

Graph-based Event Coreference Resolution

Zheng Chen

The Graduate Center
The City University of New York
zchen1@gc.cuny.edu

Heng Ji

Queens College and The Graduate Center
The City University of New York
hengji@cs.qc.cuny.edu

Abstract

In this paper, we address the problem of event coreference resolution as specified in the Automatic Content Extraction (ACE¹) program. In contrast to entity coreference resolution, event coreference resolution has not received great attention from researchers. In this paper, we first demonstrate the diverse scenarios of event coreference by an example. We then model event coreference resolution as a spectral graph clustering problem and evaluate the clustering algorithm on ground truth event mentions using ECM F-Measure. We obtain the ECM-F scores of 0.8363 and 0.8312 respectively by using two methods for computing coreference matrices.

1 Introduction

Typically, an ACE Event Detection and Recognition (VDR) system consists of two steps: first, it detects all mentions of events with certain specified types occurring in the raw text (event mention detection) and second, it unifies the event mentions into equivalence classes so that all the mentions in a given class refer to an event (event coreference resolution). ACE defines the following terminologies related with VDR:

- Event: a specific occurrence involving participants. An ACE event has six attributes (type, subtype, modality, polarity, genericity and tense), zero or more event arguments, and a cluster of event mentions.
- Event trigger: the word that most clearly expresses an event's occurrence.
- Event argument: an *entity*, or a *temporal expression* or a *value* that has a certain role (e.g., Time-Within, Place) in an event.
- Event mention: a sentence (or a text span *extent*) that mentions an event, including a distinguished trigger and involving arguments.

In contrast to entity coreference, the scenarios in event coreference are more complicated, mainly because entity coreference is word (or phrase)-level coreference whereas event coreference is sentence-level coreference and therefore the coreferring event mentions may have more flexible linguistic structures than entity mentions. We provide an example to demonstrate this diversity.

<p>^{EM1}An {explosion} in <u>a cafe at one of the capital's busiest intersections</u> killed one woman and injured another <u>Tuesday</u>, police said.</p> <p>^{EM2}Police were investigating the cause of the {explosion} in <u>the restroom of the multistory Crocodile Cafe in the commercial district of Kizilay during the morning rush hour</u>.</p> <p>^{EM3}The {blast} shattered walls and windows in <u>the building</u>.</p> <p>^{EM4}Ankara police chief Ercument Yilmaz visited <u>the site of the morning blast</u> but refused to say if <u>a bomb</u> had caused the {explosion}.</p> <p>^{EM5}The {explosion} comes a month after ^{EM6}<u>a bomb</u> {exploded} at <u>a McDonald's restaurant in Istanbul</u>, causing damage but no injuries.</p> <p>^{EM7}<u>Radical leftist, Kurdish and Islamic groups</u> are active in <u>the country</u> and have carried out {bombings} in <u>the past</u>.</p>

Table 1. Source text and event mentions

Table 1 shows the source text of a news story. As an example, we only tag the event mentions which have the event type and subtype of (Conflict:Attack). In each event mention, the trigger is surrounded by curly brackets, and arguments are underlined.

Table 2 shows the tabular representation of those event mentions.

Table 3 shows that the five event mentions in event EV1 corefer with each other. We summarize EV1 as follows: a bomb (E4-1) exploded in the restroom (E2-1) of a café (E1-1 or E1-2) during Tuesday morning's rush hour (combination of T1-1, T2-1 and T3-1). EV2 is a different attack event because the target (E6-1) in EV2 differs from the one (E1-3) in EV1. EV3 tells that the bombing attacks have occurred generically

¹ <http://www.nist.gov/speech/tests/ace/>

(thus the event attribute “genericity” is “General” whereas it is “Specific” in EV1 and EV2).

EM1	Trigger: explosion
	Arguments (ID: ROLE): (E1-1: Place) a cafe at one of the capital's busiest intersections (T1-1: Time-Within) Tuesday
EM2	Trigger: explosion
	Arguments: (E2-1: Place) the restroom of the multistory Crocodile Cafe (E3-1: Place) the commercial district of Kizilay (T2-1: Time-Within) the morning rush hour
EM3	Trigger: blast
	Arguments: (E1-2: Place) the building
EM4	Trigger: explosion
	Arguments: (E4-1: Instrument) a bomb (E1-3: Target) the site of the morning blast (T3-1: Time-Within) morning
EM5	Trigger: explosion
	Arguments: None
EM6	Trigger: exploded
	Arguments: (E5-1: Instrument) a bomb (E6-1: Target) a McDonald's restaurant (E7-1: Place) Istanbul
EM7	Trigger: bombings
	(E8-1: Attacker) Radical leftist, Kurdish and Islamic groups (E9-1: Place) the country (T4-1: Time-Within) the past

Table 2. Tabular representation of event mentions

Event	Included event mentions
EV1	{EM1,EM2,EM3,EM4,EM5}
EV2	{EM6}
EV3	{EM7}

Table 3. Event coreference results

2 Event Coreference Resolution as Spectral Graph Clustering

We view the event coreference space as an undirected weighted graph in which the nodes represent all the event mentions in a document and the edge weights indicate the coreference confidence between two event mentions. In real implementation, we initially construct different graphs for separate event types², such that, in each graph, all the event mentions have the same event type. Similar to (Nicolae and Nicolae, 2006), we formally define a framework for event coreference resolution.

² We view the 33 ACE event subtypes as event types

Let $EM = \{em_m : 1 \leq m \leq M\}$ be M event mentions in the document and $EV = \{ev_n : 1 \leq n \leq N\}$ be N events. Let $f: EM \rightarrow EV$ be the function mapping from an event mention $em_m \in EM$ to an event $ev_n \in EV$. Let $coref: EM \times EM \rightarrow [0,1]$ be the function that computes the coreference confidence between two event mentions $em_i, em_j \in EM$. Let $T = \{t_k : 1 \leq k \leq K\}$ be K event types. Thus for each event type k , we have a graph $G_k(V_k, E_k)$, where $V_k = \{em_m | f(em_m).type = t_k, em_m \in EM\}$ and $E_k = \{(em_i, em_j, coref(em_i, em_j)) | em_i, em_j \in EM\}$.

We then model event coreference resolution as a spectral graph clustering problem that optimizes the normalized-cut criterion (Shi and Malik, 2000). Such optimization can be achieved by computing the second generalized eigenvector, thus the name “spectral”. In this paper, we do not try to propose a new spectral clustering algorithm or improve the existing algorithm. Instead, we focus on how to compute the coreference matrix (equivalently, the affinity matrix in Shi and Malik’s algorithm) because a better estimation of coreference matrix can reduce the burden on clustering algorithm.

3 Coreference Matrix W

3.1 Method 1: Computing a Coreference Formula

Obviously, the trigger pair and the argument sets owned by two event mentions carry much information about whether one event mention corefers with the other. Based on a corpus, we compute the statistics about event mention pairs (with the same event type) listed in Table 4.

Let $em_i.trigger$ be the trigger in em_i , $stem(em_i.trigger)$ be the stem of the trigger in em_i , $wordnet(em_i.trigger, em_j.trigger)$ be the semantic similarity between the two triggers in em_i and em_j as computed in (Seco et al., 2004), $em_i.arg$ be the argument (ID and ROLE) set in em_i . Let \cap_1 be the conjunction operator on argument pairs whose ID³ and ROLE match, \cap_2 be the conjunction operator on argument pairs whose ID matches but ROLE does not match, \cap_3 be the conjunction operator on argument pairs whose ROLE matches but ID does not match, \cap_4 be the conjunction operator on argument pairs whose ID and ROLE do not match. We then propose the following formula to measure the coreference value between em_i and em_j .

$$w_{ij} = e^{-\frac{1}{w_{ij}^T + w_{ij}^A}} \text{ where}$$

³ We view two argument IDs “E1-1” and “E1-2” as a match if they mention the same entity which is “E1”

$$w_{ij}^T = \begin{cases} \frac{T11}{T11+T12} & \text{if } em_i.\text{trigger} = em_j.\text{trigger} \\ \frac{T21}{T21+T22} & \text{elseif } stem(em_i.\text{trigger}) = stem(em_j.\text{trigger}) \\ \frac{T31}{T31+T32} & \text{elseif } wordnet(em_i.\text{trigger}, em_j.\text{trigger}) > 0 \\ \frac{T41}{T41+T42} & \text{otherwise} \end{cases}$$

and

$$w_{ij}^A = \frac{1}{\min\{|em_i.\text{arg}|, |em_j.\text{arg}|\}} \times \left[\frac{A11}{A11+A12} |em_i.\text{arg} \cap_1 em_j.\text{arg}| + \frac{A21}{A21+A22} |em_i.\text{arg} \cap_2 em_j.\text{arg}| + \frac{A31}{A31+A32} |em_i.\text{arg} \cap_3 em_j.\text{arg}| + \frac{A41}{A41+A42} |em_i.\text{arg} \cap_4 em_j.\text{arg}| \right]$$

The strength of this formula is that it allows to give credit to different cases of trigger matching and argument pair matching between two event mentions.

T11	in those coreferring event mention pairs, how many pairs use exactly the same triggers
T12	in those non-coreferring event mention pairs, how many pairs use exactly the same triggers
T21	in those coreferring event mention pairs, how many pairs do not have the same triggers, but have the same stems of triggers
T22	non-coreferring version of T21
T31	in those coreferring event mention pairs, how many pairs do not have the same triggers nor the same stems, but the semantic similarity between two triggers is higher than 0 in WordNet.
T32	non-coreferring version of T31
T41	in those non-coreferring event mention pairs, how many pairs are not in T11 or T21 or T31
T42	non-coreferring version that is not T12 or T22 or T32
A11	in those coreferring event mention pairs, how many argument pairs whose ID and ROLE match
A12	non-coreferring version of A11
A21	in those coreferring event mention pairs, how many argument pairs whose ID matches but ROLE does not match
A22	non-coreferring version of A21
A31	in those coreferring event mention pairs, how many argument pairs whose ROLE matches but ID does not match
A32	non-coreferring version of A31
A41	in those non-coreferring event mention pairs, how many argument pairs whose ID and ROLE do not match
A42	non-coreferring version of A41

Table 4. Statistics of event mention pairs

3.2 Method 2: Applying a Maximum Entropy Model

We train a maximum entropy model to produce the confidence values for W . Each confidence value tells the probability that there exists coreference C between event mention em_i and em_j .

$$P(C|em_i, em_j) = \frac{e^{(\sum_k \lambda_k g_k(em_i, em_j, C))}}{Z(em_i, em_j)}$$

where $g_k(em_i, em_j, C)$ is a feature and λ_k is its weight; $Z(em_i, em_j)$ is the normalizing factor.

The feature sets applied in the model are listed in Table 5 by categories.

4 Experiments and Results

4.1 Data and Evaluation Metrics

We developed and tested the spectral clustering algorithm for event coreference resolution using the ACE 2005 English corpus which contains 560 documents. We used the ground truth event mentions and evaluated our algorithm based on ECM F-Measure (Luo, 2005). We reserved 60 documents for testing purpose and used the left 500 documents for training/developing purpose and for computing the statistics discussed above. We applied 10-fold cross-validation in the experiment of comparing two methods for computing coreference matrix.

4.2 Statistics of Event Mention Pairs

The results of the statistics discussed in Section 3.1 are presented in Table 6.

T11=1042, T12=1297, T21=240, T22=840, T31=257, T32=2637, T41=784, T42=5628
A11=888, A12=1485, A21=31, A22=146, A31=542, A32=6849, A41=323, A42=3000

Table 6. Results of statistics in 500 documents

From Table 6, we observe that if two event mentions use the same trigger or if they have arguments whose ID and ROLE match, it is more probable for them to corefer with each other than other cases.

4.3 Comparison of the Two Methods for Computing Coreference Matrix

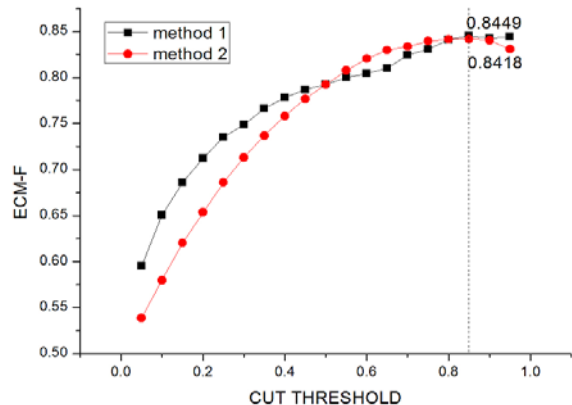


Figure 1. ECM-F scores for both methods

Category	Features	Remarks (EM1: the first event mention, EM2: the second event mention)
Lexicon	type_subtype	pair of event type and subtype in EM1
	trigger_pair	trigger pair of EM1 and EM2
	pos_pair	part-of-speech pair of triggers of EM1 and EM2
	nominal	1 if the trigger of EM2 is nominal
	exact_match	1 if the spellings of triggers in EM1 and EM2 exactly match
	stem_match	1 if the stems of triggers in EM1 and EM2 match
	trigger_sim	quantized semantic similarity score (0-5) using WordNet resource
Distance	token_dist	how many tokens between triggers of EM1 and EM2 (quantized)
	sentence_dist	how many sentences EM1 and EM2 are apart (quantized)
	event_dist	how many event mentions in between EM1 and EM2 (quantized)
Arguments	overlap_num, overlap_roles	overlap number of arguments and their roles (role and id exactly match) between EM1 and EM2
	prior_num, prior_roles	the number and the roles of arguments that only appear in EM1
	act_num, act_roles	the number and the roles of arguments that only appear in EM2
	coref_num	the number of arguments that corefer each other but have different roles between EM1 and EM2

Table 5. EM(Event Mention)-pair features for the maximum entropy model

Figure 1 shows the ECM-F scores for both methods by varying the cut threshold in the clustering algorithm. Both methods obtain the highest ECM-F score at threshold 0.85 and method 1 performs slightly better than method 2 (0.8449 vs. 0.8418, significant at 85% confidence level, $p \leq 0.1447$). We obtained the ECM-F scores of 0.8363 and 0.8312 on the test set for method 1 and method 2 respectively. We also obtained two baseline ECM-F scores, one is 0.535 if we consider all the event mentions with the same event type as a cluster, the other is 0.7635 if we consider each event mention as a cluster.

5 Related Work

Earlier work on event coreference (e.g. Humphreys et al., 1997; Bagga and Baldwin, 1999) in MUC was limited to several scenarios, e.g., terrorist attacks, management succession, resignation. The ACE program takes a further step towards processing more fine-grained events. To the best of our knowledge, this paper is the first effort to apply graph-based algorithm to the problem of event coreference resolution.

Nicolae and Nicolae (2006) proposed a similar graph-based framework for entity coreference resolution. However, in our task, the event mention has much richer structure than the entity mention, thus, it is possible for us to harness the useful information from both the triggers and the attached arguments in the event mentions.

6 Conclusions and Future Work

In this paper, we addressed the problem of event coreference resolution in a graph-based frame-

work, and presented two methods for computing the coreference matrix. A practical event coreference resolver also depends on high-performance event extractor. We will further study the impact of system generated event mentions on the performance of our coreference resolver.

Acknowledgments

This material is based upon work supported by the Defense Advanced Research Projects Agency under Contract No. HR0011-06-C-0023 via 27-001022, and the CUNY Research Enhancement Program and GRTI Program.

References

- A. Bagga and B. Baldwin. 1999. Cross-document event coreference: Annotations, experiments, and observations. In *Proc. ACL-99 Workshop on Coreference and Its Applications*.
- C. Nicolae and G. Nicolae. 2006. Bestcut: A graph algorithm for coreference resolution. In *EMNLP*, pages 275–283, Sydney, Australia, July.
- J. Shi and J. Malik. 1997. Normalized Cuts and Image Segmentation. In *Proc. of IEEE Conf. on Comp. Vision and Pattern Recognition*, Puerto Rico
- K. Humphreys, R. Gaizauskas, S. Azzam. 1997. Event coreference for information extraction. In *Proceedings of the ACL Workshop on Operational Factors in Practical Robust Anaphora Resolution for Unrestricted Texts*.
- N. Seco, T. Veale, J. Hayes. 2004. An intrinsic information content metric for semantic similarity in WordNet. In *Proc. of ECAI-04*, pp. 1089–1090.
- X. Luo. 2005. On coreference resolution performance metrics. *Proc. of HLT-EMNLP*.