HERMEVENT: A News Collection for Emerging-Event Detection

Cristiano Di Crescenzo[a] Giulia Gavazzi[a] Giacomo Legnaro[a] Elena Troccoli[a]

Ilaria Bordino[b] Francesco Gullo[b]

[a]Sapienza University of Rome, Italy
{dicrescenzo.1451751, gavazzi.1546510, legnaro.1724522, troccoli.1624367}@studenti.uniroma1.it

[b]UniCredit, R&D Dept., Italy
{ibordino, gullof}@acm.org

ABSTRACT

News portals and microblogging platforms have become people's medium of choice for breaking news and unexpected events, thanks to their ability to provide directions and useful information more timely and more effectively than official communication channels. This has caused a flourishing of research on event detection in social-media streams. However, this research is severely affected by the scarcity of publicly-available test collections, which are needed to build proper evaluation mechanisms.

In this paper we introduce a new test collection for event detection, which we dub HERMEVENT. The dataset includes a large-scale dump of tweets and news articles from a list of major Italian newspapers, spanning a time interval of approximately 3 months in 2016/2017. From this dump we extracted a set of temporal graphs with different semantic and temporal granularity. To demonstrate the good quality of our data collection, we run two state-of-the-art algorithms that detect emerging events by extracting dense subgraphs from a temporal graph. We conduct an editorial evaluation of the events discovered by the two algorithms on a set of 780 stories, achieving an accuracy of 75% in detecting real-world events. We make the text dump, the graphs and the editorial judgements freely available. We believe that this new dataset can be a really useful contribution to support research on event detection.

CCS CONCEPTS

•Information systems →Web applications; Test collections;

KEYWORDS

Web mining, Web information retrieval, Graph mining, Data collection, Event detection, Entity linking, Emerging events, Entity networks

ACM Reference format:
Cristiano Di Crescenzo[a] Giulia Gavazzi[a] Giacomo Legnaro[a] Elena Troccoli[a] Ilaria Bordino[b] Francesco Gullo[b]. 2017. HERMEVENT: A News Collection for Emerging-Event Detection. In *Proceedings of WIMS '17, Amantea, Italy, June 19-22, 2017,* 10 pages.
DOI: 10.1145/3102254.3102262

1 INTRODUCTION

The gargantuan growth of social media has opened a goldmine of data about events taking place around the world. The real-time nature and massive volume of this data has converted news portals and micro-blogging platforms into social sensors, which people increasingly turn to for breaking news and directions about emerging events, often more timely and more effectively than official communication channels. Allowing people to interact and express their feelings, these media are also a mirror of the collective consciousness, able to provide a more exhaustive picture of how events and facts happening in the real world are perceived by users.

The aforementioned reasons have caused a flourishing of research on event detection in social-media text streams, and event detection has become a well-studied task in information retrieval and data mining [1, 6, 18, 24, 36], with a special focus on Twitter [6, 25, 33, 36], which has been proved to be a very effective medium for breaking news, thanks to its ability to provide a massive real-time stream of updates, opinions, and first-hand reviews.

The design and development of accurate event-detection systems require proper evaluation mechanisms, such as test collections. A test collection typically consists of a set of documents, a list of topics or events of interest, and annotations that specify which documents are relevant to the topics [30]. As a further crucial add-on, a collection may be enriched by structured information about the co-association of the main textual elements within documents, such as words or entities, at a specific semantic granularity.

A serious obstacle to research in this field [27] is given by the fact that social networks and micro-blogging platforms often restrict the access to their content through the usage of search APIs, which allow a very limited number of requests per time. Distribution of archived content as part of a corpus is also largely limited, and datasets created by using automatic methods (e.g., crawling) are against the terms of service of most platforms. An exception to the restricted level of data access of most media is Twitter, which offers a public streaming API providing a random sample of 1%

WIMS '17, Amantea, Italy
© 2017 ACM. 978-1-4503-5225-3/17/06…$15.00
DOI: 10.1145/3102254.3102262

of all postings, plus the possibility of filtering public postings by keywords. Consequently, most of the work on event detection exploits Twitter data, which provides an incomplete view of events, as different media might provide complementary information.

As a result, research on event detection is affected by the scarcity of publicly-available test collections. The majority of works rely on the creation of a bespoke corpus which is often not made available for use by others [4, 5, 7, 15, 35]. To the best of our knowledge, only a few corpora are currently freely available. Among such exceptions, a large corpus that has become a benchmark in the last few years is the one released by Mc Minn *et al.* [20]. This dataset consists of a collection of 120M tweets, with relevance judgements for over 150K tweets, covering around 500 events. The corpus is made available by releasing only user-id and tweet-id pairs, which can be used to crawl Twitter. Other examples include the recent EveTAR collection introduced by Almerekhi *et al.* [2], which consists of a crawl of 590M Arabic tweets posted in a month and covering 66 events, for which more than 134k relevance judgments were gathered using crowdsourcing, and the CREDBANK corpus [22], a set of 60M tweets grouped into 1049 events and manually-labeled by 30 human annotators. However, such corpora provide only unstructured information, namely a set of documents and information about which documents are spanned by real-world events. They miss valuable additional content consisting in how the textual elements within the various documents are structurally organized and interact with each other.

In this paper we introduce a new structured test collection for event detection. The dataset includes a large-scale collection of tweets and news articles from a list of major Italian newspapers, spanning a time interval of approximately 3 months (from December 2016 to March 2017). For each item (tweet or article) the data provides title, url, publication date, and a list of entities extracted from the body of the article. The corpus was built using *Hermes* [8], a recent NLP tool that employs an efficient and extendable distributed-messaging architecture, to achieve the critical requirements of large-scale processing, completeness, and versatility. Hermes is an integrated, self-contained toolkit, which handles all phases of a typical NLP application, from fetching of different data sources to producing annotations such as relevant entities occurring in the text, storing/indexing the content, and making it available for smart exploration/search. We dub our collection HERMEVENT, as an allusion to the tool exploited for its construction.

From the news dump we extract, and also include in HERMEVENT, a set of temporal graphs with different semantic and temporal granularity. We vary the semantic granularity by considering as graph vertices either the entities extracted from the body of articles or tweets by Hermes' entity disambiguation module, or stemmed words. Edges are drawn between pairs of vertices that co-occur in the articles, and are weighted by the number of news where the two vertices co-occur. For what concerns the temporal dimension, we build snapshots varying the granularity of time instants from three hours, to six and twelve hours, and one day. Providing not only a set of articles, but also structural information about it represents a significant addition to our dataset, as it makes it well-suited for the evaluation of graph-based event-detection approaches, which are recognized as the most accurate and general (e.g., they are language-independent) event-detection methods.

To demonstrate the good quality of our dataset and its suitableness as a test collection for event detection or other information-retrieval/graph-mining tasks, we run two prominent state-of-the-art graph-based event detection algorithms on the temporal graphs included in HERMEVENT. The former algorithm considers the graph built at the latest (or current) time instant under consideration. The latter leverages the temporal graph to extract cohesive subgraphs that maximize a notion of temporal anomalous density.

We conduct an editorial evaluation of the events extracted by the two algorithms for ten randomly picked dates, varying the semantic and temporal granularity, and the parameters required by the algorithms. We assess a total of 780 stories, with each story evaluated by three judges. The annotation task consisted in querying a major commercial search engine with the set of entities (or words) composing a story, and annotating the cases where a correspondence with a real-world event was found, marking down a news article describing the related story. We achieve an average accuracy of 75% in detecting real-world events. We make the editorial judgments and annotations available in HERMEVENT.

To summarize, our main contributions are the following:

- We build and make freely available a new test collection for text-retrieval and data-mining tasks, dubbed HERMEVENT.[1] It consists of tweets and news articles, spanning three months between 2016 and 2017. The news items have been collected, pre-processed and cleaned with a recent NLP tool [8]. For each item we provide title, url, publication date, and entities occurring in its body.
- We extract from the news collection, and also make available in HERMEVENT, a set of knowledge graphs with different semantic granularity (entities, words), and temporal granularity (three, six, twelve hours, and a day).
- We run two state-of-the-art graph-based event-detection methods on the temporal graphs, and we editorially assess the quality of a set of 780 events. Such methods achieve an average accuracy of 75% in extracting real-world events, thus testifying the suitability of the collection for the task at hand. We collected 2 340 relevance judgements and annotations (news articles describing the event) that we also make freely available.

We believe that the HERMEVENT collection, enriched with structural information, can be a really useful contribution to support research on event detection. Among others, the dataset can serve for evaluating graph-based event-detection methods, as well as real-time methods, using retrospective data to simulate live data. As far as the language, the HERMEVENT consists of Italian articles/entities. However, the collection is targeted to a task, i.e., event detection, which is inherently language-independent, as the prominent state-of-the-art event-detection methods, such as graph-based approaches, do not depend on the specific language of the underlying data. Moreover, the elementary textual units of HERMEVENT can easily be translated in other languages by using multilanguage lexical databases and/or Wikipedia inter-language links. This makes the HERMEVENT collection general enough to support the most significant event-detection tasks.

[1] http://bit.ly/2oUaaHA

The rest of this paper is organized as follows. Section 2 describes related work, while Section 3 presents the HERMEVENT collection. In section 4 we briefly recall the two event-detection algorithms that we tested on the collection, and in Section 5 we describe our evaluation of the events extracted by the above algorithms. Finally, Section 6 offers our concluding remarks.

2 RELATED WORK

Event detection: methods. Detecting emerging events/stories from user-generated content has received considerable attention in the last years [5, 7, 9, 28, 29]. Earlier approaches are based on the identification of sets of terms/entities exhibiting anomalous temporal evolution in isolation, and a-posteriori grouping them in accordance with their temporal profile [35]. Specifically, such methods assign each term a time series, describing how anomalous (according to a specific anomaly-detection model) its level of occurrence at any time instant is, when compared to the normal level of the whole time horizon. A weakness of these approaches is that they do not exploit any co-association among terms, that is, they do not examine how terms are related to each other and how such relations change over time. Events are rather identified by analyzing each term individually, and grouping terms based on the similarity of the corresponding anomaly time series.

More advanced and effective approaches fall into the category of *graph-based event detection*, where events are detected by building a graph representing the strength of association between terms/entities, and then looking for sets of terms (subgraphs) that are cohesively connected according to a certain notion of cohesiveness [5, 7, 9, 31, 37, 39]. The degree of association between terms, i.e., the weight assigned to each edge in the co-association graph, is typically established by counting how many times those terms cooccur in the considered dataset (e.g., how many news, documents, web-search queries, or tweets contain both terms), or by means of more sophisticated measures, such as correlation measures (e.g., log-likelihood ratio, correlation coefficient) [5, 9, 39], or anomaly scores [7], computed on top of the raw co-occurrence counting.

An orthogonal problem to event detection is how to efficiently maintain events by incremental updating [4]. Moreover, effort in this area has also been devoted to related problems, such as event evolution tracking, i.e., monitoring the evolution pattern of events [16, 19], topic/meme tracking, i.e., monitoring the evolution of specific topics or short, distinctive phrases [12, 17], story-context identification, i.e., building story contexts based on the correlation with other stories [15, 38], story-link detection, i.e., given two stories, determine if they are related to each other (e.g., talk about the same topic) [23, 32], event-timeline generation. i.e., creating a coherent timeline for an event of interest [3, 11].

The dataset we release is specifically targeted to graph-based event detection. However, its general design principles make it suitable to be exploited as a valid benchmark for a number of other related tasks mentioned above, as well as different information-retrieval or graph-mining problems.

Event detection: public datasets. As discussed in the Introduction, the majority of works on event detection build their own evaluation dataset without making it available. The few publicly-available datasets for event detection are mostly Twitter collections whose tweets are grouped into events. Among them, it is worth mentioning the corpus released by Mc Minn *et al.* [20], consisting of 120M tweets, with relevance judgements for over 150K tweets, covering more than 500 events. Other authoritative examples include the EveTAR collection [2], a crawl of 590M Arabic tweets posted in a month period and covering 66 events (in 8 different categories), for which more than 134k relevance judgments were gathered using crowdsourcing, and the CREDBANK corpus [22], containing more than 60M tweets grouped into 1049 events, each annotated by 30 human annotators.

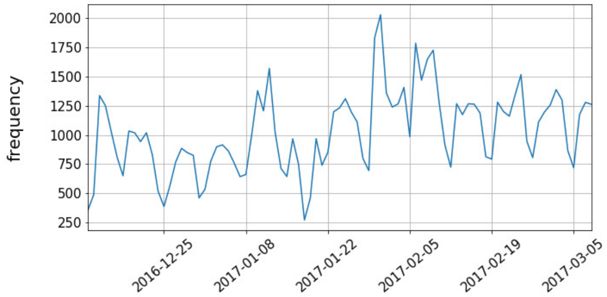

Figure 1: Number of news for each date of the selected time period.

All these public datasets are unstructured, meaning that they typically contain just a set of documents (e.g., tweets), along with some ground-truth information about the documents spanned by an event, and human judgement. In this work we create and release a new structured event-detection dataset, whose main peculiarity consists in describing the co-association among the elementary textual units (i.e., words or entities) within the starting document collection. This makes the dataset well-suited for all those tasks/methods that require data with this kind of structure, such as graph-based event detection.

3 DATASET

In this section we describe in detail our HERMEVENT dataset. We start by detailing all the phases of the dataset-construction process (Section 3.1): news collection, preprocessing, news annotation/information-extraction, temporal-graph building. Then, we report some statistics on the constructed dataset (Section 3.2). Finally, we describe how the collection is shared with the community, including the representation format and how it can be accessed (Section 3.3).

3.1 Construction

News collection and preprocessing. We collect news from the RSS feeds of a list of major Italian online newspapers, which we report in Table 1. We consider a time horizon spanning roughly three months, precisely from December 12th, 2016, to March 7th, 2017. News are collected by exploiting the news-crawling, RSS-feed-processing, and data-cleaning functionalities embedded in the Hermes tool [8]. In particular, as a main data-cleaning operation, we exploit the capability of Hermes to extract the pure textual content of the news, by identifying and removing non-textual content and/or irrelevant content, such as metadata or markups.

Table 1: The list of newspapers used to build HERMEVENT.

it.euronews.com	www.ilsole24ore.com
it.reuters.com	www.ingv.it
tg24.sky.it	www.interno.gov.it
www.agi.it	www.ladige.it
www.ansa.it	www.lagazzettadelmezzogiorno.it
www.corriere.it	www.lastampa.it
www.esteri.it	www.milanofinanza.it
www.gazzettadiparma.it	www.protezionecivile.gov.it
www.ilfattoquotidiano.it	www.rai.it
www.ilgiornale.it	www.repubblica.it
www.ilmattino.it	www.tgcom24.mediaset.it
www.ilmessaggero.it	www.viaggiaresicuri.it

All the news resulting from the pre-processing phase constitute the set of news we ultimately use as input for the construction of our collection. We hereinafter denote such a set by $\mathcal{D}$. Figure 1 depicts the number of news collected in the various dates of the considered period. The overall number of news is 88 092.

News annotation/information extraction. The next step of our dataset-construction process consists in extracting from each news in $\mathcal{D}$ useful information that can profitably be exploited to build the ultimate dataset. We consider two different semantic granularities for building our dataset, i.e., *words* and *entities*, which lead to the definition of two news representations: *word-based representation*, and *entity-based representation*.

As far as word-based representation, we first define a *word vocabulary* $\mathcal{V}_w$ from $\mathcal{D}$. $\mathcal{V}_w$ corresponds to the union of all words belonging to a news in $\mathcal{D}$. Before adding a word to $\mathcal{V}_w$, we employ a number of standard pre-processing/filtering steps so as to avoid populating the vocabulary by non-informative/noisy words. In particular, we perform stop-word removal, and stemming by using the well-established Porter's algorithm [26]. Also, we carried out an accurate analysis of the frequency distribution of a word within the news collection. Based on this, we recognize (and remove) a word as non-informative if its frequency is less than 10. The ultimate word-based representation of a news $d \in \mathcal{D}$ is given by all distinct words in $\mathcal{V}_w$ appearing in d, along with the count of how many times that word is mentioned in d.

The entity-based representation of a news $d \in \mathcal{D}$ is derived by extracting *entities* from it. This corresponds to solving a classic NLP task, termed *Entity Recognition and Disambiguation* (ERD), whose goal is to identify entity mentions in a text (entity recognition) and link them to a proper entity of a given knowledge base (entity disambiguation) [34]. To build the HERMEVENT collection, we solve the ERD task by resorting to the well-known *wikification* approach, which was first proposed by Mihalcea *et al.* [21], and then has had a huge success in the NLP community [10, 13, 14]. The wikification ERD method employs Wikipedia as a knowledge base: each article in Wikipedia is considered as an entity, and the anchor text of all hyperlinks pointing to that article constitute the possible mentions for that entity. All entities are organized in a (directed) graph structure given by the underlying Wikipedia hyperlink graph, where vertices correspond to entities and an arc from entity e_1 to entity e_2 exists if e_1 contains an hyperlink to e_2 in its body. In the wikification process the entity-recognition subtask is easily performed by generating all n-grams occurring in the input text and looking them up in a table that maps Wikipedia anchor-texts to their possible candidate entities.[2] For the entity-disambiguation subtask we employ the popular voting approach of Ferragina *et al.* [10], dubbed *Tagme*. For each news in $\mathcal{D}$ we define its entity-based representation as the set of all its extracted Wikipedia entities that do not match any stop word. Moreover, based on a careful analysis of the distribution of the frequency of an entity within the news collection $\mathcal{D}$, we also discard all entities whose frequency is larger than 3 600. The set of all entities belonging to the entity-representation of a news in $\mathcal{D}$ forms the *entity vocabulary* $\mathcal{V}_e$.

Temporal-graph construction. Given a discrete time horizon $\mathcal{T}$, a *temporal* (or *time-evolving*) graph $\mathcal{G}^{\mathcal{T}} = (V, \{E_t, w_t\}_{t \in \mathcal{T}})$ defined over $\mathcal{T}$ is an undirected, weighted graph with time-invariant vertex set V, and edge set that varies over time. Every time instant $t \in \mathcal{T}$ is assigned an edge set $E_t \subseteq V \times V$, and a function $w_t : E_t \rightarrow \mathbb{R}^+$ assigning weights to edges in E_t. In other words, every time instant t is assigned an undirected, weighted graph $G_t = (V, E_t, w_t)$. Each G_t is dubbed a *snapshot* of the temporal graph $\mathcal{G}^{\mathcal{T}}$.

The word- and entity-based representations of the news in $\mathcal{D}$, along with the word vocabulary $\mathcal{V}_w$ and the entity vocabulary $\mathcal{V}_e$, are exploited to derive two sets of temporal graphs, each one corresponding to a different semantic granularity of the information extracted from the news, i.e., words or entities. Specifically, for each granularity, we define four discrete time horizons as follows. Each news in $\mathcal{D}$ is assigned a timestamp corresponding to the time it has been published. We consider the whole time period spanned by the timestamps of all news in $\mathcal{D}$, and define each one of the considered four time horizons by splitting such a period in fixed intervals of 3 hours, 6 hours, 12 hours, and 1 day, respectively. We denote such time horizons as $\mathcal{T} = 3h$, $\mathcal{T} = 6h$, $\mathcal{T} = 12h$, and $\mathcal{T} = 1d$, respectively. As an example, for a $\mathcal{T} = 1d$, every instant $t \in \mathcal{T}$ represents a one-day-long time interval. Moreover, every time instant in a time horizon is considered to start from time 00:00 of the corresponding day. Then, for instance, in the 3-hour horizon there are eight instants for each day: from 00:00 to 02:59 am, from 03:00 am to 05:59 am, and so on.

Given a time horizon $\mathcal{T} \in \{3h, 6h, 12, h, 1d\}$, and a semantic granularity $x \in \{w, e\}$ (where w stands for "words" and e stands for "entities"), the corresponding temporal graph $\mathcal{G}_x^{\mathcal{T}}$ is defined as follows. The vertex set of $\mathcal{G}_x^{\mathcal{T}}$ corresponds to the vocabulary $\mathcal{V}_x$, i.e., either the word vocabulary or the entity vocabulary, depending on the granularity.[3] For each time instant $t \in \mathcal{T}$, the corresponding edge set E_t is defined based on all co-occurences of any two words/entities in a news whose timestamp belongs to the interval $[t_i, t_{i+1})$. Formally, for any two words/entities $u, v \in \mathcal{V}_x$, we count all news whose timestamp lies within $[t_i, t_{i+1})$ where u and v co-occur. Let $c_t(u, v)$ denote such a count. We draw an edge (u, v) (and add it to E_t) between all pairs of entities u, v such that $c_t(u, v) \geq \eta$, where η is a threshold defining when the association

[2] In our approach we generate up to 5-grams.

[3] For ease of notation, we assume the vertex set of each snapshot in the temporal graph fixed and equal to $\mathcal{V}_x$. In practice, in each snapshot all singleton vertices can be discarded. Thus, the vertex set of each snapshot actually contains only those vertices that have non-zero degree in that snapshot.

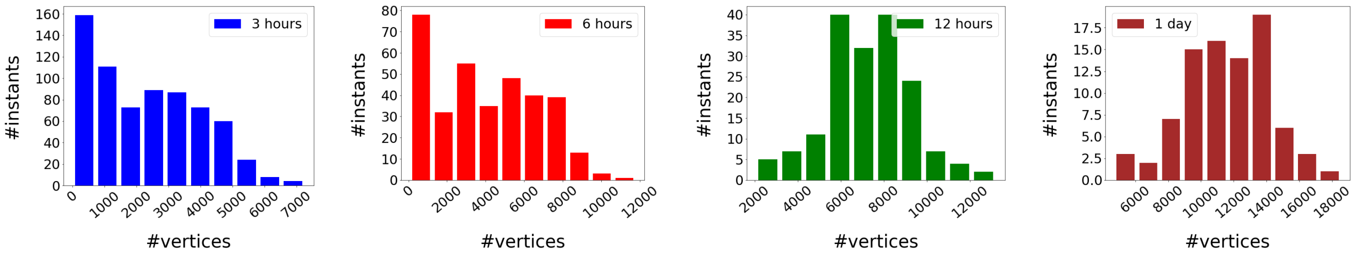

Figure 2: Distribution of number of (non-singleton) vertices of temporal graphs (word granularity).

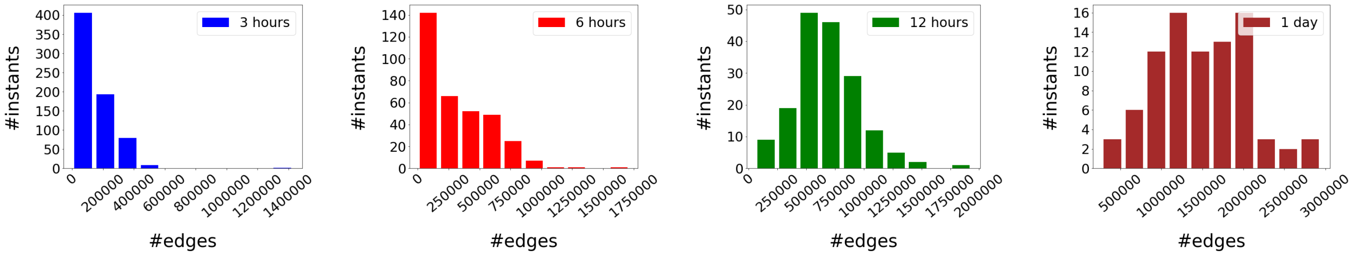

Figure 3: Distribution of number of edges of temporal graphs (word granularity).

between two words/entities is recognized as strong enough.[4] The weight of an edge (u, v) is set as $w_t(u, v) = c_t(u, v)$.

To summarize, we overall generate eight temporal graphs:

- Word granularity: graphs $\mathcal{G}_w^{(3h)}$, $\mathcal{G}_w^{(6h)}$, $\mathcal{G}_w^{(12h)}$, and $\mathcal{G}_w^{(1d)}$;
- Entity granularity: graphs $\mathcal{G}_e^{(3h)}$, $\mathcal{G}_e^{(6h)}$, $\mathcal{G}_e^{(12h)}$, and $\mathcal{G}_e^{(1d)}$.

3.2 Statistics

Table 2 shows the number of instants and the average number of news in each of the four time horizons considered.

In Tables 3–4 we report some aggregated statistics about the temporal graphs we generated: number of (non-singleton) vertices, number of edges, and minimum/average/median/maximum degree of a vertex. All properties are averaged over all time instants of the corresponding horizon. Also, Figures 2–5 show the distribution of number of (non-singleton) vertices and number of edges of all eight temporal graphs.

As expected, the word-granularity graphs are much larger, as the number of words in a news is clearly much larger than the number of entities. The word graphs are one order of magnitude (vertices) and two orders of magnitude (edges) larger than the corresponding entity graphs. The sizes observed with varying the time horizon follow an expected trend as well: the more-fine grained the time horizon, the smaller the corresponding graph. This conforms to common intuition, since smaller time intervals span smaller sets of news, thus reducing the chance of interaction among words/entities.

Table 2: Number of time instants and average number of news for every considered time horizon.

	3h	*6h*	*12h*	*1d*
#time instants	688	344	172	86
avg #news	128	256	512	1 024

Table 3: Average stats of temporal graphs (word granularity).

	3h	*6h*	*12h*	*1d*
#non-singleton vertices	2 007	3 203	5 205	7 820
#edges	189 108	404 081	823 336	1 595 255
min degree	1.83	1.25	1.01	1
avg degree	157.59	216.57	304.21	398.42
median degree	89.48	106.75	126.02	144.63
max degree	1 617.61	2 602.8	4 256.53	6 428.55

Table 4: Average stats of temporal graphs (entity granularity).

	3h	*6h*	*12h*	*1d*
#non-singleton vertices	231	471	935	1 822
#edges	1 688	3 653	7 697	16 570
min degree	1.51	1.15	1	1
avg degree	11.7	12.59	13.78	15.57
median degree	10.66	10.52	10.66	11.27
max degree	40.61	65.05	108.56	193.24

3.3 Output

The HERMEVENT collection is freely available at http://bit.ly/2oUaaHA. In particular, for each dataset $\mathcal{G}_x^{\mathcal{T}}$, where $\mathcal{T} \in \{3h, 6h, 12h, 1d\}$ and $x \in \{w, e\}$, two files are provided: one with filename "$\mathcal{T}.x.vertex$", and a second one with filename "$\mathcal{T}.x.edge$". Both files are JSON files containing information about vertices and edges of the corresponding dataset, respectively. In particular, the

[4] We set $\eta = 2$ for both word and entity granularity.

.vertex file contains the following fields: *time instant*, *vertex ID*, *word/entity*, and *count*, i.e., number of news in the corresponding instant containing that word/entity. Similarly, the *.edge* file includes: *time instant*, *vertex1 ID*, *vertex2 ID*, *edge weight*.

Moreover, we make available the raw set of news that have been employed to create the temporal graphs. The news are available in JSON format from the same URL as above. For each news, the following fields are reported: *news URL*, *timestamp*, *title*, *entities*.

4 ALGORITHMS

In this section we briefly describe the state-of-the-art graph-based event-detection methods we use to test the relevance of our HERMEVENT collection.

The first method we employ is the BUZZ algorithm [7], which has recently been recognized as one of the most effective graph-based event-detection methods. The BUZZ method extracts events from a temporal graph of user-generated content by taking both term co-associations and their (anomalous) temporal evolution into account. The BUZZ method takes a temporal graph as input and extracts emerging events by means of a two-step methodology: (i) applying an anomaly model to quantify how abnormal the association between two terms is at any time, with respect to its history, and (ii) leveraging the graph structure induced by such anomalous associations to identify cohesive subsets of terms that are strongly and anomalously associated with each other in a given time window.

The first step of the BUZZ method corresponds to a task of anomaly detection in temporal data: assign a score to every data point of a temporal sequence according to a model that quantifies its level of anomaly with respect to the remaining points. In our context we have a temporal sequence for each edge in the input temporal graph, and the data points in each sequence correspond to the (raw) weights (i.e., co-occurrence counts) assigned to the corresponding edge over all time instants. In the BUZZ method an unsupervised anomaly model is employed. First, each edge e at time t is assigned a score designed to reflect the relative importance of its weight $w_t(e)$ with respect to all other edges at time t. Importance is measured as the (mass behind the) percentile that the weight of e occupies within the global weight volume at time t. To establish how anomalous the importance of e at time t is, with respect to the past history of e, the e's percentile weight at time t_i is compared to the median of the corresponding percentiles at three *reference* past instants t_{i-r_1}, t_{i-r_2}, and t_{i-r_3}. In our evaluation we set $r_1 = 8$, $r_2 = 12$, and $r_3 = 16$.

The second step of the BUZZ method follows the (general) idea that any document related to a specific event typically tends to involve the same set of main terms. For this purpose, the BUZZ method takes the anomalous temporal graph defined in the previous step as input, along with a time window that denotes the time period under consideration and an integer N denoting the maximum number of terms in any output event, and it seeks K subgraphs of size at most N exhibiting high cohesiveness in the window of interest. As a measure of cohesiveness, the minimum degree over all vertices of the subgraph and over all time instants of the window of interest is used. Such a measure has been demonstrated to be effective and robust to outliers and free riders [7]. The algorithm employed in the BUZZ method to solve this second step extracts a subgraph that optimizes such a min-degree-based cohesiveness measure, subject to the constraint on the size of the output event, i.e., the number of vertices in the extracted subgraph should be no more than N. All vertices (along with all edges incident to them) of the extracted subgraph are removed from the graph. The process iteratively continues until either K subgraphs (events) have been extracted or the graph has become empty.

As a second graph-based event-detection method, we employ a representative of the class of approach that builds a graph of co-associations between terms and looks for cohesive subgraphs in it, without considering deviations (anomalies) from the normal level observed over the entire time horizon, nor how cohesiveness varies over a target time window [5, 9, 31, 37, 39]. As observed in [7], this corresponds to running the BUZZ method on the original graph, i.e., the graph whose edges are weighted with raw term co-occurrence counts instead of anomaly scores, and using a target time window corresponding to the (unique) time instant where events are to be identified. We refer to this method as Raw-Graph Event Detection (RG-ED).

5 EVALUATION

Testbed. We considered the temporal graph and the anomalous temporal graph extracted from the dataset, as described above. We evaluated the BUZZ and RG-ED algorithms on a test set of 10 starting instants, 5 of which were sampled from the graph built with temporal horizon $\mathcal{T} = 1d$ (daily granularity), while the remaining 5 were sampled from the anomalous temporal graph built with $\mathcal{T} = 6h$ (6-hour granularity). For the graph of topics, we fixed the maximum size of each output subgraph to $N = 10$, and the maximum number of output subgraphs to be extracted for each input date to $K = 10$. We run the BUZZ algorithm varying the window size $W \in \{1, 2, 3, 4, 5\}$, and the RG-ED algorithm, which uses no temporal window, with $W = 1$. That led us to extracting, for 10 input instants, 10 subgraphs with 6 different parameter configurations, yielding a total of 600 subgraphs extracted from the temporal graph of topics. We hereinafter refer to a subgraph (i.e., a set of words or entities) extracted by one of the selected event-detection algorithms as a *story*.

In the case of the graph of words, we used the same set of input dates and the same parameter choices that we employed to extract stories from the graph of topics. The word-based stories were more noisy and difficult to evaluate. For this purpose, due to limited editorial resources, we fixed the number of output subgraphs per date to $K = 3$. Therefore, from the graph of words we extracted for 10 input dates, 3 stories with 6 different parameter configurations, yielding a total of 180 stories.

Evaluation: Correspondence with real-world events. Our evaluation was devoted to assessing whether the detected stories match real-world events, which we did by conducting an editorial study with human assessors. We recruited eight judges and asked them to provide a YES/NO answer to the question: "Does the story match a real event?". We encouraged editors to query their preferred search engine with the terms and dates of a story, and explore the corresponding results. In case of a "YES" answer, the

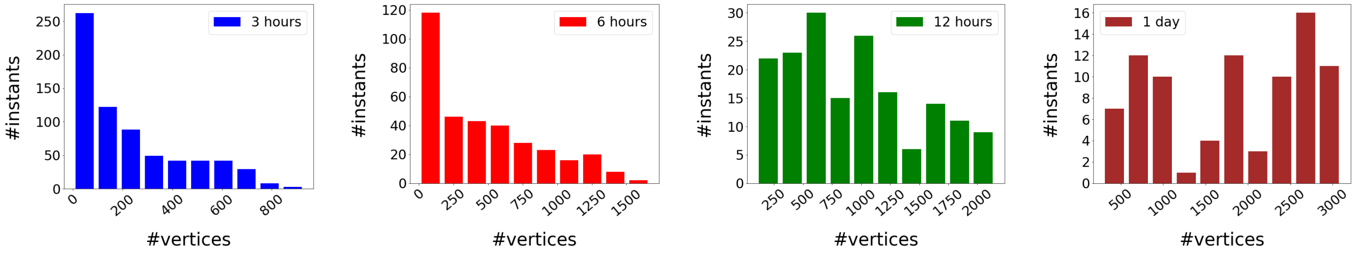

Figure 4: Distribution of number of (non-singleton) vertices of temporal graphs (entity granularity).

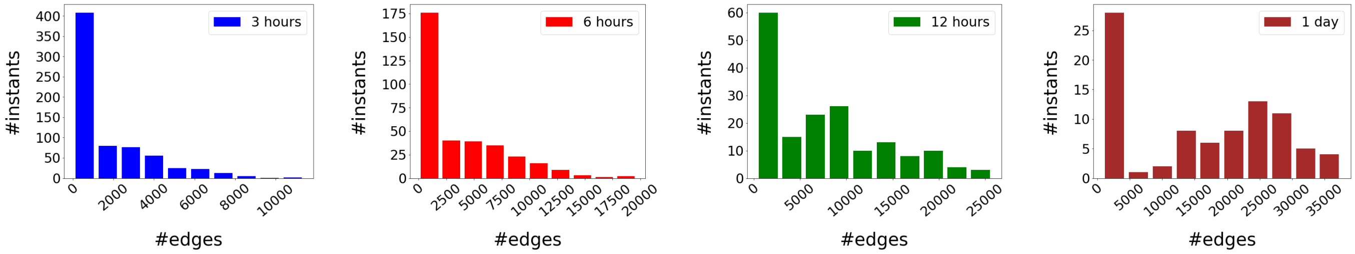

Figure 5: Distribution of number of edges of temporal graphs (entity granularity).

judges were asked to provide a justification of their conclusion (e.g., a link to a news article or web page describing the event).

For each candidate event, editors were shown the topics or words of the story and the dates in the time window. The stories returned by different algorithms and different parameter configurations were randomly mixed. Each judge was asked to assess a fraction of the whole set of stories. All eight judges were employed to evaluate the stories extracted from the graph of topics, while only four worked on assessing the smaller set of stories extracted from the graph of words. We collected three YES/NO labels for each story, and assigned a story the label that was chosen by at least two editors.

Table 5 summarizes the results, illustrating, for each choice of graph, algorithm, and window size taken into consideration, the number of stories extracted, and the editorial decisions assigned to them, which are reported in terms of number and percentage fraction of stories that were marked as corresponding to real-world events ("YES"), followed by the number and percentage fraction of detected stories which were conversely marked as *not* corresponding to actual events ("NO"). Note that the cases $W > 1$ only refer to the BUZZ algorithm, as RG-ED makes no use of a temporal window.

The accuracy of the stories extracted by the two methods was very high: up to 93%, and about 75% on average. This confirms the suitability of our HERMEVENT collection for the event-detection task. In terms of comparison between the two methods, if we look at the stories extracted from graphs of entities (upper half of the table), we can observe that in the case of $W = 1$, i.e., when no temporal window is actually used, the RG-ED algorithm outperforms BUZZ. When the one-day granularity is used to build the snapshots, RG-ED captures 10% of actual stories more than BUZZ. The former algorithm also wins when the six-hour granularity is used for snapshots (76% of stories matching real-world events against the 72% yielded by BUZZ). This might seem to suggest that when no temporal window is used, the raw co-occurrence edge weights employed by the RG-ED algorithm carry more useful information than the anomalous weights used by BUZZ, for the task of detecting relevant and actual events. However, this finding is not confirmed by the results obtained from the graph of words, which are showed in the bottom half of Table 5. For the graphs of words, the opposite finding emerges, i.e., BUZZ outperforms RG-ED: while the two algorithms achieve the same accuracy (93.33%) on the stories extracted from the temporal graph built with daily snapshot granularity, BUZZ is the clear winner in the case of six-hour snapshot granularity, where it yields more than two times the amounts of true stories achieved by RG-ED (93.33% against 46.67%). When using a temporal window larger than 1, the BUZZ algorithm achieves in general a good accuracy in extracting stories that match real-world events, with the exception of some configurations, especially for the case of $W = 3$, which consistently appears as the worst choice for window size in all the experiments. The algorithm performs better with either small window ($W = 2$) or larger windows ($W = 4$ or $W = 5$).

We measured the agreement among editors with the Krippendorff's Alpha coefficient, which is a generalization of the well-established Fleiss' Kappa measure, to be used in case of missing judgements (which are present in our context as every judge evaluated a subset of all extracted stories). We obtained a Krippendorf's Alpha value of 0.411 and 0.486, for the word graphs and entity graphs, respectively.

Anecdotal evidence. In Table 6 we show some examples of stories extracted from HERMEVENT in our experimental evaluation. For each story we report an example link to a news article describing the event captured by the story (chosen among the links provided by the three editors to motivate their positive assessment). As

Table 5: Editorial evaluation

Graph	Method	$\lvert W \rvert$	# Events	YES Events		NO Events	
				#	%	#	%
$\mathcal{G}_e^{(1d)}$	RG-ED	1	50	45	90.00	5	10.00
	BUZZ	1	50	40	80.00	10	20.00
		2	50	34	68.00	16	32.00
		3	50	35	70.00	15	30.00
		4	50	41	82.00	9	18.00
		5	50	40	80.00	10	20.00
$\mathcal{G}_e^{(6h)}$	RG-ED	1	51	40	78.43	11	21.57
	BUZZ	1	50	38	76.00	12	24.00
		2	49	36	73.47	13	26.53
		3	50	30	60.00	20	40.00
		4	50	36	72.00	14	28.00
		5	50	38	76.00	12	24.00
$\mathcal{G}_w^{(1d)}$	RG-ED	1	15	14	93.33	1	6.67
	BUZZ	1	15	14	93.33	1	6.67
		2	15	9	60.00	6	40.00
		3	15	8	53.33	7	46.67
		4	15	9	60.00	6	40.00
		5	15	9	60.00	6	40.00
$\mathcal{G}_w^{(6h)}$	RG-ED	1	15	7	46.67	8	53.33
	BUZZ	1	15	14	93.33	1	6.67
		2	15	14	93.33	1	6.67
		3	15	11	73.33	4	26.67
		4	15	13	86.67	2	13.33
		5	15	12	80.00	3	20.00

the table shows, the two graph-based event detection algorithms employed in our experiments (BUZZ and RG-ED) are able to extract events on different topics that were buzzing in the test days, such as politics, showbiz, crime news, natural disasters or catastrophic events. Some of the stories refers to Italian events, while others report about facts and events that had a worldwide relevance and echo. The first five stories were extracted from the graphs of topics, whereas the last five were obtain from the graphs of words.

Story #1 is about the Rigopiano avalanche, which occurred on the Gran Sasso mountain in Italy, on January 18th 2017, shortly after a series of earthquakes hit the region. The avalanche struck a luxury resort hotel, killing twenty-nine people and injuring eleven others. The story, detected a couple of days after the striking of the avalanche, describes the severe difficulties that emergency services and rescuers faced to reach the hotel, due to large amounts of snow that had fallen for several days prior to the disaster.

Story #2 reports about the protests that occurred in Washington, D.C. during the ceremonies for the inauguration of Donald Trump as the 45th President of the USA. The story includes Obama's remarks about the importance of respecting the freedom of the press.

Story #3 was detected following the announcements of the 2017 Oscar nominations, faithfully capturing the actors and movies that were raising the highest expectations.

Story #4 refers to Elon Musk's announcement that his SpaceX intends to send tourists to the moon: two people have paid for a private mission around the moon, tentatively set for launch in 2018 with the private company's yet untested Falcon Heavy rocket.

Story #5 is about the funeral of Paolo Prodi, a well-known historian and the brother of Romano Prodi, a former politician and economist who served twice as the Prime Minister of Italy.

Among the examples extracted from the graphs of words we find Story #6, which captures a news that had a large echo both in Italy and internationally, about a Catholic priest under investigation for private violence and abetting prostitution.

Story #7 describes the discovery of TRAPPIST-1, a group of seven new Earth-sized planets orbiting a star 39 light years away from the Earth.

Donald Trump appears again in Story #8, which refers to James Robart, a federal judge in Seattle who ordered a national halt to enforcement of Trump's controversial travel ban on citizens from seven predominantly Muslim nations. Judge Robart ruled Friday in favor of Washington Attorney General Bob Ferguson, who sued to invalidate key provisions of Trump's executive order.

Not surprisingly in the present times, the last two stories concern episodes of terrorism. Story #9 is about Anis Amri, a Tunisian suspected to be behind the terror attack on a Christmas market that took place in Berlin, on December 2016. The man was stopped by two police officers in a routine check in Milan. Asked for his documents, Amri pulled out a gun, ensuing a shootout in which he was killed by the police.

Story #10 is about Iakhe Mashrapov, the Kyrgyz man identified as responsible for the night club shooting that took place in Istanbul, on January 3rd, 2017.

6 CONCLUSIONS

In this paper we have introduced HERMEVENT, a novel test collection for detecting emerging events. The dataset includes a large-scale dump of tweets and news articles from a list of major Italian newspapers, from which we have extracted a collection of temporal graphs with different semantic and temporal granularity. To demonstrate the good quality of our data collection and its suitableness as test collection for event detection and other information-retrieval/data-mining tasks, we have run two state-of-the-art graph-based event-detection algorithms, and we have conducted an editorial evaluation of the events discovered by the two algorithms on a set of 780 stories, achieving a good accuracy in detecting stories that match real-world events. The text dump, the graphs and the editorial judgements are made freely available.

We remark that while the few corpora that are currently publicly available as test collections for event detection only provide unstructured information, HERMEVENT is a *structured* test collection for event detection, as it provides not only a set of articles, but also structural information about its content, in the form of graphs with variable semantic and temporal granularity. Such structural information represents a valuable enrichment for the dataset, making it well-suited for the evaluation of graph-based event-detection approaches, which are recognized as the most accurate and general event-detection methods.

We therefore believe that HERMEVENT can be a really useful contribution to support research on event detection.

REFERENCES

[1] C. C. Aggarwal and K. Subbian. Event detection in social streams. In *Proc. of SIAM Int. Conf. on Data Mining (SDM)*, pages 624–635, 2012.

[2] H. Almerekhi, M. Hasanain, and T. Elsayed. Evetar: A new test collection for event detection in arabic tweets. In *Proc. of Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 689–692, 2016.

[3] T. Althoff, X. L. Dong, K. Murphy, S. Alai, V. Dang, and W. Zhang. Timemachine: Timeline generation for knowledge-base entities. In *Proc. of ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD)*, pages 19–28, 2015.

[4] A. Angel, N. Sarkas, N. Koudas, and D. Srivastava. Dense subgraph maintenance under streaming edge weight updates for real-time story identification. *Proc. VLDB Endowment (PVLDB)*, 5(6):574–585, 2012.

[5] N. Bansal, F. Chiang, N. Koudas, and F. W. Tompa. Seeking stable clusters in the Blogosphere. In *Proc. of Int. Conf. on Very Large Data Bases (VLDB)*, pages 806–817, 2007.

[6] H. Becker, M. Naaman, and L. Gravano. Beyond trending topics: Real-world event identification on twitter. In *Proc. of AAAI Int. Conf. on Weblogs and Social Media (ICWSM)*, 2011.

[7] F. Bonchi, I. Bordino, F. Gullo, and G. Stilo. Identifying buzzing stories via anomalous temporal subgraph discovery. In *Proc. of IEEE/WIC/ACM Int. Conf. on Web Intelligence (WI)*, pages 161–168, 2016.

[8] I. Bordino, A. Ferretti, M. Firrincieli, F. Gullo, M. Paris, S. Pascolutti, and G. Sabena. Advancing NLP via a distributed-messaging approach. In *Proc. of IEEE Int. Conf. on Big Data*, pages 1561–1568, 2016.

[9] A. Das Sarma, A. Jain, and C. Yu. Dynamic relationship and event discovery. In *Proc. of Int. Conf. on Web Search and Data Mining (WSDM)*, pages 207–216, 2011.

[10] P. Ferragina and U. Scaiella. TAGME: on-the-fly annotation of short text fragments (by wikipedia entities). In *Proc. of ACM Int. Conf. on Information and Knowledge Management (CIKM)*, pages 1625–1628, 2010.

[11] J. Fulda, M. Brehmer, and T. Munzner. TimeLineCurator: Interactive authoring of visual timelines from unstructured text. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 22(1):300–309, 2016.

[12] Z. Gao, Y. Song, S. Liu, H. Wang, H. Wei, Y. Chen, and W. Cui. Tracking and connecting topics via incremental hierarchical dirichlet processes. In *Proc. of IEEE Int. Conf. on Data Mining (ICDM)*, pages 1056–1061, 2011.

[13] J. Hoffart, M. A. Yosef, I. Bordino, H. Fürstenau, M. Pinkal, M. Spaniol, B. Taneva, S. Thater, and G. Weikum. Robust disambiguation of named entities in text. In *Proc. of Conf. on Empirical Methods in Natural Language Processing (EMNLP)*, pages 782–792, 2011.

[14] S. Kulkarni, A. Singh, G. Ramakrishnan, and S. Chakrabarti. Collective annotation of wikipedia entities in web text. In *Proc. of ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD)*, pages 457–466, 2009.

[15] P. Lee, L. V. Lakshmanan, and E. Milios. Cast: A context-aware story-teller for streaming social content. In *Proc. of ACM Int. Conf. on Information and Knowledge Management (CIKM)*, pages 789–798, 2014.

[16] P. Lee, L. V. S. Lakshmanan, and E. E. Milios. Incremental cluster evolution tracking from highly dynamic network data. In *Proc. of IEEE Int. Conf. on Data Engineering (ICDE)*, pages 3–14, 2014.

[17] J. Leskovec, L. Backstrom, and J. Kleinberg. Meme-tracking and the dynamics of the news cycle. In *Proc. of ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD)*, pages 497–506, 2009.

[18] R. Li, K. H. Lei, R. Khadiwala, and K. C. Chang. TEDAS: A twitter-based event detection and analysis system. In *Proc. of IEEE Int. Conf. on Data Engineering (ICDE)*, pages 1273–1276, 2012.

[19] H. Liu, J. He, Y. Gu, H. Xiong, and X. Du. Detecting and tracking topics and events from web search logs. *ACM Transactions on Information Systems (TOIS)*, 30(4):21:1–21:29, 2012.

[20] A. J. McMinn, Y. Moshfeghi, and J. M. Jose. Building a large-scale corpus for evaluating event detection on twitter. In *Proc. of ACM Int. Conf. on Information and Knowledge Management (CIKM)*, pages 409–418, 2013.

[21] R. Mihalcea and A. Csomai. Wikify!: Linking documents to encyclopedic knowledge. In *Proc. of ACM Int. Conf. on Information and Knowledge Management (CIKM)*, pages 233–242, 2007.

[22] T. Mitra and E. Gilbert. CREDBANK: A large-scale social media corpus with associated credibility annotations. In *Proc. of AAAI Int. Conf. on Web and Social Media (ICWSM)*, pages 258–267, 2015.

[23] T. Nomoto. Two-tier similarity model for story link detection. In *Proc. of ACM Int. Conf. on Information and Knowledge Management (CIKM)*, pages 789–798, 2010.

[24] S. Petrovic, M. Osborne, and V. Lavrenko. Streaming first story detection with application to twitter. In *Proc. of Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT)*, pages 181–189, 2010.

[25] S. Petrović, M. Osborne, and V. Lavrenko. Using paraphrases for improving first story detection in news and twitter. In *Proc. of Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT)*, pages 338–346, 2012.

[26] M. F. Porter. Readings in information retrieval. chapter An Algorithm for Suffix Stripping, pages 313–316. 1997.

[27] C. Reuter and S. Scholl. Technical limitations for designing applications for social media. In *Proc. of Mensch & Computer 2014 - Workshopband*, pages 131–139, 2014.

[28] P. Rozenshtein, A. Anagnostopoulos, A. Gionis, and N. Tatti. Event detection in activity networks. In *Proc. of ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD)*, pages 1176–1185, 2014.

[29] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *Proc. of Int. World Wide Web Conf. (WWW)*, pages 851–860, 2010.

[30] M. Sanderson. Test collection based evaluation of information retrieval systems. *Foundations and Trends in Information Retrieval*, 4(4):247–375, 2010.

[31] N. Sarkas, A. Angel, N. Koudas, and D. Srivastava. Efficient identification of coupled entities in document collections. In *Proc. of IEEE Int. Conf. on Data Engineering (ICDE)*, pages 769–772, 2010.

[32] C. Shah, W. B. Croft, and D. Jensen. Representing documents with named entities for story link detection (SLD). In *Proc. of ACM Int. Conf. on Information and Knowledge Management (CIKM)*, pages 868–869, 2006.

[33] D. A. Shamma, L. Kennedy, and E. F. Churchill. Peaks and persistence: Modeling the shape of microblog conversations. In *Proc. ACM Conf. on Computer Supported Cooperative Work (CSCW)*, pages 355–358, 2011.

[34] W. Shen, J. Wang, and J. Han. Entity linking with a knowledge base: Issues, techniques, and solutions. *IEEE Transactions on Knowledge Engineering (TKDE)*, 27(2):443–460, 2015.

[35] G. Stilo and Velardi. Efficient temporal mining of micro-blog texts and its application to event discovery. *Data Mining and Knowledge Discovery (DAMI)*, 30(2):372–402, 2016.

[36] J. Weng and B. Lee. Event detection in twitter. In *Proc. of AAAI Int. Conf. on Weblogs and Social Media (ICWSM)*, 2011.

[37] J. Weng, Y. Yao, E. Leonardi, and B. Lee. Event detection in Twitter. In *Proc. of AAAI Int. Conf. on Weblogs and Social Media (ICWSM)*, 2011.

[38] M. Zhao, C. Zhang, S. Lu, and H. Zhang. Steller: An approach for context-aware story detection using different similarity metrics and dense subgraph mining. In *Proc. of Int. Conf. on Computer Supported Cooperative Work in Design (CSCWD)*, pages 152–157, 2016.

[39] Q. Zhao, T.-Y. Liu, S. S. Bhowmick, and W.-Y. Ma. Event detection from evolution of click-through data. In *Proc. of ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD)*, pages 484–493, 2006.

Table 6: Examples of stories detected in our experiments.

#	Graph	Date	W	N	K	Story	Corresponding News Article
1	$\mathcal{G}_e^{(6h)}$	2017-01-20 06	2	10	10	protezione civile, neve, valanga, soccorso alpino, corpo nazionale dei vigili del fuoco, rigopiano, italia centrale, guardia di finanza, dipartimento della protezione civile, farindola	http://www.ilfattoquotidiano.it/2017/01/19/terremoto-centro-italia-valanga-sullhotel-le-lacrime-dei-soccorritori-dopo-la-marcia-di-20-ore-qui-non-ce-piu-niente/3326686/
2	$\mathcal{G}_e^{(12h)}$	2017-01-20 00	1	10	5	bill de blasio, michael moore, new york, robert de niro, barack obama, insediamento del presidente degli stati uniti d'america, usa, columbus circle, washington, casa bianca	http://www.rainews.it/dl/rainews/articoli/Oggi-Trump-diventa-presidente-tafferugli-a-Washington-Inizia-la-resistenza-Obama-rispetti-liberta-stampa-a8fe6169-5430-43e0-8b38-dac68615c2
3	$\mathcal{G}_e^{(1d)}$	2017-01-25	3	10	20	ryan gosling, damien chazelle, manchester, natalie portman, emma stone, meryl streep, hacksaw ridge, mel gibson, casey affleck, la la land	http://www.ilpost.it/2017/01/24/oscar-2017-nomination/
4	$\mathcal{G}_e^{(1d)}$	2017-03-03	5	10	30	apollo, orbita terrestre bassa, la nasa, phil larson, stazione spaziale internazionale, fra spacex, programma apollo, esplorazione spaziale, space launch system, space launch system e di orion	http://www.repubblica.it/scienze/2017/02/27/news/spacex_nel_2018_due_turisti_intorno_alla_luna-159397130/
5	$\mathcal{G}_e^{(3h)}$	2016-12-19 12	4	10	5	università di bologna, romano prodi, virginio merola, adriano prosperi, archiginnasio di bologna, paolo prodi, discipline umanistiche, rettore (università), professore	http://www.ansa.it/emiliaromagna/notizie/2016/12/19/a-bologna-il-saluto-a-paolo-prodi_88768950-de1e-4712-b56d-b91180022006.html
6	$\mathcal{G}_w^{(1d)}$	2017-01-05	3	20	5	provvista, fuggito, jugoslavia, lazzaro, catalogati, balcanico, canonica, org sessuali, vibratori, predisposta, avvicinamento, filmate, hard, curia, avvenivano, etichette, minacciata, inconsapevoli, annotava	http://www.tgcom24.mediaset.it/cronaca/veneto/padova-sesso-con-sette-donne-la-parrocchia-dello-scandalo-di-don-andrea_3049658-201702a.shtml
7	$\mathcal{G}_w^{(6h)}$	2017-02-22 18	1	20	10	nana, pianeti, eso, solare, ospitare, astronomi, distante, liegi, telescopio, gillon, temperatura, european, trappist, planetario, sosia, nasa, abitabile, nature, ultrafredda	http://www.ansa.it/canale_scienza_tecnica/notizie/spazio_astronomia/2017/02/22/scoperto-qualcosa-oltre-il-nostro-sistema-solare_a8647f10-e3ee-42ae-8f98-2d395aae841f.html
8	$\mathcal{G}_w^{(3h)}$	2017-02-04 06	4	20	10	appropriato, ingiunzione, bloccarne, aeroporti, eliminato, incostituzionale, muslim, restrittiva, contrari, determinata, hawaii, valida, opposti, fondamento, sospesa, minacce, ban, emessa, inizialmente	http://www.ilfattoquotidiano.it/2017/02/04/trump-giudice-federale-blocca-lo-stop-agli-immigrati-islamici-nessuno-e-sopra-la-legge-nemmeno-il-presidente/3367518/
9	$\mathcal{G}_w^{(12h)}$	2016-12-23 12	2	30	10	amri, fermato, killer, terrorista, strage, spalla, somatici, deceduto, identificato, stazione, scat, attentato, colpendolo, sparando, sparato, sparatoria, anis, poliziotti, poliziotto, pistola, zaino, tunisino, agente, agenti, berlino, ferito, ucciso, movio, fermata	http://www.ansa.it/lombardia/notizie/2016/12/23/milano-spara-ad-agenti-durante-un-controllo-ucciso_7dbfa79d-ca32-4d74-ac88-30038a841756.html
10	$\mathcal{G}_w^{(1d)}$	2017-01-03	1	20	10	croce, turche, serva, turchi, addestrato, caricatori, interrogativi, assunzione, canna, agito, assalto, celebrano, taxi, agendo, eroico, attentatore, nightclub, festivit, agenti, giaccone	http://gds.it/2017/01/03/istanbul-caccia-al-terrorista-cinese-moglie-arrestata-non-sapevo-dellisis_611305/