Composite Events: A Fact-based Representation

Ndapandula Nakashole, Carnegie Mellon University

Abstract

For any given newsworthy event, thousands of articles, blog posts, micro-blog posts and social network status updates are often published about it. This is an overload to a reader who wants to quickly grasp the key aspects of an event. Addressing this requires a representation that characterizes events in a *succinct* yet *descriptive* manner. This poster proposes a fact-based event representation. Methods from prior work mostly generate events as clusters of related entities. No explicit semantic relations between the entities are given. Thus the entity-based representation exhibits succinctness but lacks descriptiveness. Preliminary experiments show the potential of a fact-based event representation.

Keywords: information extraction; event extraction; event representation;information overload

Citation: Nakashole, N. (2015). Composite Events: A Fact-based Representation. In *iConference 2015 Proceedings*.

Contact: ndapandula@gmail.com

1 Introduction

Motivation. Consider a keen but busy news reader who wishes to keep abreast of important events. The reader may often stop at headlines. However, on occasion, some events capture her attention and she wishes to know more. Time spent, by the reader, understanding the key aspects of the event can be greatly reduced by an effective event representation. Prior methods on event discovery and representation have relied on a general form of relatedness (Angel, Koudas, Sarkas, & Srivastava, 2012; Sarma, Jain, & Yu, 2011; Ritter, Mausam, Etzioni, & Clark, 2012; Sakaki, Okazaki, & Matsuo, 2010; Shahaf & Guestrin, 2010), based on temporal co-occurrences. Under this representation, two entities are said to be in the same story if they co-burst; co-occurring frequently over a given time period (Sarma et al., 2011; Ritter et al., 2012). While concise, this representation is not very descriptive. In particular, the reader has to guess how the various entities in a story relate to one another semantically.

Goal and Contribution. Our goal in this work is to develop a representation for events that is both concise and descriptive. Towards this end, we introduce the *fact-based representation*. Each event is a cluster of highly related facts. A fact is a triple of the form: *subject-predicate-object*. We first discover individual events by identifying related facts. Within each event, our method differentiates facts that are essential to the event from auxiliary ones. Our contribution of thus two-fold: 1) a fact-based representation for events; 2) an algorithm for transforming unstructured news articles into a fact-based representation.

2 Fact-based Event Representation

In our representation, an event is completely defined in terms of its facts as follows:

Fact-based Event. Given a set Φ of facts extracted from a collection of news articles, where a fact is an entity-(typed)phrase-entity triple $(e1, p, e2)$, an event is a set of facts $S_i \subset \Phi$ that are closely related.

Notice that the above definition requires a notion of fact relatedness. In our work to compute fact-relatedness, we make use of the content of the documents from which the facts are extracted. Intuitively, two documents are likely to be discussing the same event if they have a large overlap in the facts they state. Thus, we have:

Fact-based Document Similarity. The fact-based similarity of a pair of documents $d1$ and $d2$ is quantified by the *Jaccard similarity* of their facts.

$$\begin{aligned} sim(d1, d2) &= Jaccard(d1, d2) \\ &= \frac{|\{facts \in d1\} \cap \{facts \in d2\}|}{|facts \in d1\} \cup facts \in d2\}|} \end{aligned}$$

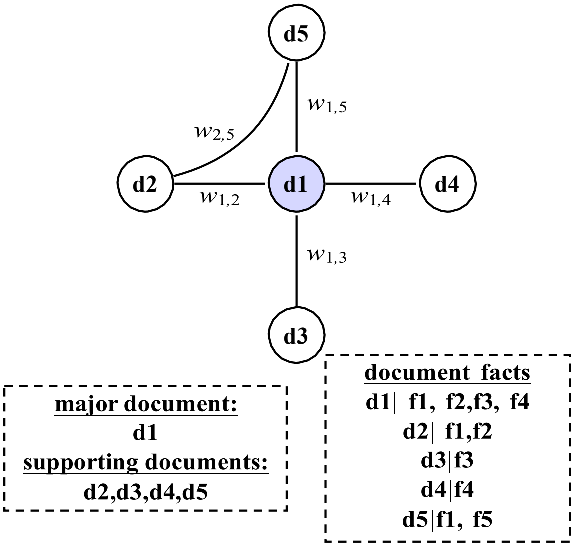

Figure 1: A document-fact graph depicting a single event. Edge weights are inter-document fact similarity.

We extract facts from documents by identifying pairs of noun phrases in a sentence and then checking if the intervening phrase occurs in the Patty collection.

We define a *document-fact graph*, consisting of documents linked based on their fact similarity, as follows:

Document-Fact Graph. A document-fact graph G = (V,E) is a weighted undirected graph consisting of a node set V representing documents and an edge set E representing fact-based document similarity between documents. Each edge has a weight w indicating the fact-based document similarity score.

The *document-fact graph* contains dense clusters of subgraphs. Each such dense cluster expresses a event. Our task is to mine events from the graph. To solve this task, algorithms for identifying dense subgraphs can be applied. Our approach is based on random walks. We run a random walk starting at a document node v, and obtain a ranking score for each visited node $v_j \in V$. The ranking score of a visited node reflects how likely the document is part of the same event. We can apply this algorithm to every document node in the graph; however, it would not be clear what the events are. To counter this, we introduce the concepts of *major documents* and *supporting documents*.

Major & Supporting Documents. Given a document fact graph G, a document node v_i is a major document if it has a large number of immediate neighbors. The document is said to be a major document for a yet to be discovered event S. $V^* \in V$ is the set of all nodes corresponding to major documents. For a given event S_i, its supporting documents is given by nodes V_i^-, which are the nodes that can be reached from one of the event's major document through random walks.

The intuition is that a *major document* is vital to the event. Such a document states facts that are central to the event. It may also contain some non-central facts. We hypothesize that facts that are central to a event are repeated in most documents pertaining to that event. Such facts are usually given as background information to the reader. This means that a *major document* v has a large number of immediate neighbors (above a specified threshold). Nodes in v's neighborhood repeat v's set of facts to some degree. We show an example graph in Figure 1.

2.1 Event Formation Algorithm

Given a document-fact graph we us the event formation algorithm to discover events. The event formation algorithm begins by identifying all *major documents*. The next step is to form neighborhoods around the major documents. For given a *major document* $v \in V^*$, we want to compute a relevance score for the rest of the nodes in the graph. Relevance scores are computed by random walks that start with an initial distribution over the nodes where the entry corresponding to a major document is set to one and all other entries are set to zero. As in standard Markov chains, the probability of taking a particular edge is proportional to the edge weight over all the outgoing edges. Moreover, walks are occasionally restarted with probability c; in these cases, the restart always jumps back to the original starting point. The result of the random walks is an approximation of the stationary distribution of reaching other nodes from the given major document.

All the nodes whose relevance score (with respect to the major document) is above a threshold constitute the *supporting documents*. Note that a document can be a supporting document in multiple stories. *Algorithm 1* describes the event formation procedure.

Algorithm 1 Event Formation

```
procedure ExtractStories($G$)
    $S \leftarrow \emptyset$;   // initialize event set
    $V^* \leftarrow \emptyset$;   // initialize major docs
    for node $v_i \in G$ do
        $L(v_i) \leftarrow$ neighbors of $v_i$;
        if $|L(v_i)| >$ threshold $\theta$
            $V^* \leftarrow V^* \cup \{v_i\}$;
    endfor
        $R \leftarrow$ computeRandomWalkScores(G);
        $V_i^- \leftarrow \{ v_j \in V : (R_j > c) \}$
        $S^i \leftarrow V_i - \cup \{v_i\}$;
        $S \leftarrow S \cup S_i$;
    endfor
    return $S$
```

3 Evaluation

To assess succinctness and descriptiveness, we compared to the entity-based representation using human evaluators from Amazon Mechanical Turk (MTurk). Mturk users were presented with a description of an event in one of the two representations and a list of three news headlines. The headlines were picked from documents belonging to different events. One of the three headlines was from a document of the described event; this ground-truth was not known to the turkers. The turkers were then asked to select the headline which matches the described event. Evaluations were carried out using 50 events. Each event was presented to 3 turkers, resulting in 300 assessments. The results are shown in Table 1.

	Precision
Fact-based representation	*45%*
Entity-based representation	38%

Table 1: Quality assessment of event representations.

The fact-based representation outperformed the entity-based one. However, precision is quite low at 45%. This low precision was largely due to challenges in extracting facts. Therefore, as methods for automatic fact extraction become advanced, our we may also see improvements the fact-based representation..

4 Conclusion

We proposed a structured event representation and presented an algorithm for generating events in this format. We conducted preliminary experiments on Mechanical Turk that showed that our solution has potential. Much remains to be done in terms of extracting precise and complete facts. For future work, we hope to improve our fact extraction module.

References

Angel, A., Koudas, N., Sarkas, N., & Srivastava, D. (2012). Dense subgraph maintenance under streaming edge weight updates for real-time story identification. *PVLDB*, *5*(6), 574–585.

Nakashole, N., Tylenda, T., & Weikum, G. (2013). Fine-grained semantic typing of emerging entities. In *Proceedings of the 51st annual meeting of the association for computational linguistics, ACL 2013, 4-9 august 2013, sofia, bulgaria, volume 1: Long papers* (pp. 1488–1497).

Ritter, A., Mausam, Etzioni, O., & Clark, S. (2012). Open domain event extraction from twitter. In *The 18th ACM SIGKDD international conference on knowledge discovery and data mining, KDD '12, beijing, china, august 12-16, 2012* (pp. 1104–1112).

Sakaki, T., Okazaki, M., & Matsuo, Y. (2010). Earthquake shakes twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on world wide web, WWW 2010, raleigh, north carolina, usa, april 26-30, 2010* (pp. 851–860).

Sarma, A. D., Jain, A., & Yu, C. (2011). Dynamic relationship and event discovery. In *Proceedings of the forth international conference on web search and web data mining, WSDM 2011, hong kong, china, february 9-12, 2011* (pp. 207–216).

Shahaf, D., & Guestrin, C. (2010). Connecting the dots between news articles. In *Proceedings of the 16th ACM SIGKDD international conference on knowledge discovery and data mining, washington, dc, usa, july 25-28, 2010* (pp. 623–632).