Social Media: Trends, Events, and Influential Users

Theodoros Semertzidis
Information Technologies Institute, Center for Research and Technology Hellas, Thermi-Thessaloniki, and Electrical and Computer Engineering Department, Aristotle University of Thessaloniki, Thessaloniki, Greece

Christos Athanasiadis
Information Technologies Institute, Center for Research and Technology Hellas, Thermi-Thessaloniki, Greece

Michael Gerassimos Strintzis
Information Technologies Institute, Center for Research and Technology Hellas, Thermi-Thessaloniki, and Electrical and Computer Engineering Department, Aristotle University of Thessaloniki, Thessaloniki, Greece

Petros Daras
Information Technologies Institute, Center for Research and Technology Hellas, Thermi-Thessaloniki, Greece

Abstract
The streaming nature of the social media content, the increasing population of social media users, and the all-connected devices have significantly amplified the amounts of shared content. Navigating through these vast amounts of content and extracting meaningful information and knowledge has become an extremely interesting research topic in recent years. Many researchers have proposed algorithms and methods to organize the shared content into meaningful ways and thus enable efficient navigation through and exploration of the shared content. In this entry, we discuss the progress in three different but overlapping topics: detection of social trends, detection of events, and detection of influential users.

INTRODUCTION

Social media services enable users to create and share content in virtual communities and networks. As recent statistics show, billions of users share content through the major social media sites (e.g., Facebook, Twitter, Google Plus, etc.) every day. Even though social media platforms have been created to enable users' communication and knowledge sharing, the abundant information and its ephemeral nature makes it difficult for the users to navigate and exploit it. The unstructured, noisy, and heterogeneous online content, which most of the times lack any kind of curation, requires a form of aggregation or organization for higher-level semantics to emerge. Moreover, tools for information filtering are needed more than ever before.

There are seemingly different groups of methods and algorithms that try to organize the social content for different goals and aggregate information. These are: a) the detection of social trends; b) the detection of social events; and c) the detection of influential users. These groups of methods share common concepts and have overlaps on the algorithmic tools they use. In the social trends group of methods, the aim is to identify highly popular and interesting content in a certain time frame and to separate it from the noisy and spammy dump of information. On the other hand, the social event detection group of methods aims to identify social events created by people or for people who appear in online networks. There are different definitions of events in various works; however, the general concept is to identify a solid event that happened in a certain point in time and at a specific place. Finally, a cast of methods aims to identify users who influence the rest of the community and are the ones who affect the topics and information chunks that persist for longer periods in a community of users.

In this work, we explain the basic concepts and discuss important works in the literature on the aforementioned groups of methods. The rest of the entry is organized as follows. In the section "Trends Detection in Social Streams," trend detection techniques are presented. In the section "Event Detection in Social Streams," event detection in social streams is discussed. In the section "Influence Detection in Social Streams," influence detection is presented, while the conclusion of this entry and future challenges are discussed in the section "Conclusion."

TRENDS DETECTION IN SOCIAL STREAMS

Trending topics in social streams are defined as sets of words that frequently appear in a discussion that occur often

Encyclopedia of Computer Science and Technology, Second Edition DOI: 10.1081/E-ECST2-120054035

in response to recent real-world events. A set of words or phrases that are tagged at a greater rate than other sets is said to be a "trending topic." Trending topics are becoming popular either through a concerted effort by users, or because of an event that prompts people to talk about a specific topic. These trends help users to understand what is happening in the world in real time. Furthermore, marketers and companies use trend detection tools to discover emerging trends and capture the popularity of products and campaigns or design new marketing strategies based on the extracted trends. Fig. 1 presents a high-level overview of a typical trend detection process to extract trends from a social stream in a specific time period.

In this section, we analyze the state-of-the-art techniques and algorithms that extract dynamically the emerging trends in social media streaming.

One of the most common challenges in social media is to discover subtle trending topics in a constantly changing environment. Due to the dynamic nature of social media content, emerging trends constantly change over time. Therefore, temporal information plays a crucial role in emerging trend detection algorithms. Moreover, the enormous volumes of information shared online makes discovering subtle topics very challenging. The target is a strategy that filters useless information that does not lead to meaningful topics (for example, in text, the articles, pronouns, etc.), as well as a strategy that assembles common information into groups leading to generation of topics.

In social media, every user shares several posts (documents) per day. Every document is a set of several terms that could be either text or multimedia content. An obvious approach to detect topics is to simply measure the raw frequencies of each term. However, it is known that using raw frequencies alone has major drawbacks, as the most frequent terms in the streams tend to be the less informative. A typical process for calculating a more robust score for terms is the term frequency-inverse document frequency (TF-IDF) weighting process. TF-IDF process is an information retrieval algorithm that weights a document relevance to a query based on term's frequency and inverse document frequency. TF (t, d) is the number of times that term t occurs in document d. IDF is a measure of how much information the word provides, that is, whether the term is common or rare across all documents in the corpus. It is the logarithmically scaled fraction of the documents that contain the word, obtained by dividing the total number of documents by the number of documents containing the term, and then taking the logarithm of that quotient. However, the term vector derived from the process is subject to the "curse of dimensionality" when the text of the document is long. Furthermore, the temporal order of words and the semantic and syntactic features of the text are discarded by the term vectors. To overcome these problems, several tools from natural language processing (NLP) are applied, for instance, Latent Dirichlet Allocation (LDA) or Latent Semantic Indexing (LSI)[1] in order to reveal hidden topics from the noisy social streams.

A first categorization of trend identification techniques regarding the features that are used is among: a) *text-based* techniques; b) *multimedia-based* techniques; and c) *metadata-based* techniques. One important category of techniques for trend detection using textual content is the *burst word detection*. A keyword is characterized as burst when it is encountered at an unusually high rate in the social stream. For example, the keywords "world cup" could appear in a rate of 20 documents per minute in the social stream and then suddenly exhibit an unusual high rate of e.g. 5000 documents per minute. Such spike in keyword frequency is typically affiliated with a sudden popular interest in a particular topic and is often driven from emerging news or events. For example, a sudden burst in social stream in keyword "world cup" could be associated with a real-time soccer "world cup" competition. Efficient detecting and grouping of burst words lead to a system that can actually detect trends in a social stream. A different approach in trend identification is the *structural analysis* of social networks in order to detect trends as in the case of Budak et al.[2] A structural trend is a topic that is popular within different subgroups (clusters) of the network. The challenges here are to find the subgroups of the social network and to develop techniques to detect structural trends.

In Jin et al.,[3] a study in multimedia content in Flickr is presented. Authors study the behavior of several trending topics in social media in the domains of politics, economics, and marketing. They use several features to characterize a query in Flickr. These features are divided

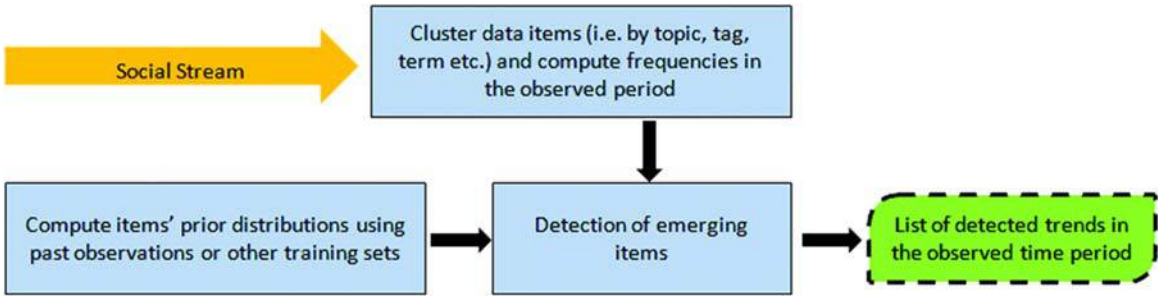

Fig. 1 A high-level schematic of the trend detection process.

into two categories: *metainformation features* (tag, description, spatial, and temporal features) and *visual features*. In their experiments, however, they only use the *metainformation features*. More specifically, the following features/indicators are used in order to build prediction models:

- Relevant images per day (IPD), month (IPM), quarter (IPQ), and year (IPY)
- Relevant images that are tagged with the query keyword (TIPD, TIPM, TIPQ, and TIPY)
- Relevant images tagged with the query keyword from unique users (TUPD, TUPM, TUPQ, and TUPY)

Relevant images in the case of *metainformation* features are the images that are tagged with the query terms. The core of the approach is the implementation of a prediction model to forecast future sales values of products. *Autoregressive models* as well as *bass forecast* models are applied in order to estimate product sales. Authors make the assumption that the number of related photos uploaded online in Flickr can reflect the number of product sales. They perform two experiments with iPod and Mac sales and calculate autoregressive and bass estimations about the correspondent sales using as a feature the tagged images with unique users per quarter (TUPQ). They make use of the seasonal-trend decomposition (STL) algorithm, which is a filtering procedure that decomposes time series in three fundamental components: trend, seasonal, and remainder. They perform STL in TUPM in order to decompose the signal to its fundamental components. The current value of TUPQ and the decomposed trend signal are used in prediction models in order to estimate the future values of TUPQ. It is claimed that Flickr features can provide successful estimation measures of future product sales. Besides the prediction of TUPQ values of products, they perform experiments for Presidential American Election of 2009. The figures of TUPD and TUPM, which stand for tagged images with unique users per day and per month in the elections, respectively, are generated. This study shows that TUPD and TUPM features in Flickr provide hints that indicate the final outcome of the elections.

In Cataldi et al.,[4] a system that detects emerging topics on Twitter is proposed. Keywords are extracted in real time from the Twitter streams and for every tweet a tweet vector is defined as

$$\overrightarrow{\mathbf{tw_j}} = \{w_{j,1}, w_{j,2}, \ldots, w_{j,u}\}$$

where $w_1, w_2, \ldots, w_u$ are the weight for every word in the tweet. The weight is calculated as $w_{j,x} = 0.5 + 0.5 \times (tf_{j,x}/tf_j^{max})$, with $tf_{j,x}$ to be the term frequency of the keyword in the jth tweet and x the index of the keyword in tweet. Therefore, every streaming tweet is represented with the tweet vector. The next step of the system is to measure users' influence in the Twitter stream. To do so, PageRank[5] algorithm is applied in a graph G(V, E), with nodes V representing the Twitter users and edges E the Twitter following relationship among the users (i.e., one user is following the other). The strength of each keyword is calculated as the sum of weights for every tweet containing that keyword combined with the user authority of that tweet defined as nutrition and given in the following equation:

$$nutr_k^t = \sum_{\mathbf{tw_j} \in TW_k^t} w_{k,j} * auth(user(\mathbf{tw_j}))$$

where TW_k^t are the tweets of keyword k in t interval and auth(user) is the PageRank value of a user. The time interval is set to 15 days. In order to separate the commonly popular keywords from emerging keywords, they use as a measure of emerging influence of a keyword the difference of keyword nutrition between successive time intervals. The authors define the measurement of emergence of keyword as **energy** using the parameter **s**, which is set in their experiment to 2 days (generally variable s should be less than the time interval t).

$$energy_k^t = \sum_{x=t-s}^{t} \left(((nutr_k^t)^2 - (nutr_k^x)^2) \cdot \frac{1}{t-x} \right)$$

Consequently, for selecting keywords as emerging, the authors introduce the *critical drop* as the average of the energies of all keywords. Next, two approaches are proposed. In the first, each keyword with *energy* over the computed average is considered as emergent. In the second, the keywords are ranked in descending order according to their *energy* values and a maximum drop between the ranked energies is computed. For the keywords above the maximum drop, an average drop between pairwise energies is computed and the first higher energy drop in this list is called critical point. Keywords that are ranked better than this point are considered emergent. The final step is to create topics from emerging keywords. A topic is defined as a minimal set of semantically related keywords. In doing so, a keyword topic graph $TG(K^t, E)$ is performed where K^t is a set of vertices containing all captured keywords, while the edges between a pair of keywords reveals the correlation between two nodes. The correlation among two keywords z and k is related to the set of documents containing both terms. Given a keyword k that represents a node in topic graph TG^t, they find the set of vertices S reachable from k through a path using Depth First Search (DFS) algorithm. Furthermore, they repeat the process with reverse edges in order to find the set of vertices T that can access the node k with a path using DFS. The vertices (keywords) within S and T form the final topic.

In Budak et al.,[2] the authors propose a novel method for identification of significant topics in social networks, which takes into account network topology. They introduce social network structure into trend analysis. In fact they named the derived topics as structural topics. Structural is a topic that

is popular within different subgroups of a network. They present two alternate definitions for emerging topics, which are *coordinated* and *uncoordinated* topics. In the former, the number of connected peers (users) discussing a topic is considered as a measure of trendiness of the topic. In the latter, the score of a topic is based on the number of unrelated (unconnected) people who are interested in it. In contrast, the traditional definition about topic trendiness is the total number of people who discuss the topic inside the network. The combined class of coordinated and uncoordinated trends is referred to as structural trends. The problem of structural trend identification in Twitter graph $G = (V, E)$ (with V representing Twitter users and E the Twitter mention relationship among users) is considered as an information diffusion maximization problem with probability p denoting that user n_i talks about topic T_x independently from any of its neighbors and the probability q that user talks about a topic that another user in the neighborhood also mentions. The proposed model is an extension of the independent cascades model, called Independent Trend Formation Model (ITFM). In order to evaluate the significance of structural trends, the authors model the process of trend diffusion using ITFM. In their experiments, they compare the results of structural trends against the traditional trends and try to reveal the nature of the detected structural trends.

Mathioudakis and Koudas[6] propose *Twitter Monitor*, a system that performs trend detection in Twitter stream in three steps. In the first step, their system detects bursty keywords; keywords that suddenly appear in a Twitter stream with unusually high frequency. The second step of the system groups those keywords into trends based on their co-occurrences in the stream. In their approach, a trend is a group of bursty keywords that co-occur frequently in the same tweets. The algorithm, called *Queue burst*, is a one-pass algorithm, meaning the data stream is read only once. Moreover, the identification is performed in real time and the method is adjustable against spurious bursts. In some cases, a keyword may appear in a short period of time by coincidence in many tweets. The system is tuned to avoid reporting such words as bursty keywords. Another system characteristic is that the system is adjustable against spam. Spam users repetitively generate large numbers of similar tweets. The system is tuned to ignore such behavior. The second step of the system is the implementation of *Group-Burst* algorithm that groups bursty keywords into trends. For this purpose, the history of tweets is retrieved for each burst keywords and keywords that are found to co-occur frequently in a large number of tweets are categorized in the same group. In the third step, the Twitter Monitor system implements context extraction algorithms such as Singular Value Decomposition (SVD) in order to detect correlated words in recent history and expand trend vocabulary. Finally, a chart is produced for each trend that depicts the evolution of its popularity over time.

Leskovec et al.[7] attempt to capture new topic ideas and memes shared through social and mainstream media. Their focus is to find the persistent and novel temporal patterns in the news cycle. One significant observation derived from their study is the existence of 2.5 hr lag between the peaks of attention of trend topics of mainstream media and blogs, respectively. The first step of the proposed approach is to cluster phrases from the corpus of articles into relative clusters. They use the term "item" to define every article in the corpus and with the term "phrase" a quoted string that occurs many times in articles. Their aim is to cluster all the phrases that occur in a corpus into distinct phrase clusters. To do so, they conduct a phrase graph where each node represents a phrase from the corpus and each edge in the graph corresponds to the semantic relation between every phrase. Every edge (p, q) from the nodes p, q correspond to two phrases with the restriction that p has lower word length than q and the semantic distance between p and q be less than a threshold (number of co-occurring words). That semantic distance is related to word concurrences between the pair. Finally, a directed acyclic graph (DAG) is constructed since all edges point from shorter phrases to longer phrases and a phrase clustering is performed. In order to identify phrase clusters in the phrase graph (which is called DAG partitioning), they try to eliminate the nodes with low weights, which correspond to the nodes that connect unrelated subgroups of the graph. The following problem is a well-known optimization problem called *multiway cut* problem. An approximation of DAG partitioning is implemented in the proposed approach. The final step of the proposed approach is the temporal analysis of the extracted phrase clusters that captures the dynamics of news cycles both globally and locally. In the global analysis, they try to formulate a model for the news cycle capturing: a) the imitation between sources; and b) the recency (the decrease of popularity over time); with $f(n_j) \cdot \delta(t - t_j)$ representing the two components, where n_j denotes the number of items related to the thread j, t the current time, and t_j the time when j was produced. The $\delta(\cdot)$ component is monotonically decreasing and $f(\cdot)$ is monotonically increasing. For example, one possible choice is $f(n_j) = (a + bn_j)^{\gamma}$ and $\delta(\cdot) = t^{-1}$. Finally, in the local temporal analysis, they try to model the dynamics around the peak of news cycle, which is found to be a combination of exponential and logarithmic functions.

In Table 1, the examined social media site, the crawling duration, the corpus size, and the evaluation approaches are depicted for all methods analyzed in this section.

EVENT DETECTION IN SOCIAL STREAMS

The term "event" is defined in the literature as a social activity or a phenomenon that happened in real life at some point in time and in specific place, either *planned* or *abrupt*. A system that could identify social events and their associated

Table 1 Trend detection approaches

Reference	Social media	Crawling duration	Corpus size	Evaluation
[3]	Flickr	Several studies	–	Mean absolute square error
[4]	Twitter	April 13–28, 2010	3 M tweets	Energy value
[2]	Twitter	7 months	20 M users 467 M tweets	Average precision
[6]	Twitter	Online	10 M tweets/day	Online interface
[7]	Blogspot.com	August 1 to October 31, 2008	90 M articles (from 1.65 M blogs)	Temporal analysis

user-based social media information could improve browsing and searching in these media and help users to navigate better by filtering the noisy information. Users tend to post in social media updates about their daily life and news, which includes social events such as concerts, athletic events, exhibitions, as well as disastrous phenomena such as earthquakes, fires, tornadoes, etc. However, due to the large amount of messages in social streams, it is not straightforward to analyze and extract such meaningful information. When an event is occurring, the relevant messages are usually buried by a majority of irrelevant messages. Thus, it is crucial to mine the useful information from the social streams so as to provide navigational means for exploring such content. In this entry, we present state-of-the-art approaches for that task.

In order to understand the way in which event detection systems function, we have to portray the basic features of events in social media. Events, a) are at most times massive (a great number of users talk about them); b) have a great influence on user's daily life (thus users share information about their experience during an event); and c) have both spatial and temporal regions so that real-time location evaluation is possible. Event detection algorithms aim to discover such real-time event occurrences from the large and noisy social media streams.

There are several challenges arising when developing such a system. The first is to deal with the massive amount of data arriving per minute. The second is to classify data into potentially millions of events. Another is to deal with the fact that the set of events that we assign data items to is constantly growing and changing. Moreover, spam handling is very important in such dynamic streams. Fake or misleading multimedia content and its distribution through social networks constitutes an increasingly crucial and challenging problem, especially in the context of emergencies and critical situations as for example when an earthquake or a typhoon takes place. Finally, event detection algorithms should manage to separate unimportant personal updates from real-life events. In the majority of the event detection algorithms, it is assumed that all the documents from data streams are in some way related with a number of undiscovered events. However, in social media, this is not exactly the case, because most users update documents that are not related with some important real-life event but with "useless" personal updated information. Fig. 2 presents an overview of social event detection components. The figure does not follow a clear flow of the information between components since this is part of the design decisions a social event detection algorithm should take.

Depending on the extracted features that are used, event detection techniques are classified into the following categories: a) approaches that try to detect events from *text-based* content; b) approaches that try to detect events from *visual-based content* like photos or videos; and c) approaches that detect events using *metadata* information

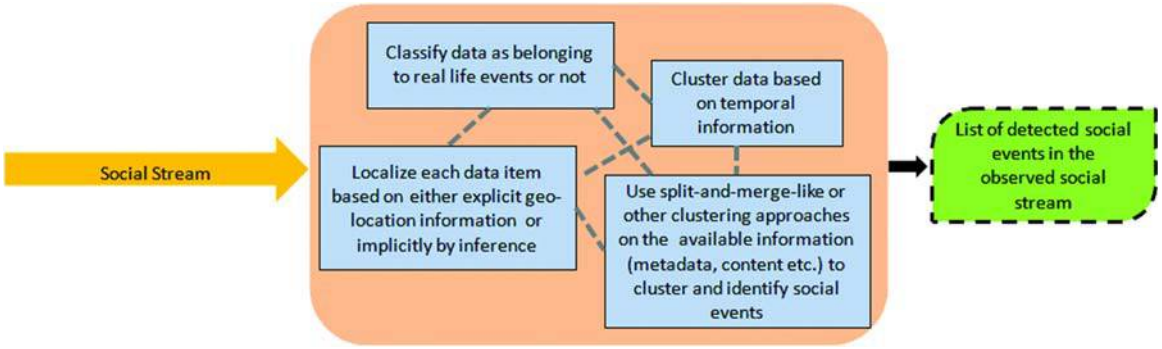

Fig. 2 A high-level overview of typical processes that take place in social event detection algorithms.

like tags geolocation. The text-based techniques rely in most cases on NLP techniques combined with machine learning methods in order to extract linguistically motivated features such as LDA[1] or LSI. Visual content-based techniques apply several techniques from the fields of computer vision, machine learning, and scene categorization to extract useful information about the relation of images with events. Depending on the detection method, event detection techniques are casted as *clustering* or *classification* techniques. On the former, clustering-based approaches attempt to discover distinct groups of information in the data. The scope of clustering approaches is to cluster all social media information to events (for instance, every tweet in an event), whereas on classification-based techniques, a database is used in order to train a system that will be able to detect whether or not an event is taking place. Furthermore, when the number of events is known beforehand, classification techniques could be applied to categorized documents to the specified events.

Depending of the type of event, these techniques could be grouped into either *planned* or *abrupt*. In the case of abrupt events, we can classify disasters such an earthquake or a tsunami, whereas in planned case, we can classify events that have been programmed before they took place, like Wall Street Occupation, sports events such as world cup final, elections, etc. Since it is not possible to avail prior information about abrupt events, such abrupt event detection techniques rely on the temporal signal of social streams to detect the occurrence of a real-world event. These techniques typically require monitoring of abnormal topics or bursts of a topic in streams, grouping the features with identical trend into events, and ultimately classifying the events into different categories. On the other hand, the planned event detection relies on specific information and features that are known about the event, such as a venue, time, type, and description, which are provided by the user or from the event context and aim to identify whether a datum belongs to the specific event or not.

Becker et al.[8] the authors define event identification as a clustering problem and propose a method for learning similarity metrics for social event identification. Their problem, as it is formulated, is to identify documents that refer to a specific event from the social media data. Those documents are derived from Flickr network. To do so, they create a distinctive representation for every document and apply a document similarity in order to cluster and detect events. For every document, they use as features the name of the user that creates the document, the title and the name of the document, a short description that summarizes the paragraph contents, a set of tags describing the document content, and finally time and location that the document was published. The above context features provide complementary cues for deciding when documents correspond to the same event, since using all features collectively provide more reliable evidence than using individual features. In order to be able to use similarity metrics, the authors transform their textual features to a TF-IDF weight vector and use *cosine* similarity as similarity metric. Traditional text processing steps such as stop-word removal and stemming are implemented. As a temporal distance between two items, the following metric is used: $1 - ((|t_1 - t_2|)/y)$ where t_1, t_2 are the timestamps of two items. For spatial distance between two items, the Harversine formula between two pairs of longitude and latitude coordinates is used.

In order to cluster the derived features into an event, a single-pass incremental clustering is proposed. Incremental clustering considers each document in turn, and determines the suitable cluster assignment based on the document's similarity function to any existing cluster. Moreover, the use of a threshold m is proposed. If there is no cluster with similarity against the document greater than m, a new cluster is generated. Otherwise the document d is assigned to the predefined clusters. To tune the threshold m, a training dataset is used and exhaustive search regarding parameter m is applied in order to achieve the best clustering performance measured by normalized mutual information (NMI) and B-Cubed algorithms. The scope of the approach is to cluster several documents to events by combining the several modalities using the incremental clustering and compute a similarity metric that combines all modalities either with a classification or with an ensemble-based clustering technique. In the former, SVM[9] classifier is applied in order to learn the similarity between pairs of documents using as input features the similarity between documents for every modality (text, location, and time information). The classifier is used as the similarity metric in order to cluster all features to events, whereas in the latter, an ensemble of clusterers is applied in order to combine all modalities. For every modality, an incremental clusterer is applied and the threshold m is tuned correspondingly. Finally, the incremental clustering is applied in order to cluster the output of all clusterers to several events. The distance metric of the ensemble clustering is related with the NMI and B-Cubed scores, which are calculated (in the process of parameter m tuning) for every modality.

Weng and Lee[10] propose a method called event detection with clustering of wavelet-based signals (EDCoW), which constructs a signal for each word in Twitter stream corpus and use a wavelet analysis in order to detect bursts in the signal. Frequently recurring bursts are filtered using their autocorrelation. The remaining signals are cross-correlated and clustered using a modularity-based graph partitioning of the resulting cross-correlation matrix. The four main components of the EDCoW system are: a) signal construction; b) cross-correlation; c) modularity graph partitioning; and d) measurement of event significance. In the first stage, the signal construction for every word is based on its TF-IDF score for several discrete time intervals. Subsequently, frequency domain metrics are implemented in order to calculate the final signal. The metric used is the Shannon wavelet entropy in every discrete interval. Once the construction of signal is performed, cross-correlation

is applied in order to measure the similarity of signals and create the correlation matrix of all words. EDCoW detects events by grouping a set of words with similar patterns of burst. To do so, cross-correlation can be viewed as adjacency matrix of graph G = (V, E, W) with V representing the signals, E the edges in adjacency matrix, and the weight W the cross-correlated similarity of the signals. Next, modularity-based graph partitioning is performed in the adjacency matrix to cluster all signals to events. Finally, a measurement of event significance is computed to define the importance of every cluster and differentiate the big events from trivial ones. The significance score of events is based on: a) the number of words and b) the cross-correlation among the words relating to the event.

Sakaki et al.[11] consider Twitter users as sensors and tweets as sensor information. They assume that a user, acting as a sensor, detects a target event and makes a report about it in Twitter. The work presents results on data collected using the Twitter API with keywords about earthquakes and typhoons every t seconds. The proposed model is constructed in three steps. In the first step, an SVM classifier decides whether a tweet is related to an event or not. In the second, a temporal analysis of the tweets is performed to estimate a waiting time for raising an alarm. Finally, in the last step, the location information of each tweet is used to calculate an estimate of the earthquake center or the trajectory of the typhoon.

The features used in the classifier of the first step are: a) the number of words in a tweet message; b) the position of the query word within the tweet; c) the full set of words from the tweet; and finally d) the words before and after the keyword in the tweet. The authors perform temporal analysis and observe that the number of tweets over time for the crawled data follow an exponential distribution of events. In their temporal analysis, the parameters of the exponential distribution are estimated from historical data and then used to calculate a reliable wait time before raising an alarm. Finally, for the spatial estimation step, a Kalman filter or particle filter is used.

Yin et al.[12] developed a system that aims to extract situation awareness information from Twitter. The proposed system detects bursts of words from the text data, by using a binomial distribution to model and estimate the number of tweets that contain a specific word. If the actual number of word occurrences is higher than the estimated number, then the word is categorized as bursty. Next, a classifier is built in order to automatically detect tweets that contain information about the impact of a disaster on the infrastructure such as roads, bridges, railways, etc. In the experiments, the authors examine both support vector machines and naive Bayesian classifiers with SVM. In order to discover important and emerging topics, an online incremental clustering algorithm[8] is applied on the burst items. In contrast with Becker et al.,[8] here, there is only one modality (the TF_IDF vector from the tweet) and the only parameter that has to be tuned is the clustering threshold m (tuned empirically). As similarity measure for clustering, the best results are given using the Jaccard similarity:

$$\mathrm{sim}_{\mathrm{jac}}(T,\ C) = \frac{|v_i \cap v_j|}{|v_i \cup v_j|}$$

Petkos et al.[13] propose a methodology for clustering multimedia content as social events from social multimedia sites such as Flickr. As the authors state, the case of detecting events from multimedia content is challenging due to the heterogeneity and the multimodality of the content itself. Since these collections are typically accompanied with rich metadata information along with visual descriptors, a multimodal approach fits well. The proposed methodology aims to compute "same cluster" relationships between items of the collection using the similarities of all available modalities. First, in a dataset of images that need to be clustered, the pairwise distance matrix between all items for every modality is calculated. A classification step is performed in order to determine whether two images belong to the same category. The matrix of pairwise distance between items is transformed to a pairwise similarity indicator matrix via the classification step. Finally, k-means clustering or spectral clustering is applied on this indicator matrix to cluster every image in an event. The NMI metric is used to measure the performance based on the available ground truth. The merit of their clustering approach is that there is no need for designing a fusion strategy for the several modalities.

Rafailidis et al.[14] present a data-driven approach to detect social events. Their proposed methodology takes into account that the collected social multimedia contain noisy metadata, with missing values or possible errors in their metadata descriptions. As such, they consider building initial clusters from content that contains spatial metadata, while creating singleton events for content with missing spatial information. Next, a single-pass procedure is followed to split the created clusters based on temporal information and create "must-link" sets of data (with fixed spatiotemporal information) named anchored clusters. The intercorrelations between anchored and singletons or among singleton clusters are computed to merge them into clusters. Finally, the remaining singleton clusters (single content objects with missing spatial information) are merged to form new clusters if their intercorrelations are over a given threshold. The intercorrelation between clusters is computed as the aggregated similarity from the different supported modalities, i.e., user descriptions, content titles, visual features, and sets of tags. Fig. 3 presents the computational steps for the proposed approach.

In Tables 2 and 3, we present characteristics from all proposed approaches and from the evaluated experiments. It is important to mention here that most approaches consist of a combination of several features and techniques.

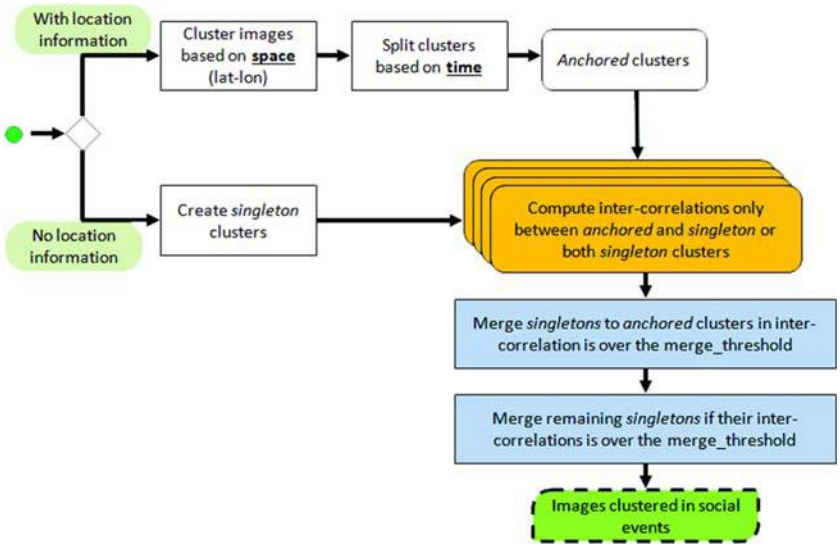

Fig. 3 The process of social event detection as described by Rafailidis et al.[14]

INFLUENCE DETECTION IN SOCIAL STREAMS

Influential people or *opinion leaders* are the individuals who spread the information faster and/or affect other peoples' behavior, inside and beyond their social communities. The influence can be defined as the ability of an individual to drive other people to action, as a consequence of personal behavioral interactions, and as such it reflects the user's authority and prestige inside a social network. The application of such knowledge that significantly helped the field gain its popularity is in the marketing and business domains. However, identifying and following the updates of those users is also a means of summarizing information about a community's topics and interests. By targeting those users who are considered influential in the social network, a marketing campaign will be more effective due to rapidly diffused information through authoritative entities. For example, when a new music album is released, music promoters engage those social media users who are influential in the music topic (community) and especially on the specific genre to potentially influence other users to purchase that album. A typical approach for algorithms of influencers detection is sketched in Fig. 4.

The simplest approach for calculating the influence of users inside a social community is to apply as a metric the number of followers or the number of friends that a user has in the social network. This number possibly indicates how many individuals may consume the content the user uploads. However, this is a rather naive metric since a large number of friends or followers in most cases correspond to celebrities' accounts or to well-known brands with no actual influence since there is no insight about user information diffusion inside the network. Alternative and more accurate measurements of influence in social media could be the actual propagation of user content through the network (the frequency with which followers consume user content), the novelty of user content, the quality of user content, or the frequency at which a user updates information inside social media. Influence identification approaches are classified into: a) approaches that use *heuristic* methods in order to measure user influence and rank users considering that heuristic methods such as retweets, mentions count, etc.; b) *centrality measures* such as betweenness centrality or PageRank; c) *influence maximization* approaches that try to maximize the influence diffusion inside the social graph; and finally d) *Trendsetters* approaches that try to locate early adopters.

Table 2 Characteristics of event detection algorithms

Reference	Abrupt	Planned	Clustering	Classification	Visual	Text	Spatiotemporal
[8]	✓	✓	✓	✓		✓	✓
[10]	✓		✓			✓	
[11]	✓	✓	✓	✓		✓	✓
[13]		✓	✓	✓	✓	✓	✓
[12]	✓		✓	✓		✓	✓
[14]		✓	✓		✓	✓	✓

Table 3 Event detection datasets and evaluation in discussed papers

Reference	Social media	Crawling duration	Corpus size	Evaluation
[8]	Flickr, Last FM	January 1, 2006 to December 31, 2008	9,515 unique events and 270 K photos Flickr 24,958 events and 594 K photos Flickr	NMI, B-Cubed
[10]	Twitter	June 2010	19,256 users 4 M tweets	Precision of EDCoW
[11]	Twitter	Twitter API Streaming	597 tweets for SVM training	F-score
[13]	Flickr	Mediaeval Challenge 2012	73,645 photos	NMI
[12]	Twitter	March 2010 to February 2011	66 M tweets and 2.51 M users.	Detection rate and silhouette score
[14]	Flickr	Mediaeval Challenge 2013	437 K images	F1-Score, NMI, DIV-F1

Heuristics provide a baseline for identifying some influencers; however, relying only on such approaches gives low quality of results. In the centrality methods, the most widely used are degree, closeness, betweenness centrality, and PageRank. On the other hand, influence maximization methods aim to find k influential nodes, i.e., those that maximize the information spread to the network. These methods attempt to model social influence through the process of information diffusion. The more influential a user is, the wider the information is spread in the network. The most common choices for the influence propagation models are independent cascade or linear threshold. Finally, a new set of studies appeared to study "Trendsetters," i.e., nodes that do not have high degree of centralities; however, they have high impact on other nodes by propagating innovative information. To be an innovator, a user should be one of the first users inside a social network to adopt a new trend. However, not all innovators are trendsetters since only few have the ability to propagate their information inside the network.

Weng et al.[15] proposed the TwitterRank algorithm aiming to identify influencers in Twitter. The authors claim the existence of homophily phenomenon in Twitter graph, and conduct experiments to support their statement. The experiments target the following questions: a) are Twitter users with at least one-directional relationship (i.e. the first user is following the second) more similar than those without? and b) are Twitter users with bidirectional relationship more similar than those without? Aiming to calculate the pairwise similarity of users, users' topics are extracted using LDA. The Jensen–Shannon divergence is used to measure the difference between the probability distributions of topics for a pair of users. Next, a graph D(V, E) with nodes V representing Twitter users and edges E representing the "following" relationship between nodes is constructed. The proposed model is a modified PageRank algorithm that combines the traditional random surfer with the topic similarity among nodes of the graph to identify topical influencers in the network.

Kempe et al.[16] consider influence detection as an influence maximization problem. The evaluation of the proposed technique is performed in the arXiv database, which contains scientific papers, their authors, and the coauthorship as a relationship for the pair of authors. A coauthorship graph is constructed using authors as nodes and the coauthorship as the edge of the graph. The proposed algorithm is performed in discrete time steps using two different influence models, namely, linear threshold model and independent cascade model. Each node of the graph in every step could be either active or inactive. An active node could possibly activate neighboring nodes in every time step. In the first model, each node u of the graph is influenced by each neighbor w according to a weight $b_{u,w} = (c(u, w))/d(u)$, where $c_{u,w}$ is the total number of coauthorship between nodes and d_u is the degree of node u. In order for a node u to be active in a step t, the sum of

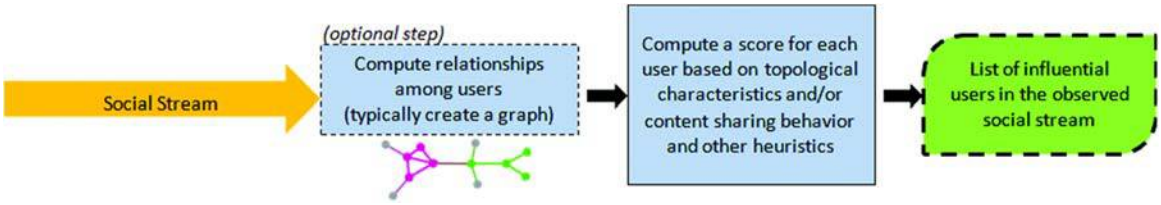

Fig. 4 Overview of a typical procedure for influencer's detection.

neighboring weights should be greater than the node threshold $\theta u \sim U[0, 1]$:

$$\sum_{w \text{ active neighbor } u} b_{u,w} \geq \theta_u$$

The process starts with an initial set of nodes A_o and stops when no more activation is possible. In case of independent cascades, the process starts again with an initial active set. If a node u is active in step t, the activation probability for neighbor node v in step $t + 1$ is $1 - (1 - p)^{c(u,v)}$, where p is a uniform probability. As in the case of linear thresholds, the process runs until no more activation is possible. The aim of the influence maximization is to find the best initial set A of k nodes that maximizes influence (i.e., maximize the number of active users). The list of top k influencers is composed of an influence submodular function f(S) for the above influence model, which is the expected number of active nodes. The problem is to find the k-element set S for which f(S) is maximized, which is considered as a constrained NP-hard optimization problem with f (S) as the objective function. In order to evaluate the performance of their approach, they compare the two approaches against heuristics based on nodes' degrees and centralities.

Another study on influence detection from the perspective of social diversity was performed by Huang et al.[17] In this approach, the authors build a Twitter graph $G = \{V, E\}$ with V used to represent Twitter users and E the relationship among the users. All the edges in E are associated with a transition probability TP(u, v) representing the probability that a user u is influenced by user v. In the case of retweet-following graph, user v is a follower of user u and has propagated some information of user u (with Twitter retweet feature). The transition probability for a pair of users is defined as

$$TP(u, v) = \frac{mp_{uv}}{\sum_{k \in IN(v)} mp_{wv}}$$

where the numerator stands for the propagated messages (mp) from user u to v, while the denominator sums up the number of received messages from all neighbors of user v. This definition captures how much attention a user could draw from its outbound neighbors IN(v). The next definition is the social diversity metric SD(v) for each user, which aims to measure how diverse a user v is within a network and is calculated as one divided by the number of clusters that v belongs to. In order to calculate the clusters that a node v belongs to, the star clustering algorithm is used. Finally, the diversity-dependent influence algorithm is defined as a combination of transition probability (TP), the social diversity measure (SD), and the PageRank random surfer model as in Weng et al.[15]

A work for locating early adopters in graph is presented by Saez-Trumper et al.[18] To identify important trendsetters there are two important factors, the former is the topic of innovation while the latter is the time when a user adopts an innovation. Traditional centrality measures do not consider the time constraint; instead they consider only statistics for a static network topology. The authors here define as $G_k = (N_k, E_k)$ the Twitter graph related to a topic **k**, with topic defined as the set of hashtags (trends) $\mathbf{k} =$ [#tag1, #tag2, …, #tagM]. The set of nodes N_k in the graph are all nodes that adopt at least one trend of topic k, and edges E_k represent all edges (u, v) such that $u, v \in N_k$. They define two vectors $\mathbf{s}_1$ and $\mathbf{s}_2$ for all $u, v \in N_k$:

$$\mathbf{s}_1(u)_i = \begin{cases} 1, & \text{if } t_i(u) > 0, \\ 0, & \text{otherwise} \end{cases}$$

and

$$\mathbf{s}_2(u, v)_i = \begin{cases} e^{\frac{-\Delta}{\alpha}}, & \text{if } t_i(v) > 0 \quad \text{and} \quad t_i(v) < t_i(u) \\ 0 \end{cases}$$

for $i = 1, \ldots, h_k$, where $t_i(u)$ is the time when node u adopts a trend $h_i \in k$, $\Delta = t_i(u) - t_i(v)$, and α is a control parameter (defined to one day, i.e., 86,400 sec). Vector $\mathbf{s_1}(u)$ informs whether a node u adopts the k trend and at what time, while $\mathbf{s_2}(u, v)$ shows whether u adopted a trend before v and weights the relation as a fraction of time that u, v adopt the trend. Finally, regarding influence of u over v (for topic k):

$$I_k(u, v) = \frac{I_k^*(u, v)}{\sum_{w \in \text{Out}G_k(u)} I_k^*(u, w)}$$

where $\text{Out}G_k(u)$ is the outcoming neighbor sets for node u and $I_k^*(u, v)$:

$$I_k^*(u, v) = \left(\frac{s_1(u) \cdot s_2(u, v)}{||s_1(u)|| \times ||s_2(u, v)||}\right) \times \left(\frac{L(s_2(u, v))}{n_k}\right)$$

Making an analogy with the random surfer model in the PageRank algorithm, they combine PageRank with the proposed influence measurement $I_k(u, v)$ in order to calculate authoritative users. The proposed algorithm is called trendsetters rank (TS).

In Fig. 5, a simple example for TS algorithm is illustrated. There are two tables: the first one contains the adjacency matrix of nodes v1, v2, …, v10 and connectivity between nodes. In the second table, the time when a node adopts a new topic k is illustrated. The node v8 is the first one that adopts a topic h, while node v1 is the last one that adopts the trending topic. Node v8 cannot be considered as an innovator since the information is passed only to node v9 and in the rest of the graph. Moreover, regarding PageRank, the node v3 is considered the top-ranked node because it diffuses the information to many nodes. According to PageRank, node v1 and v2 have the same rank. However, if the time information is considered, node v2 is the top trendsetter as it is the first that adopts the trend and it is followed directly or indirectly to many other nodes.

In the million follower fallacy,[19] the dynamics of Twitter user influence with respect to in-degree, retweets, and

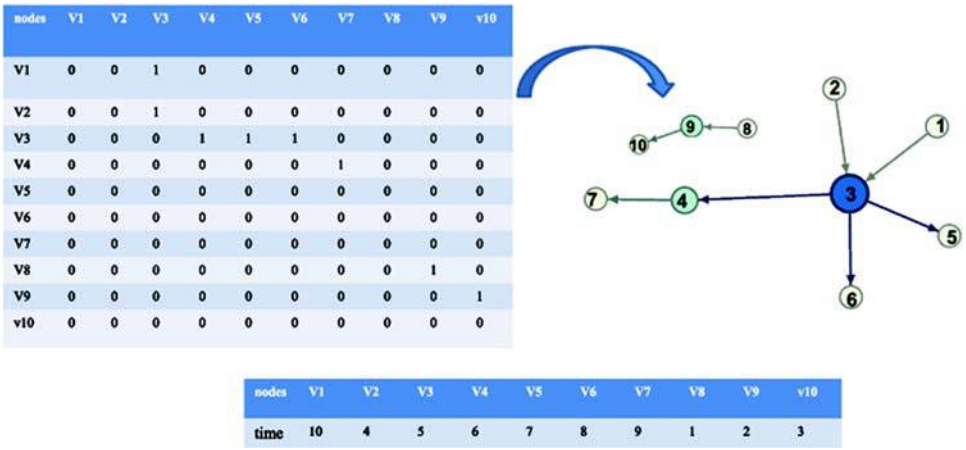

Fig. 5 An example for the TS algorithm of Saez-Trumper et al.[18]

mentions in Twitter streams are investigated. Their findings show that in-degree represents the popularity of a user, retweets represent the quality of tweets, and finally mention illustrates the value of a user name. Authors state that the most mentioned users in Twitter are celebrities while mainstream news organizations propagate a great deal of information and gain a high level of retweets over different topics. Their dataset consists of 80 M users with 1.9B social links. The authors calculated the Spearman's rank correlation among all users, the top 10% of the ranked users, and the top 1% of the ranked users. For the calculation with all users' ranking, they found that the correlation between measures is biased from the users with low in-degree, retweets, and mentions; thus it is not a reliable metric. Therefore, the authors calculated the rank list for top 10% and 1% ranked users and found high correlation among retweet and mention measure.

Furthermore, the influence dynamics across different topics and the variation of three measures mentioned above were discussed. More specifically, common topics users discussed in 2009 such as Michael Jackson's death, Iran election, and influenza H1N1 were investigated and the Spearman's rank correlation were calculated. Top-ranked users, with respect to mentions, showed strong correlation among topics. Lastly, a temporal analysis was applied to investigate how these measures change over time. To do so, they used the 233 all-time influential individuals who are top-ranked users in the three measures. They tracked their influence score over an 8-month period and the mention and retweet probability of users were studied as well as mention and retweet probability over three topics cited above.

In Table 4, the categorization of the discussed approaches in influence detection are depicted. Table 5 summarizes the datasets used in the discussed papers and the evaluation techniques used.

CONCLUSION

Navigation through unstructured and uncurated data in social streams has become a significant problem due to the large amount of data that users upload daily. The problem will continue to increase as the volumes are increasing exponentially and new data sources are introduced every day. As with the data creation boost by the mobile devices, yet another boost is foreseen with the Internet of Things. All these facts support the need for new algorithms and tools for navigation and browsing of online social content toward a specific goal each time. Based on the aims of the navigation, different algorithms have been proposed in the literature.

The trend detection algorithms focus on the identification of interesting and popular topics and themes within the social media streams. As such, topic modeling and

Table 4 Classification of the proposed techniques

Reference	Influence maximization	Centrality measures	Heuristics	Trendsetters
[15]	✓	✓		
[16]	✓		✓	
[17]	✓	✓		
[18]		✓		✓
[19]			✓	

Table 5 Characteristics of the proposed techniques

Reference	Social media	Crawling duration	Corpus size	Evaluation
[15]	Twitter	July 2006–April 2009	6,748 Twitter users and 1 M tweets	Kendall correlation score
[16]	ArXiv graphs	–	10,748 nodes and 53,000 pairs of nodes	Active size/target set size
[17]	Twitter	December 4–17, 2012	151,305 users, 75 K tweets, and 400 K retweets	Influence spread/rank users
[18]	Twitter	All data until August 2009	~50 M users and 1.6 B tweets	Kendall τ rank
[19]	Twitter	All data until August 2009	>50 M users, ~1.7 B tweets, and 1.9 B social links	Spearman's rank correlation

mining approaches are typically used to group users' posts. Moreover, a kind of prior distribution of the topics is required to identify the emerging ones. A known problem of the trend detection algorithms that need to be further investigated is the difficulty of detecting smaller (in content volumes) trends that are typically buried under the widely visible trends.

The social event detection group of algorithms aims to associate the social web posts with real-life events that happened in a certain time and location. As such, algorithms that belong to this group need to exploit any available implicit or explicit spatiotemporal information to place each post on a map. However, a prior step that is required in the cases that we have unfiltered content is to classify first whether the post is referring to a real-life event or not. As in the case of trend detection, social event detection algorithms also have problems identifying events that happen together with other larger events; however, the spatiotemporal information (when available) helps to drastically distinguish events and identify smaller localized real-life events.

Finally, the influencer detection group of algorithms aim to identify those social accounts that play a key role within a community of accounts and contribute greatly in the propagation of information, i.e., the creation of new trends or the filtering of content generated within the community. There are mainly two approaches that are taken in this group of algorithms. The first is to use the topological characteristics and the position of each user account in the community of the accounts, and the second is to track the content sharing behavior of each account to produce a final score. The combination of the topology and behavior tracking approaches has also been examined to perform well.

As a future challenge for the social web navigation, researchers should work toward algorithms that will enhance the user experience through guidance and dynamically supported navigation, in contrast to being intrusive or follow strict personalization models.

In this entry, we discussed the state-of-the-art approaches that intent to support efficient content browsing and navigation in social media aiming for the detection of trends, social events, and influential users. The multimodality of the shared content and the different user intentions in browsing social content as well as the unstructured forms and the big amounts of content demand the usage of heterogeneous approaches and a large variety of features and methodologies to identify the targeted content.

ACKNOWLEDGMENT

This work was partially supported by the EU FP7-funded project LASIE (Large Scale Information Exploitation of Forensic Data)—nr. 607480 (http://www.lasie-project.eu/).

REFERENCES

1. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent Dirichlet allocation. J. Mach. Learn. Res. **2003**, *3*, 993–1022.
2. Budak, C.; Agrawal, D.; El Abbadi, A. Structural trend analysis for online social networks. Proc. VLDB Endowment Homepage Archive, **2011**, *4* (10), 646–656.
3. Jin, X.; Gallagher, A.; Cao, L.; Luo, J.; Han, J. The wisdom of social multimedia: using Flickr for prediction and forecast. In Proceedings of the International Conference on Multimedia, Washington, D.C., 2010; ACM: New York, NY, 2010.
4. Cataldi, M.; Di Caro, L.; Schifanella, C. Emerging topic detection on Twitter based on temporal and social terms. In Proceedings of the Tenth International Workshop on Multimedia Data Mining, Washington, D.C., 2010; ACM: New York, NY, 2010.
5. Page, L.; Brin, S.; Motwani, R.; Winograd, T. The PageRank Citation Ranking: Bridging Order to the Web; Stanford InfoLab, 1999; 1999–66.
6. Mathioudakis, M.; Koudas, N. TwitterMonitor: trend detection over the Twitter stream. In Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data, Indianapolis, IN; ACM: New York, NY, 2010; 1155–1158.
7. Leskovec, J.; Backstrom, L.; Kleinberg, J. Meme-tracking and the dynamics of the news cycle. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge

Discovery and Data Mining, Paris, France; ACM: New York, NY, 2009; 497–506.
8. Becker, H.; Naaman, M.; Gravano, L. Learning similarity metrics for event identification in social media. In Proceedings of the Third ACM International Conference on Web Search and Data Mining, New York, NY; ACM: New York, NY, 2010; 291–300.
9. Cortes, C.; Vapnik, V. Support-vector networks. Mach. Learn. **1995**, *20* (3), 273–297.
10. Weng, J.; Bu-Sung, L. Event detection in Twitter. In Proceedings of the Fifth International Conference on Weblogs and Social Media; The AAAI Press: Barcelona, Spain, 2011.
11. Sakaki, T.; Okazaki, M.; Matsui, Y. Earthquake shakes Twitter users: real-time event detection by social sensors. In Proceedings of the 19th International Conference on World Wide Web, Raleigh, NC; ACM: New York, NY, 2011; 851–860.
12. Yin, J.; Lampert, A.; Cameron, M.A.; Robinson, B.; Power, R. Using social media to enhance emergency situation awareness. IEEE Intell. Syst. **2012**, *27* (6), 52–59.
13. Petkos, G.; Papadopoulos, S.; Kompatsiaris, Y. Social event detection using multimodal clustering and integrating supervisory signals. In Proceedings of the 2nd ACM International Conference on Multimedia Retrieval, Hong Kong, China; ACM: New York, NY, 2012; 231–238.
14. Rafailidis, D.; Semertzidis, T.; Lazaridis, M.; Strintzis, M.; Daras, P. A data-driven approach for social event detection. In Proceedings of the MediaEval 2013 Multimedia Benchmark Workshop; Larson, M., Anguera, X., Reuter, T., Jones, G.J.F., Ionescu, B., Schedl, M., Piatrik, T., Hauff, C. & Soleymani, M., Eds.; Barcelona, Spain, October 18–19, 2013; Volume 1043 of CEUR Workshop Proceedings, CEUR-WS.org, 2013.
15. Weng, J.; Ee-Peng, L.; Jing, J.; Qi, H. TwitterRank: finding topic-sensitive influential twitterers. In Proceedings of the Third ACM International Conference on Web Search and Data Mining, New York, NY; ACM: New York, NY, 2010; 261–270.
16. Kempe, D.; Kleinberg, J.; Tardos, E. Maximizing the spread of influence through a social network. In Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, D.C., 2003; ACM: New York, NY, 2003; 137–146.
17. Huang, P.Y.; Liu, H.Y.; Chen, C.H.; Cheng, P.J. The impact of social diversity and dynamic influence propagation for identifying influencers in social networks. In International Conference of Web Intelligence WI IEEE, Atlanta, GA, November 17–20, 2013; IEEE: Piscataway, NJ, 2013.
18. Saez-Trumper, D.; Comarela, G.; Almeida, V.; Baeza-Yates, R.; Benevenuto, F. Finding Trendsetters in information networks. In Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Beijing, China; ACM: New York, NY, 2012; 1014–1022.
19. Meeyoung, C.; Haddadi, H.; Benevenuto, F.; Gummadi, K.P. Measuring user influence in Twitter: the million follower fallacy. In Proceedings of International AAAI Conference on Weblogs and Social Media, Washington, D.C., 2010; AAAI: Palto Alto, CA, 2010.