Detecting Subevent Structure for Event Coreference Resolution

Jun Araki, Zhengzhong Liu, Eduard Hovy, Teruko Mitamura

Language Technologies Institute, Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213, USA
junaraki@cs.cmu.edu, liu@cs.cmu.edu, hovy@cmu.edu, teruko@cs.cmu.edu

Abstract

In the task of event coreference resolution, recent work has shown the need to perform not only full coreference but also partial coreference of events. We show that subevents can form a particular hierarchical event structure. This paper examines a novel two-stage approach to finding and improving subevent structures. First, we introduce a multiclass logistic regression model that can detect subevent relations in addition to full coreference. Second, we propose a method to improve subevent structure based on subevent clusters detected by the model. Using a corpus in the Intelligence Community domain, we show that the method achieves over 3.2 BLANC F1 gain in detecting subevent relations against the logistic regression model.

Keywords: event coreference resolution, subevent structure, event relation learning

1 Introduction

Event coreference resolution is the problem of determining whether two event mentions refer to the same event. This problem is important in that resolved event coreference is useful in various tasks such as topic detection and tracking, information extraction, question answering, textual entailment, and contradiction detection (Bejan and Harabagiu, 2010).

However, one aspect that makes the problem challenging is that events can form a complex structure and relate to each other in various ways (Huttunen et al., 2002; Bejan and Harabagiu, 2008). In particular, some of event relations exhibit subtle deviation from the perfect identity of events (Hovy et al., 2013). One of these relations is a *subevent* relation. This relation forms a stereotypical sequence of events, or a script (Schank and Abelson, 1977; Chambers and Jurafsky, 2008). Figure 1 shows some examples of that relation in the illustrative text below. In this figure, we say that E15 is a subevent of E12, for example.

> Ismail said the fighting, which lasted several days, intensified when forces loyal to Egal's Ha-bar Awal sub-clan of the Issak **attacked**(E12) a militia stronghold of his main opposition rival, . . .
>
> Egal militia, claiming to be the national defence force, said they had **captured**(E15) two opposition posts, **killing**(E16) and **wounding**(E17) many of the fighters, **destroying**(E18) three technicals (armed pick-up trucks) and **confiscating**(E19) artillery guns and assorted ammunition.

As Figure 1 shows, we see that E15, E16, E17, E18, and E19 form a cluster under their parent E12. Let us call this cluster a *subevent cluster*. In this work, we also pay attention to undirected relations between subevents sharing the same parent. Let us call them *subevent sister* relations, exemplified by lines in Figure 1.

In the context of event coreference resolution, we adopt the approach of (Hovy et al., 2013) in which a subevent relation exhibits partial identity of events whereas normal event coreference represents full identity of them. For clarification, we refer to the latter as *full (event) coreference* in this paper. Unlike previous work on event coreference, we deal with subevent parent-child and sister relations as additional classes to be assigned. Detecting subevent structure is important for event coreference resolution because we can reduce the difficulty of full coreference resolution by excluding subevent relations from candidates of full coreference chains after finding such structure.

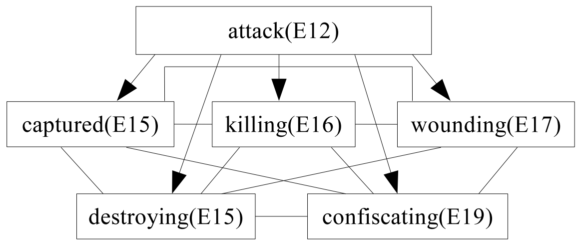


Figure 1: Examples of the subevent relation. An arrow represents a subevent relation with the direction from a parent to its subevent.

In this paper, we propose a novel two-stage approach to detecting subevent structure, and evaluate the approach. We focus on within-document full coreference and subevent relations in the Intelligence Community (IC) domain. The contributions of this work are as follows:

- This is the first work to systematically detect subevent parent-child relations as partial coreference for event coreference resolution. We address the problem from the perspective of subevent structure based on subevent clusters.

- We present a multiclass logistic regression model using a rich set of features to represent different linguistic characteristics, which can identify both full coreference and subevent relations. It is also able to determine the direction of subevent relations.

- We show that the logistic regression model gains reasonable performance for both full coreference and subevent relations.

- We propose a voting algorithm to select out parents for subevents in subevent clusters captured by the model. We show that the method can successfully detect those parents and improve the performance of detecting subevent parent-child relations.

2 Related Work

Event coreference is much less studied as compared to a large body of work on entity coreference. All previous work on event coreference except (Cybulska and Vossen, 2012) deals only with full coreference.

Early work (Humphreys et al., 1997; Bagga and Baldwin, 1999) performed event coreference resolution on scenario specific events. Pradhan et al. (2007) dealt with both entity and event coreference by taking a three-layer approach. Chen et al. (2009) proposed a clustering algorithm using a maximum entropy model with a range of features. They showed that features related to four event attributes had a big impact on intra-document event coreference resolution. Bejan and Harabagiu (2010) built a class of nonparametric Bayesian models using a (potentially infinite) number of features to resolve both intra- and inter-document event coreference. Lee et al. (2011) formed a system with deterministic layers to make coreference decisions iteratively while jointly resolving entity and event coreference. Their systems perform only on full coreference, and do not detect any other type of event relations.

More recently, Cybulska and Vossen (2012) presented an unsupervised model to capture semantic relations and coreference resolution. Although their model considered non-full coreference in addition to full coreference, they did not show quantitatively how well their system performed in each of these two cases. This work also differs from their work in that we focused specifically on subevent parent-child relations while capturing subevent structure.

In relation to subevent structure detection, there has been some previous work on extracting scriptal event schemas. Chambers and Jurafsky (2008) presented an unsupervised learning approach to extracting a chain of temporal ordering of events called narrative schemas. The database of narrative schemas turns out to be useful for detecting subevent sister relations. Regneri et al. (2010) explored a multiple sequence alignment algorithm to construct a graph representation of temporal event structure of scripts. Balasubramanian et al. (2013) employed co-occurrence statistics of triples in the form of (Arg1, Relation, Arg2), and achieved more coherent event schemas. They all focused on subevent sister relations, not on subevent parent-child relations.

3 Approach

3.1 First Stage: Event Relation Learning

Given that event mentions are annotated in a corpus, the goal of this stage is to build up a multiclass event coreference resolver that classifies a relation between two event mentions into one of the following four classes: full coreference (FC), subevent parent-child (SP), subevent sister (SS), or no coreference (NC). Our model is based on the pairwise coreference model (Chen et al., 2009; Bengtson and Roth, 2008), which examines the relation between each pair of two event mentions. We use L2-regularized logistic regression to avoid overfitting. After training, it exclusively assigns one of the four classes above to each pair. We regard this model as our baseline system.

One additional note is that in the case of SP, our system internally models the directionality of that relation from the perspective of the discourse flow. Thus, it can output which event is a parent and which is its subevent, if necessary, in addition to an SP decision.

3.2 Second Stage: Subevent Detection

Our motivation for this stage comes from the result of the first stage. As we describe in Section 4, it turns out that our logistic regression model gains relatively high precision on SS relations. Therefore, we hypothesize that we can rely on the SS relations and resulting subevent clusters obtained in the first stage, and use a voting algorithm to select their parent for improving the system performance on SP relations.

The basic idea is that for each subevent cluster, we enumerate all event mentions (parent candidates) outside the cluster, and calculate probabilities of SP between each parent candidate and the cluster using the logistic regression model trained in the first stage. We then select out an event mention with the highest SP probability as the most likely parent for that cluster among the parent candidates. We consider two options for calculating the highest probability. In Option 1, we regard the highest probability as the highest SP probability among all pairs of parent candidates and sisters in the cluster. In Option 2, we sum up SP probabilities between a parent candidate and the sisters, and take the largest out of the sums.

3.3 Evaluation

Since our system deals with four different relations between event mentions, it is natural to use link-based metrics for evaluation. Thus, we used BLANC (Recasens and Hovy, 2011), which claims that the metric is more adequate for coreference scoring. BLANC was developed to compute precision, recall, and the F1 score separately for two types of link (i.e., positive and negative links), and then average them for the final score. More specifically, if a system gains precision P_p and recall R_p for positive links, and precision P_n and recall R_n for negative ones, the BLANC F1 score is computed as follows:

$$F_{BLANC} = \frac{F_p + F_n}{2} = \frac{P_p R_p}{P_p + R_p} + \frac{P_n R_n}{P_n + R_n}$$

where F_p and F_n denote the F1 score for positive links and negative ones, respectively. Following the original definition, we apply BLANC to the four-class case as follows. Given system output as a 4x4 confusion matrix, we convert the matrix into four 2x2 one-vs-all confusion matrices, each of which represents a binary decision of the system as to each class. From these 2x2 matrices, we compute P_p, R_p, P_n, and R_n for each class, and then use them to compute F_{BLANC}.

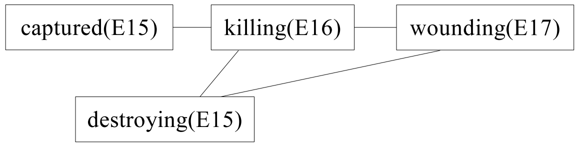
4 Experimental Results

4.1 Corpus

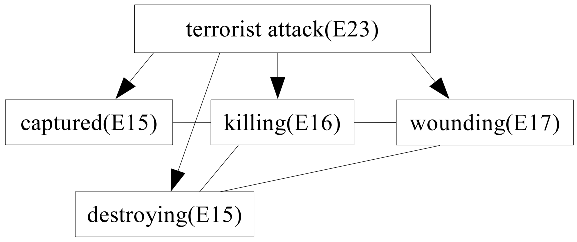
We used a corpus consisting of 65 newspaper articles in the IC domain. The inter-annotator agreement numbers for FC and SP are 0.620 and 0.467 in terms of Fleisss kappa, respectively. In addition to relations manually annotated in the corpus, we also considered subevent relations extended from the combination of FC and SP relations. For instance, if A is a subevent of B, and B is coreferential with C, then A is also a subevent of C. We regarded this type of relation as an SP relation. Table 1 shows the corpus statistics, including our data split. We conducted 5-fold cross validation using the split.

	Training+Dev	Test	Total
# Articles	49	16	65
# Relations	26499	9409	35908
FC	1037	216	1253
SP	997	201	1198
SS	399	139	538
NC	24066	8853	32919

Table 1: Corpus statistics.



(a) A subevent cluster extracted in the first stage.



(b) A subevent structure extracted in the second stage.

Figure 2: An example of system output obtained in the two stages.

4.2 Results

We constructed the logistic regression model using 135 features[1]. We employed MetaCost (Domingos, 1999) to address the data imbalance shown in Table 1. Table 2 shows the system performance in the first stage. In this table, P and R stand for precision and recall, respectively, for positive and negative links, and F1 stands for the final BLANC F1 score. Table 2 indicates that the system achieved relatively high precision on SS relations in the first stage. This is basically because we incorporated more effective features for SS.

Relation	Pos Links		Neg Links		Avg
	R	P	R	P	F1
FC	41.20	41.59	98.64	98.62	70.01
SP	**8.46**	**34.00**	**99.64**	**98.03**	**56.19**
SS	14.39	66.67	99.89	98.73	61.49
NC	98.18	95.36	23.92	45.24	64.02

Table 2: BLANC scores gained in the first stage.

Relation	Pos Links		Neg Links		Avg
SP	R	P	R	P	F1
Option 1	13.43	31.03	99.35	98.13	58.74
Option 2	14.43	33.33	99.37	98.15	59.45

Table 3: BLANC scores gained in the second stage.

Table 3 shows the performance on SP relations in the second stage in terms of the BLANC scores. As compared to the baseline performance (the second row in Table 2), the second stage with Option 1 and 2 improved the BLANC F1 score by 2.5 points and by 3.2 points, respectively. We also see from Table 3 that Option 2 achieved a better performance than Option 1.

[1]See Appendix A for more details.

Figure 2 illustrates how the system performs subevent structure detection through the two-stage process with respect to the subevent structure shown in Figure 1. As shown in Figure 2a, the extracted subevent cluster lost E19, but still captured four subevents out of the five in the gold standard. Figure 2b shows a subevent structure that the system obtained in the second stage from the subevent cluster extracted in the first stage. The system selected out E23 for a parent of the four subevents, which is different from E12 in Figure 1. However, E12 and E23 are coreferential in the gold standard annotation. Hence, all detected links in the subevent structure shown in Figure 2b are correct by means of extended subevent relations.

5 Discussion

The comparison between Option 1 and 2 gives us an interesting insight on voting of subevents in an obtained cluster. Figure 3 provides an evidence to show where the performance difference between the two options comes from. In this figure, a numeric value stands for a subevent probability between a parent candidate and a subevent. E22 is the correct parent for the subevent cluster {E23, E24} in this case. The parent selection algorithm with Option 1 mistakenly chose E21 for the parent because the highest subevent probability 0.881 comes from the subevent relation between E21 and E23.

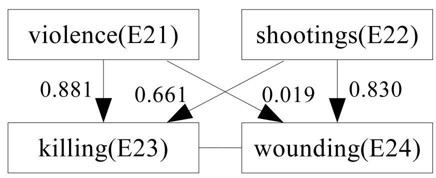


Figure 3: Parent selection from subevent sisters.

Our error analysis indicated that a common error derives from linguistic complexity in the expression of a subevent parent. For instance, E14 and E15 are subevents of E16 in the text below. E16 is a rare, abstract term, making it difficult to capture SP relations.

> Over 90 Palestinians and one Israeli soldier have been **killed**(E14) since Israel **launched**(E15) a massive air and ground **offensive**(E16) into the Gaza Strip on June 28, ...

6 Conclusion

We presented a multiclass logistic regression model that can detect subevent relations in addition to full coreference. We then proposed and evaluated a novel approach to improving subevent structure using a voting algorithm. Our evaluation indicates that the approach achieves significantly better performance gain. To the best of our knowledge, this is the first work to differentiate subevent relations as partial coreference from full coreference, and examine subevent structure including subevent sister relations.

One possible extension to this work is to systematically check structural consistency beyond pairwise decisions and resolve inconsistency in detected subevent structures, thereby obtaining a better performance on SP and SS. In addition, we can construct a library of domain event backbones by aggregating improved subevent structures, and then use it as a background knowledge resource for resolving full coreference in related domains.

7 Acknowledgements

This research was supported in part by DARPA grant FA8750-12-2-0342 funded under the DEFT program. Any opinions, findings, and conclusion or recommendations expressed in this material are those of the authors and do not necessarily reflect the view of the DARPA or the US government. We would like to thank anonymous reviewers for their helpful comments.

8 References

Amit Bagga and Breck Baldwin. 1999. Cross-Document Event Coreference: Annotations, Experiments, and Observations. In *Proceedings of ACL 1999 Workshop on Coreference and Its Applications*, pages 1–8.

Niranjan Balasubramanian, Stephen Soderland, Mausam, and Oren Etzioni. 2013. Generating Coherent Event Schemas at Scale. In *Proceedings of EMNLP 2013*, pages 1721–1731.

Cosmin Bejan and Sanda Harabagiu. 2008. A Linguistic Resource for Discovering Event Structures and Resolving Event Coreference. In *Proceedings of LREC 2008*.

Cosmin Bejan and Sanda Harabagiu. 2010. Unsupervised Event Coreference Resolution with Rich Linguistic Features. In *Proceedings of ACL 2010*, pages 1412–1422.

Eric Bengtson and Dan Roth. 2008. Understanding the Value of Features for Coreference Resolution. In *Proceedings of EMNLP 2008*, pages 294–303.

Nathanael Chambers and Dan Jurafsky. 2008. Unsupervised Learning of Narrative Event Chains. In *Proceedings of ACL-HLT 2008*, pages 789–797.

Nathanael Chambers and Dan Jurafsky. 2009. Unsupervised Learning of Narrative Schemas and their Participants. In *Proceedings of ACL-IJCNLP 2009*, pages 602–610.

Zheng Chen, Heng Ji, and Robert Haralick. 2009. A Pairwise Event Coreference Model, Feature Impact and Evaluation for Event Coreference Resolution. In *Proceedings of RANLP 2009 Workshop on Events in Emerging Text Types (eETTs)*, pages 17–22.

Timothy Chklovski and Patrick Pantel. 2004. VerbOcean: Mining the Web for Fine-Grained Semantic Verb Relations. In *Proceedings of EMNLP 2004*, pages 33–40.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural Language Processing (Almost) from Scratch. *Journal of Machine Learning Research*, 12:2493–2537.

Agata Cybulska and Piek Vossen. 2012. Using Semantic Relations to Solve Event Coreference in Text. In *In LREC 2012 Workshop on Semantic Relations-II Enhancing Resources and Applications*, pages 60–67.

Dipanjan Das, Nathan Schneider, Desai Chen, and Noah A. Smith. 2010. Probabilistic Frame-Semantic Parsing. In *Proceedings of NAACL-HLT 2010*, pages 948–956.

Pedro Domingos. 1999. MetaCost: A General Method for Making Classifiers Cost-Sensitive. In *Proceedings of SIGKDD 1999*, pages 155–164.

Radu Florian, John F. Pitrelli, Salim Roukos, and Imed Zitouni. 2010. Improving Mention Detection Robustness to Noisy Input. In *Proceedings of EMNLP 2010*, pages 335–345.

Kartik Goyal, Sujay Kumar Jauhar, Huiying Li, Mrinmaya Sachan, Shashank Srivastava, and Eduard Hovy. 2013. A Structured Distributional Semantic Model for Event Co-reference. In *Proceedings of ACL 2013*, pages 467–473.

Graeme Hirst and David St-Onge. 1998. Lexical Chains as Representations of Context for the Detection and Correction of Malapropisms. In Christiane Fellbaum, editor, *WordNet: An Electronic Lexical Database*, pages 305–332, Cambridge, Massachusetts. The MIT Press.

Eduard Hovy, Teruko Mitamura, Felisa Verdejo, Jun Araki, and Andrew Philpot. 2013. Events are Not Simple: Identity, Non-Identity, and Quasi-Identity. In *Proceedings of NAACL-HLT 2013 Workshop on Events: Definition, Detection, Coreference, and Representation*, pages 21–28.

Kevin Humphreys, Robert Gaizauskas, and Saliha Azzam. 1997. Event Coreference for Information Extraction. In *Proceedings of ACL/EACL 1997 Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts*, pages 75–81.

Silja Huttunen, Roman Yangarber, and Ralph Grishman. 2002. Complexity of Event Structure in IE Scenarios. In *Proceedings of COLING 2002*, pages 1–7.

Jay J. Jiang and David W. Conrath. 1997. Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. In *Proceedings of International Conference Research on Computational Linguistics (ROCLING X)*, pages 19–33.

Claudia Leacock and Martin Chodorow. 1998. Combining Local Context and WordNet Similarity for Word Sense Identification. In Christiane Fellfaum, editor, *WordNet: An Electronic Lexical Database*, pages 265–283, Cambridge, Massachusetts. The MIT Press.

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 *Proceedings of CoNLL-2011 Shared Task*, pages 28–34.

Michael Lesk. 1986. Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone. In *Proceedings of SIGDOC 1986*, pages 24–26.

Dekang Lin. 1998. An Information-Theoretic Definition of Similarity. In *Proceedings of ICML 1998*, pages 296–304.

Pablo Mendes, Max Jakob, and Christian Bizer. 2012. DBpedia: A Multilingual Cross-domain Knowledge Base. In *Proceedings of LREC 2012*.

Sameer S. Pradhan, Lance Ramshaw, Ralph Weischedel, Jessica MacBride, and Linnea Micciulla. 2007. Unrestricted Coreference: Identifying Entities and Events in OntoNotes. In *Proceedings of the International Conference on Semantic Computing (ICSC)*, pages 446–453.

Marta Recasens and Eduard Hovy. 2011. BLANC: Implementing the Rand index for coreference evaluation. *Natural Language Engineering*, 17(4):485–510.

Michaela Regneri, Alexander Koller, and Manfred Pinkal. 2010. Learning Script Knowledge with Web Experiments. In *Proceedings of ACL 2010*, pages 979–988.

Philip Resnik. 1995. Using Information Content to Evaluate Semantic Similarity in a Taxonomy. In *Proceedings of IJCAI 1995*, pages 448–453.

Roger C. Schank and Robert P. Abelson. 1977. *Scripts, Plans, Goals, and Understanding: An Inquiry into Human Knowledge Structures*. Lawrence Erlbaum Associates.

Stephen Tratz and Eduard Hovy. 2011. A Fast, Accurate, Non-Projective, Semantically-Enriched Parser. In *Proceedings of EMNLP 2011*, pages 1257–1268.

Zhibiao Wu and Martha Palmer. 1994. Verbs Semantics and Lexical Selection. In *Proceedings of ACL 1994*, pages 133–138.

Appendix A Features

Table 4 shows the 135 features used in the logistic regression model described in Section 3.1. They can be organized into five groups as shown in the table. Our feature selection study showed that 'Subevent Ontology' and 'Narrative Schemas' are effective for SP and SS relations. As for the former feature, we developed a subevent ontology tree from our training data set, shown in Figure 5. From the tree, we observed that some event words (e.g., 'raid' and 'explosion') show up as a subevent parent only, and others (e.g., 'kill' and 'injure') as a subevent only, while several words (e.g, 'attack' and 'bomb') can be both. Narrative schemas[2] aggregate structured sets of related events. Figure 4 shows parts of the narrative schemas that are relevant to the IC domain. We observed that the resource is particularly effective in capturing SS relations.

[2] http://www.usna.edu/Users/cs/nchamber/data/schemas/acl09/

score=24.877186
Events: arrest kill shoot charge identify wound found endanger threaten harm
Scores: 6.440 6.149 5.657 5.174 4.690 4.617 4.407 4.283 4.255 4.034
...
score=16.74399
Events: destroy loot burn smash damage steal kill rip
Scores: 5.390 4.751 4.589 4.066 3.748 3.747 3.729 3.139
...
score=12.323438
Events: kill shoot wound ambush murder kidnap
Scores: 5.359 4.346 4.140 3.772 3.683 3.136
...

Figure 4: Excerpts from narrative schemas relevant to events in the IC domain. In each schema, the first line shows the overall score for that schema, and the third shows the individual verb scores, aligned with verbs in the second.

```
Root [350 (100.0)]
|-- attack [87 (24.9)]
|   |-- kill [30 (8.6)]
|   |-- wound [9 (2.6)]
|   |-- injure [6 (1.7)]
|   |-- fire [4 (1.1)]
...
|-- bomb [46 (13.1)]
|   |-- kill [17 (4.9)]
|   |-- injure [6 (1.7)]
|   |-- wound [4 (1.1)]
|   |-- explode [4 (1.1)]
...
|-- fight [24 (6.9)]
|   |-- kill [8 (2.3)]
|   |-- attack [5 (1.4)]
|   |-- wound [3 (0.9)]
|   |-- capture [1 (0.3)]
...
|-- raid [18 (5.1)]
|   |-- kill [4 (1.1)]
|   |-- arrest [3 (0.9)]
|   |-- bomb [1 (0.3)]
|   |-- setting [1 (0.3)]
...
```

Figure 5: Excerpts from the subevent ontology tree constructed from the training data set. The numbers in each node show a frequency of the headword of an event mention and its ratio (percentage) to the total number of occurrences of event mentions, which is 350. The tree shows subevent parents in the first level and subevent sisters in the second level.

Group	Feature Name	Description
Lexical (11)	Event String Similarity	Binary or numeric features by various string similarity measures between headwords of event mentions, including the Levenshtein distance, the Jaro coefficient, and the Dice coefficient.
	Modifier Similarity	Binary or numeric features based on the Dice coefficient between modifiers and event mentions.
Syntactic (44)	Part of Speech	Binary features as to plurality, tense, nominality and verbality of headwords of event mentions.
	Syntactic Dependency	Binary features as to a particular type of dependency between event mentions, annotated by the FANSE parser (Tratz and Hovy, 2011).
	Modifier Similarity	Binary features as to whether event mentions are modified and whether headwords of event mentions are both modified by negation; numeric features by the Dice coefficient of modifiers of event mentions (if both exist).
	Determiner	Binary features as to existence of a determiner of an event mention.
Semantic (41)	Subevent Ontology	A binary feature as to whether event mentions are in the subevent ontology constructed from the training data.
	Narrative Schemas	Numeric features by scores given in the database of Narrative Schemas (Chambers and Jurafsky, 2009).
	Event as Entity	Binary features as to whether nominal event mentions are resolved into entities by the Stanford coreference resolution system (Lee et al., 2011).
	WordNet Similarity	Numeric features by various WordNet similarity scores between event mentions, including (Lesk, 1986), (Wu and Palmer, 1994), (Resnik, 1995), (Jiang and Conrath, 1997), (Hirst and St-Onge, 1998), (Leacock and Chodorow, 1998), and (Lin, 1998).
	SENNA Embeddings	Numeric features by the cosine similarity between word vectors for headwords of event mentions, given by the SENNA system (Collobert et al., 2011).
	Distributional Semantics	Binary features as to whether event mentions are identical, decided by a semantic database of distributional semantic similarity between event mentions. The underlying model to compute distributional semantic similarity is described in (Goyal et al., 2013).
	VerbOcean	Numeric features with a score by VerbOcean (Chklovski and Pantel, 2004) as to a particular relation between head verbs of event mentions.
	Semantic Frame	Binary features as to whether event mentions trigger the same semantic frame, extracted by SEMAFOR (Das et al., 2010).
	Mention Type	Binary features as to whether event mentions have the same mention type, extracted the IBM SIRE system (Florian et al., 2010).
Semantic (arguments) (31)	Agent/Patient	Binary or numeric features as to whether arguments are identical, decided by different matching algorithms (including the Stanford coreference resolution system and the Dice coefficient), and whether the numbers (e.g., 12 in 12 Somali) associated with arguments are identical.
	Location	Binary or numeric features as to whether locations of event mentions are identical. This is decided by various matching algorithms, including the Dice coefficient, the Stanford coreference resolution system, and location subsumption (e.g., New York in the United States) using geographical knowledge bases such as DBpedia (Mendes et al., 2012).
Discourse (8)	Sentence Distance	Numeric features with the number of sentences between two event mentions.
	Event Distance	Numeric features with the number of event mentions between two event mentions.
	Position	Binary features as to whether an event mention is in the title or in the first sentence.

Table 4: List of features for a pair of event mentions. A number within parentheses in each feature group shows how many features belong to that group.