thefts**' POS tag identifies plurality while **robbery**' identifies singularity. Additionally, in pairs like ”Duncan Rait **died** after slipping . . . ” and ”. . . when he **fell** , sliding down . . . ”, the lemmatization ensures that the headword similarity is calculated between ”die” and ”fell” which improves matters. In fact in our development set, this particular type of example(that of distinguishing between verbs) is improved, once we include lemmatization(accounting for most of the FP to TN cases).

Thus the error analysis adds support to the conclusion that we chose features that by and large improves pairwise precision.

6 Conclusion and Future Work

Our results show that pairwise models that favor precision can be used as a basis to find accurate coreferent clusters of event mentions, and that such models can outperform existing event coreference clustering systems even without relying on event arguments explicitly. These results also 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 detected by a event detector.

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 like to do joint entity and event coreference to improve the overall event coreference resolution. We would also like to combine the aforementioned two ideas and explore building a generic neural model without hand-crafted features that builds sentence representations to solve event co-reference, in the mould of work used in Recognizing textual entailment [28]. Such an end-to-end LSTM model can be used for event coreference in line with recent work on entity coreference[18].

Acknowledgements

This publication was made possible by NPRP grant NPRP 8-1337-1-243 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

References

1. Ahn, D.: The stages of event extraction. In: Proceedings of the Workshop on Annotating and Reasoning about Time and Events. pp. 1–8. Association for Computational Linguistics (2006)
2. Allan, J., Carbonell, J., Doddington, G., Yamron, J.: Topic detection and tracking pilot study final report. In: Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop (1998)
3. Araki, J., Mitamura, T.: Joint event trigger identification and event coreference resolution with structured perceptron. In: Proceedings of EMNLP. pp. 2074–2080 (2015)
4. Azzam, S., Humphreys, K., Gaizauskas, R.: Using coreference chains for text summarization. In: Proceedings of the Workshop on Coreference and its Applications. pp. 77–84. Association for Computational Linguistics (1999)
5. Bagga, A., Baldwin, B.: Algorithms for scoring coreference chains. In: The First LREC Workshop on Linguistic Coreference. pp. 563–566 (1998)
6. Bejan, C.A., Harabagiu, S.: Unsupervised event coreference resolution with rich linguistic features. In: Proceedings of ACL. pp. 1412–1422. Association for Computational Linguistics (2010)
7. Bejan, C.A., Harabagiu, S.: Unsupervised event coreference resolution. Computational Linguistics **40**(2), 311–347 (2014)
8. Bird, S., Klein, E., Loper, E.: Natural Language Processing with Python — Analyzing Text with the Natural Language Toolkit. O'Reilly Media (2009)
9. Chen, Z., Ji, H.: Graph-based event coreference resolution. In: Proceedings of the 2009 Workshop on Graph-based Methods for Natural Language Processing. pp. 54–57. Association for Computational Linguistics (2009)
10. Chen, Z., Ji, H., Haralick, R.: A pairwise event coreference model, feature impact and evaluation for event coreference resolution. In: Proceedings of the Workshop on Events in Emerging Text Types. pp. 17–22. Association for Computational Linguistics (2009)
11. Choubey, P.K., Huang, R.: Event coreference resolution by iteratively unfolding interdependencies among events. In: Proceedings of EMNLP. pp. 2117–2123 (2017)
12. Cybulska, A., Vossen, P.: Guidelines for ECB+ annotation of events and their coreference. Tech. rep., NWR-2014-1, VU University Amsterdam (2014)
13. Cybulska, A., Vossen, P.: Using a sledgehammer to crack a nut? Lexical diversity and event coreference resolution. In: LREC. pp. 4545–4552 (2014)
14. Cybulska, A., Vossen, P.: "Bag of Events" approach to event coreference resolution. supervised classification of event templates. International Journal of Computational Linguistics and Applications **6**(2), 11–27 (2015)
15. Cybulska, A., Vossen, P.: 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. pp. 1–10 (2015)
16. Kenyon-Dean, K., Cheung, J.C.K., Precup, D.: Resolving event coreference with supervised representation learning and clustering-oriented regularization. CoRR **abs/1805.10985** (2018)

17. Lee, H., Recasens, M., Chang, A., Surdeanu, M., Jurafsky, D.: Joint entity and event coreference resolution across documents. In: Proceedings of EMNLP-CONLL. pp. 489–500. Association for Computational Linguistics (2012)
18. Lee, K., He, L., Lewis, M., Zettlemoyer, L.S.: End-to-end neural coreference resolution. In: EMNLP (2017)
19. Liu, Z., Araki, J., Hovy, E.H., Mitamura, T.: Supervised within-document event coreference using information propagation. In: LREC. pp. 4539–4544 (2014)
20. Loria, S., Keen, P., Honnibal, M., Yankovsky, R., Karesh, D., Dempsey, E.: Textblob: simplified text processing. Secondary TextBlob: Simplified Text Processing (2014)
21. Lu, J., Ng, V.: Joint learning for event coreference resolution. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 90–101. Association for Computational Linguistics, Vancouver, Canada (July 2017)
22. Lu, J., Ng, V.: Event coreference resolution: A survey of two decades of research. In: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18. pp. 5479–5486. International Joint Conferences on Artificial Intelligence Organization (7 2018). https://doi.org/10.24963/ijcai.2018/773
23. Luo, X.: On coreference resolution performance metrics. In: Proceedings of HLT-EMNLP. pp. 25–32. Association for Computational Linguistics (2005)
24. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: Proceedings of Workshop at ICLR (2013)
25. Narayanan, S., Harabagiu, S.: Question answering based on semantic structures. In: Proceedings of COLING. pp. 693–701. International Committee on Computational Linguistics (2004)
26. Pradhan, S., Luo, X., Recasens, M., Hovy, E., Ng, V., Strube, M.: Scoring coreference partitions of predicted mentions: A reference implementation. In: Proceedings of ACL. pp. 30–35. Association for Computational Linguistics (2014)
27. Řehůřek, R., Sojka, P.: Software Framework for Topic Modelling with Large Corpora. In: Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks. pp. 45–50. ELRA, Valletta, Malta (May 2010)
28. Rocktschel, T., Grefenstette, E., Hermann, K.M., Kocisky, T., Blunsom, P.: Reasoning about entailment with neural attention. In: International Conference on Learning Representations (ICLR) (2016)
29. Vilain, M., Burger, J., Aberdeen, J., Connolly, D., Hirschman, L.: A model-theoretic coreference scoring scheme. In: Proceedings of MUC. pp. 45–52 (1995)
30. Yang, B., Cardie, C., Frazier, P.: A hierarchical distance-dependent Bayesian model for event coreference resolution. Transactions of the ACL **3**, 517–528 (2015)