Topic Detection in Twitter Using Topology Data Analysis

Pablo Torres-Tramón(✉), Hugo Hromic, and Bahareh Rahmanzadeh Heravi

Insight Centre for Data Analytics, National University of Ireland, Galway, Ireland
{pablo.torres,hugo.hromic,bahareh.heravi}@insight-centre.org

Abstract. The massive volume of content generated by social media greatly exceeds human capacity to manually process this data in order to identify topics of interest. As a solution, various automated topic detection approaches have been proposed, most of which are based on document clustering and burst detection. These approaches normally represent textual features in standard n-dimensional Euclidean metric spaces. However, in these cases, directly filtering noisy documents is challenging for topic detection. Instead we propose Topol, a topic detection method based on Topology Data Analysis (TDA) that transforms the Euclidean feature space into a *topological space* where the shapes of noisy irrelevant documents are much easier to distinguish from topically-relevant documents. This topological space is organised in a network according to the connectivity of the points, i.e. the documents, and by only filtering based on the size of the connected components we obtain competitive results compared to other state of the art topic detection methods.

1 Introduction

Social Network Sites (SNS) are one of the most important communication channels nowadays. SNS users interact with one another generating a considerable amount of content of various media types such as text, images or videos. This content has the potential of reaching a very wide audience, where feelings, political opinions or breaking news can be transmitted. One particular kind of SNS are *microblogging* sites, where messages are constrained and normally rather short. Twitter is the prime world-wide example of a microblogging system. In this environment, information is shared and circulated faster than in more conventional SNS such as blogs or forums, reaching a large audience in a shorter time.

Real-world events have shown the key role of microblogs for spreading news and supporting the information flow between communities in the social sphere. For example, the Mumbai 2008 bomb blasts, the 2011 crash of the US Airways Flight 1549, the Arab Spring movements, and the Boston Marathon bombing were all very important global events where social media played a crucial role in reporting and covering the news [9]. In such situations, users acted as *real-life sensors* [5], reporting what was happening nearby and posting information almost in real-time. All of this content can be mined in order to explore and monitor real-world events. In particular, we are interested on detecting related

F. Daniel and O. Diaz (Eds.): ICWE 2015 Workshops, LNCS 9396, pp. 186–197, 2015.
DOI: 10.1007/978-3-319-24800-4_16

topics inside the context of a larger story. We want to identify related stories that may not have been previously considered, and hence enrich the main story itself. This use case is key within a journalism context where the journalist is concerned about all the details for a particular story [1].

Numerous research studies have been conducted to create methods that automatically detect topics in real-world events such as government crises [15], natural disasters [18] or political elections [6]. Most of these methods use ranking or clustering to determine whether a topic is of relevance or not. Clustering, for instance, requires defining a linkage strategy and a series of thresholds to select candidates. A similar situation occurs for ranking-based methods because they need to select a subset of the highest candidates. Even if clustering and ranking approaches are suitable and have good results for a large number of use cases, they often require to repeatedly train the model when facing new data in order to calibrate the thresholds. This requirement makes topic detection methods too rigid for the context of breaking news analysis in microblogs systems.

In this paper we propose Topol, a novel unsupervised method for detecting topics in Twitter data based in Topology Data Analysis (TDA). The fundamental goal of TDA is to recognise shapes or patterns present within the data [14]. TDA defines a coordinate system, the *topological space*, generated using a *distance* function and transforms the input data so that this new space does not consider coordinates but distances instead. The central idea of topology analysis is the fact that it allows studying the properties of data shapes that are invariable under small deformations [14]. In addition, TDA also allows to study different perspectives of the same data. Our solution *explores the shapes* formed by Twitter data represented as a network of overlapping clusters, and uses this graph to determine underlying topics. Our intuition is that major topics are concentrated on large and densely connected components within this network. On the other hand, noisy topics are represented as small and isolated groups of nodes.

In our experiments, Topol shows to be competitive compared to state of the art topic detection methods for the same use case. While current approaches rely mostly on clustering and filtering techniques, our method identifies topics and generates their descriptions only from the shapes of the data alone, according to the constructed topological spaces using TDA.

The remainder of the paper is organised as follows. In Section 2 we provide a brief description of the Twitter topic finding problem that we address in this work. Section 3 discusses current state of the art methods for the above task. In Section 4 we describe the general TDA approach and in Section 5 we introduce our algorithm, Topol. Section 6 describes the experiments and results, and Section 7 concludes the paper and provides future interesting directions for our research. Because we focus on the Twitter context, from now on we will use the terms *tweets*, *post*, and *documents* interchangeably.

2 Problem Description

We address the problem of topic detection in Twitter. This task can be defined as identifying prominent topics in a document corpusunder a User-centred scenario [2], where the documents in this case are *tweets* posted in Twitter. Since Twitter data is continuously generated, the aforementioned document collection is then inherently stream-based and suitable approaches require constantly updating their output according to newly arriving data items in order to incorporate the latest changes, i.e. new tweets being created.

One mechanism to handle streams of documents is the usage of sliding windows techniques. This scheme defines an *update rate*, which in turn creates *time slots* as the time period between each update. The value for this update rate parameter is dependent on the nature of the event under consideration. For example, if the event continues for a few minutes the time slots should be small, but in contrast, if the event lasts for days, the time slots period should be larger. We then refine the topic finding problem to identifying topics in each of those time slots or windows. Furthermore, we represent the discovered topics as a list of keywords and a satisfactory detection of topics should bring the most representative keywords for each of them.

The content of tweets normally includes a wide range of subjects, such as personal feelings, political opinions, breaking news information, spam or comments. Such variety imposes difficult challenges and complexities for the topic detection task. In order to frame the experiments described in this paper, the input data is narrowed down by predefining a set of keywords such that every tweet must content at least one of those keywords. This a priori information is considered to be provided by the end-user and the keywords are assumed to be highly related to some event of interest for studying.

3 Related Work

Topic detection on large streaming data, such as Twitter, gained notorious interest by researchers in the last few years. There are two main branches of approaches: (1) *document-pivot* where the topics are identified from the documents and (2) *feature-pivot* where the topics instead are generated according to relations found in a diverse range of features.

An example of a document-pivot approach can be found in [16]. The authors address topic detection in Twitter by clustering documents (tweets). Because generating clusters is a time-consuming task, they implemented a more efficient method by using Local Sensitive Hashing (LSH). This improvement allows to find the nearest clusters for a new document in constant time, dramatically reducing the computational effort for document comparison. Additionally, in order to reduce non-relevant topics, they established *threads* of topics such that each thread corresponds to the evolution of a particular topic across time. This information is used to filter out non-interesting topics. However, this method still is a form of clustering and hence it suffers from data fragmentation. In contrast,

Topol groups documents together according to the connections present in the feature spaces, found by repeatedly sampling the tweets being analysed.

Feature-pivot methods rely on finding associations in a subset of defined features. The goal of these approaches is to (1) reduce the computation time by considering only a subset of features and (2) improve the topic detection results by only using a higher score for this subset of features. Several strategies have been proposed to *identify a suitable subset of features* such as probabilistic models [7], ranking [20] or Wavelet analysis [22]. Once features are selected, they are *analysed* in order to extract associations to later build topics. For this there are several techniques as well, such as clustering [8], ranking [1] or noise reduction [10].

Selecting feature subsets contribute to reducing non-relevant topics, but also create a bias in the final output depending on the selection criteria. Our algorithm, Topol, does not require selecting features but instead studies the topological shapes of the data directly according to a chosen similarity function.

It is also common to find a combination of strategies [1,10]. In general, feature-pivot approaches tend to generate misleading correlations between features and found topics that in reality are not associated with any event of interest [3,11].

Finally, Sayyadi et al. [20] proposed a similar approach to Topol. The authors represented the document features in a graph of keywords where nodes are terms and the links between them are the co-occurrence degrees of two terms in the tweets. Afterwards, topics are determined by community structures within this graph. This method differs from our approach since Topol builds a network according to a certain distance function instead. This particularity makes it possible for one feature to co-occur in several documents but, if they are not *close* in a topological sense, they will be not associated with the same topic.

4 Topology Data Analysis (TDA)

In many topic detection approaches, a text stream is traditionally represented using a vector where each feature corresponds to one coordinate in a Cartesian system. The similarity (or dissimilarity) of those vectors is defined using a distance function such as the well-known Euclidean-distance or Cosine-similarity functions. Moreover, this distance function is assumed to be continuous for all text streams, which means that it is always possible to define a distance between any two documents. However, these assumptions are far from being realistic in most real-world use cases.

For the above reason, instead of assuming a Cartesian system, it is preferable to study the data without considering the raw underlying metric space, therefore reducing the background noise embedded within this coordinate system. For this purpose, we use Topology Data Analysis (TDA) to generate representations of the data that allow us to study the inherent invariant shapes within this data. TDA is rooted on the field of *Topology*, which is the branch of Mathematics that deals with *qualitative* instead of quantitative information [4]. In addition,

Topology is coordinate-free, which means it studies the geometric properties of the data without depending on any particular coordinate system, and uses the notion of *infinite nearness* instead of a distance function.

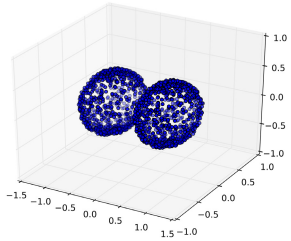
In this work, we employed the MAPPER algorithm [21] to generate topological representations of Twitter data. This method is based on a generalised version of Reeb graphs [17]. MAPPER, as suggested by its name, applies a *mapping function* to construct a network-based representation of the input data points. This input is first valued according to a *distance function*. The algorithm iteratively samples the constructed distance matrix in small subsets of points that are evaluated by a *filter function*, whose image is further divided into intervals that are related to those subsets.

The aforementioned distance function is the core mathematical tool that characterises each point in the feature space. The interval size parameter, called the *resolution* and denoted as r_p, is variable. With bigger intervals a more general vision of the data can be obtained. On the other hand, if the intervals shrink, the generated output is built according to the smaller shapes of the input data. It can be noted that this size parameter determines the amount of intervals used by the filter function. The points assigned to the same interval can be considered as partial *clusters*, which later corresponds to a node in the output network.

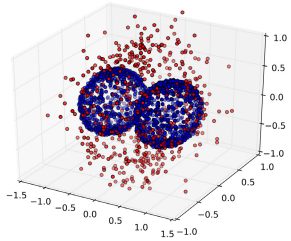
The graph generated is a representation of the connection of the points in the space, the mapping function is designed to intentionally overlap the intervals to some degree, allowing for a bunch of points to co-occur in between a group of intervals. This number of occurrences among the intervals reflects how connected the points are in the space. An *overlapping* parameter, denoted as o_p, is then defined that ranges between 0% and 100%. This value controls the overlapping degree, with a larger value meaning that there will be a greater probability for the same points to lie in two or more intervals.

The final output of the MAPPER algorithm is a network-based representation of the input data such that each partial cluster is a node and if two partial clusters have one or more shared points – according to the overlapping intervals – the nodes are linked together. Figure 1 shows a toy example of this output using a 3-dimensional synthetic input dataset. This dataset is a collection of points that resemble two touching spheres (Figure 1(a)). After generating the graph representation of the partial clusters using MAPPER, we obtain the network shown in Figure 1(c), assuming an Euclidean distance. The colours in this network represent the output values of the filter function associated with each partial cluster (node).

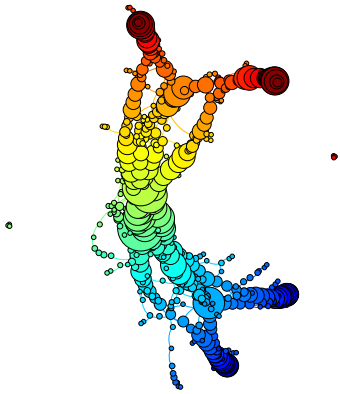
If we now add extra noise points to the synthetic dataset (Figure 1(b)) the generated network now represents those noisy points in isolated nodes – as shown in Figure 1(d) – because they can be easily separated as such from a topological perspective. Moreover, these two independent datasets in this space are represented as clearly isolated components in the network thus enabling them to be studied separately.



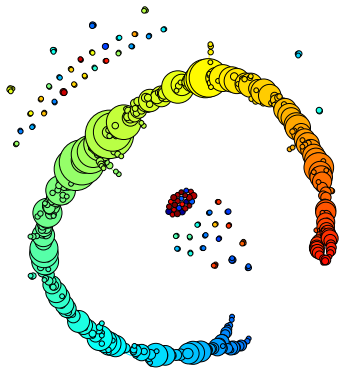
(a) Clean three-dimensional input dataset containing two touching spheres



(b) Noisy three-dimensional dataset with randomly added noise points



(c) Output network from the MAPPER algorithm for the clean dataset



(d) Output network from the MAPPER algorithm for the noisy dataset

Fig. 1. Example network outputs for the *mapper* function. A clean input dataset 1a is represented as a graph that describes how the points are connected 1c. For the case of a noisy dataset 1b, the output graph models the noise as isolated nodes 1d. The colours in the networks show the output values of the *filter* function for each node.

5 Topol: A TDA-Based Topic Detection Approach

We now provide an overview of TOPOL, our topic detection algorithm for Twitter based on Topology Data Analysis. We divide our method in three steps: *pre-processing*, *mapping* and *topic detection*. All of those are described below.

5.1 Pre-Processing Step

In TOPOL, we represent documents (the tweets) as a *bag of words* weighted by the standard Terms frequency (TF) and Inverse Document Frequency (IDF) measures [19]. Furthermore, each document has a timestamp associated indicating the moment when it was created.

We then use the windowing scheme described in Section 2 and for each time slot we perform a cleansing and filtering processes since the data still can contain undesired posts such as spam. For this we follow a similar strategy as suggested by Ifrim et al. [10]. This approach assumes that a tweet is noisy if the number of *user mentions* or *hashtags* (user-provided tags) are above a defined threshold. Even though this strategy is very simple, Ifrim et al. reported that it effectively reduces noisy tweets, specially spam and advertising since posts in these categories tend to have a high number of user mentions and hashtags. For our experiments we decided to set a conservative filtering threshold of 2, leading to 20% of the input tweets being removed.

For extracting the features we will later study using TDA from each tweet, we employ the following approach: first, we eliminate all the *URLs*, *hashtags*, *user mentions* and any non-textual symbols for all the remaining tweets. Later, all non-ASCII characters are further removed as well as punctuation marks, digits and stop words. The remaining text for each tweet is then tokenised according to white spaces and a *TF-based vector* is generated to represent the tweet. Finally we perform an additional filtering by only selecting those TF vectors that have at least more than four distinct terms or features. In summary, at the end of this pre-processing step, each selected tweet is represented as a TF vector using a globally kept dictionary.

5.2 Mapping Step

After pre-processing, we apply the Mapper algorithm to the TF vectors in the current time slot. for this, first it is necessary to generate an input metric space. Therefore, we compute the *all-to-all* distance matrix M for all the tweets in the window. For the required filter function, we generate a rectangular diagonal matrix by applying the standard Singular Value Decomposition (SVD) technique to the distance matrix M. The values of this function are then used for sampling the distance matrix. The resolution (r_p) and overlapping (o_p) parameters are set to different values in our experiments to obtain a variety of network-based representations of the TF vectors that model the input tweets (see Section 6).

Mapper divides the input space using the following work-flow: (1) it selects the maximum and minimum values of the filter function, (2) it calculates the length of the intervals according to the resolution parameter r_p, and (3) the intervals I_i are set such that they overlap using the overlapping parameter o_p. For example, if $o_p = 50\%$ the resulting intervals will share half of the available space as follows:

$$I_0 = [x_0, x_1]$$
$$I_1 = [x_1 - r_p * 0.5, x_1 + r_p * 0.5]$$

Note that all possible intervals in the image of the filter function are covered, starting from the minimum value found to the maximum. In other words, for each interval I_i, Mapper selects points such that the image of those points lie in the interval I_i. When there are enough points (> 5) in an interval, the algorithm

performs clustering using Single-linkage Clustering [12]. After this, each cluster is modelled as a node in the output network of MAPPER. If one or more of the selected points are already assigned to a different node (i.e. cluster), MAPPER creates a link between them in the output network.

5.3 Topic Detection Step

To this point, the network-based representation generated with MAPPER for each window represents the data in the feature space according to the filter function for that particular time slot. Since the noisy tweets tend to create isolated nodes in this network, the most *relevant* connected components are good candidates for identifying interesting topics. Furthermore, their most common features can be used as the topic descriptions.

Therefore, we define the topics we are interested in as the connected components in the resulting network such that the number of tweets associated with the component is above a defined threshold α. On the other hand, we use the β-most frequent features in the same components as their descriptions.

Our proposed process for identifying topics is performed independently on each time slot. Once all time slots are processed, we track similar topics across all the time windows by measuring the Cosine similarity between the topics in the current and preceding time slots. For this we create independent time series for each topic such that the topic does not match any other topic according to the similarity function. For example, if the topic t_0 is present in the windows w_0, w_1, w_2 and the topic t_1 only in the window w_1, we generate two independent time series as follows:

$$ts_0 = \{tf(t_0), tf(t_1), tf(t_2)\}$$
$$ts_1 = \{0, tf(t_1), 0\}$$

6 Experiments and Results

To evaluate TOPOL we use the same evaluation framework proposed by Aiello et al. in [1], where they studied three major real-world events that occurred in 2012. We selected one in particular, the *FA Cup Final*, to conduct our experiments. The FA Cup Final is the final match of the Football Association Challenge Cup played by the Chelsea FC and Liverpool FC teams on May 5th of 2012. Chelsea won the match with a final score of 2-1. A set of keywords and *hashtags* provided by experts was used to retrieve related posts from Twitter. The identifiers of those tweets are all publicly available[1].

We retrieved the tweets using the Twitter REST API[2]. The dataset was partitioned in time slots considering the nature of the event (using time slots corresponding to 1 minute). Aiello generated a ground truth for the dataset consisting of a manual review of published media reports about the event. This

[1] http://www.socialsensor.eu/results/datasets/72-twitter-tdt-dataset
[2] https://dev.twitter.com/rest/public

gold-standard includes 13 topics: the goals scored by players Ramirez, Drogba and Carrol respectively, as well as the kick-off, half-time and the end of the match, among others. According to Aiello, the stories selected were "*significant, time-specific and well represented on news media*". The start time assigned for each story corresponds to the time that the story emerged in mainstream news. To compare our own results we use the same metrics proposed by Aeillo et al. in their work:

Topic Recall (T-REC) is the percentage of ground truth topics correctly detected by the method. A topic is successfully detected if the keywords that comprise the topic description and the keywords mentioned in the ground truth description have a Levenshtein similarity >= 0.8 (as defined by Aiello).

Keyword Precision (K-PREC) is the percentage of successfully detected keywords in the topic description over the total keywords found by the method for the topic description.

Keyword Recall (K-REC) is the percentage of successfully detected keywords for the topic description over the total keywords included in the topic description of the ground truth.

Since there are many other topics in the dataset that are not described in the ground truth, it is not possible to calculate the true *Topic Precision*. More information about this dataset can be found in [1].

Table 1 shows the maximum T-REC and K-REC values achieved for different configurations of TOPOL. We evaluated a wide range of values for the tunable parameters, including the distance function, resolution and overlap as well as

Table 1. Comparison of Topic Recall (T-REC) and Keyword Recall (K-REC) for different distance functions, resolutions (r_p) and overlapping degrees (o_p).

T-REC for Euclidean distance

Res (r_p)	Overlapping (o_p)		
	25	50	75
5	0.385	0.462	**0.538**
10	0.308	0.308	0.462
25	0.231	0.154	0.308
50	0.231	0.231	0.231

T-REC for Cosine similarity

Res (r_p)	Overlapping (o_p)		
	25	50	75
5	0.231	0.385	0.385
10	0.308	0.308	**0.462**
25	0.231	0.308	0.308
50	0.308	0.308	0.308

K-REC for Euclidean distance

Res (r_p)	Overlapping (o_p)		
	25	50	75
5	0.571	0.667	0.643
10	**0.714**	0.529	0.583
25	0.500	0.500	0.692
50	0.600	0.571	0.600

K-REC for Cosine similarity

Res (r_p)	Overlapping (o_p)		
	25	50	75
5	0.571	0.600	0.600
10	0.556	0.526	0.591
25	0.556	0.600	**0.667**
50	0.600	0.600	0.600

other parameters. Surprisingly, the Euclidean distance function has the better T-REC on average than the Cosine similarity, as opposed to the intuition that Cosine similarity is better suited for text documents. However, since the length of the tweets in Twitter is relatively short and pretty much constant, the Euclidean distance can distinguish elements better than the Cosine similarity. This explains why the performance of our method increases when using the Euclidean distance.

We also observe that Topic Recall increases when the overlapping degree grows, suggesting that MAPPER requires an increased sampling in order to generate better connected components in the output network. This in turn suggests that the tweets are fairly scattered in the space independently of the distance function used for the mapping process. Therefore the connected components cannot be easily linked together in the network if we use a low overlapping value.

In contrast, when the resolution increases the Topic Recall metric decreases. With higher resolutions, the generated networks will have more nodes since the intervals of the filter function will be shorter. This creates networks with few connected components and this reflects the high separability of Twitter data at smaller levels, preventing a too connected network. Since we assume that small connected components in the output network are correlated to noise, in this scale the number of candidate topics becomes nearly zero. This observation explains the low Topic Recall obtained.

We studied the influence of the α and β parameters for selecting and describing topics by modifying their values while keeping the other parameters constant (see Figure 2). In this experiment, Topic Recall remained almost unchanged. This indicates that TOPOL benefits greatly from the Topology Data Analysis (TDA) mapping process, and even more than from the burst-based topic descriptions event detection approach.

Finally, we compared TOPOL with state-of-the-art methods studied by Aiello et al. in [1]. We found that our method has competitive results as seen in Table 2.

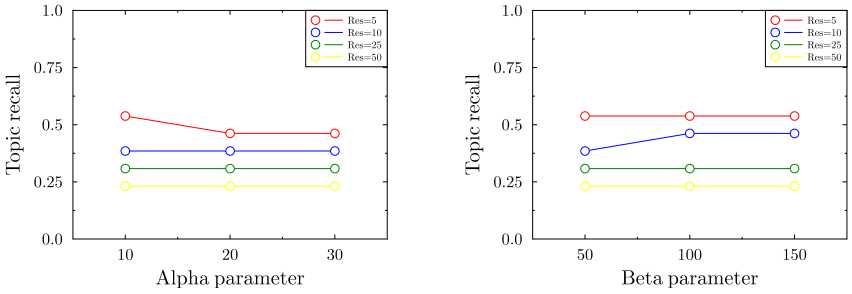


Fig. 2. Topic Recall (T-REC) for different values of α (with a fixed $\beta = 50$), β (with a fixed $\alpha = 10$) and sampling resolution (Res) parameters. The remaining parameters are maintained invariable.

Table 2. Comparison of state-of-the-art topic detection methods studied by Aiello et al. [1] and TOPOL using the Euclidean distance, $r_p = 5$ and $o_p = 75$ as parameters.

Topic Detection Method	T-REC	K-PREC	K-REC
Latent Dirichlet Allocation (LDA)	0.6923	0.1637	**0.6829**
Document-pivot	**0.7692**	0.3373	0.5833
Frequent Pattern Mining (FPM)	0.3077	**0.7500**	0.4286
Soft Frequent Pattern Mining (SFPM)	0.6154	0.2336	0.6579
BNgram	**0.7692**	0.2989	0.5778
TOPOL (based on TDA)	0.5380	0.3000	0.6430

7 Conclusions and Future Work

Detecting events in Social Network Sites (SNS) is a complex process that demands a combination of techniques such as data mining, information retrieval and text mining in order to find stories of interest that are trending in the SNS.

We introduced TOPOL, a novel method to detect topics in Twitter using Topology Data Analysis (TDA). Our method generates a network-based representation of Twitter posts that correlates with the topological shape of keywords modelled as term frequency (TF) vectors according to different distance functions. We evaluated our approach with a standard dataset and distance methods, obtaining competitive results compared to using state-of-the-art approaches [1].

We also found that the most influential parameters for our method are the overlapping degree (o_p) and the sampling resolution (r_p). Both parameters provided significant improvements in our evaluation metrics, specially Topic Recall (T-REC). In addition we showed that TOPOL relies mostly on the usage of TDA than the selection of features, improving the robustness of our approach.

Several future directions can be considered in order to improve the performance and quality of our topic detection method. Many other alternatives to the MAPPER algorithm have been developed in recent years [13]. These new outcomes avoid filtering functions and improve the computational performance. Additionally, the study of the effects of other distance functions is promising. Furthermore, different approaches can be explored as well for detecting topic changes in the topological networks, and also other algorithms for detecting bursty topics in the time series. Finally, more representational models for the SNS documents can be considered for potentially improving our initial results.

References

1. Aiello, L.M., et al.: Sensing trending topics in Twitter. IEEE Transactions on Multimedia **15**(6), 1268–1282 (2013)
2. Allan, J.: Topic Detection and Tracking: Event-based Information Organization, vol. 12. Springer Science & Business Media (2002)
3. Atefeh, F., et al.: A Survey of Techniques for Event Detection in Twitter. Computational Intelligence (2013)

4. Carlsson, G.: Topology and Data. Bulletin of the American Mathematical Society **46**(2), 255–308 (2009)
5. Castillo, C., et al.: Information credibility on twitter. In: Proc. of WWW, pp. 675–684. ACM (2011)
6. Conover, M., et al.: Political polarization on twitter. In: Proc. of ICWSM, AAAI (2011)
7. Fung, G.P.C., et al.: Parameter free bursty events detection in text streams. In: Proc. of VLDB, pp. 181–192. VLDB Endowment (2005)
8. He, Q., et al.: Bursty feature representation for clustering text streams. In: Proc. of SDM, pp. 491–496. SIAM (2007)
9. Heravi, B.R., et al.: Introducing Social Semantic Journalism. The Journal of Media Innovations **2**(1), 131–140 (2015)
10. Ifrim, G., et al.: Event Detection in Twitter using Aggressive Filtering and Hierarchical Tweet Clustering. In: SNOW-DC @ WWW, pp. 33–40. ACM (2014)
11. Imran, M., et al.: Processing Social Media Messages in Mass Emergency: A Survey. arXiv preprint arXiv:1407.7071 (2014)
12. Jain, A.K., et al.: Algorithms for Clustering Data, vol. 6. Prentice Hall, Englewood Cliffs (1988)
13. Liu, X., et al.: A Fast Algorithm for Constructing Topological Structure in Large Data. Homology, Homotopy and Applications **14**(1), 221–238 (2012)
14. Lum, P., et al.: Extracting insights from the shape of complex data using topology. Scientific Reports 3 (2013)
15. Panisson, A.: Visualization of Egyptian revolution on Twitter (February 2011). https://www.youtube.com/watch?v=2guKJfvq4uI
16. Petrović, S., et al.: Streaming first story detection with application to Twitter. In: Proc. of HLT, pp. 181–189. ACL (2010)
17. Reeb, G.: Sur les points singuliers d'une forme de Pfaff completement intégrable ou d'une fonction numérique. CR Acad. Sci. Paris **222**, 847–849 (1946)
18. Sakaki, T., et al.: Earthquake shakes Twitter users: real-time event detection by social sensors. In: Proc. of WWW, pp. 851–860. ACM (2010)
19. Salton, G., et al.: Term-weighting Approaches in Automatic Text Retrieval. Information Processing & Management **24**(5), 513–523 (1988)
20. Sayyadi, H., et al.: Event detection and tracking in social streams. In: Proc. of ICWSM. AAAI (2009)
21. Singh, G., et al.: Topological methods for the analysis of high dimensional data sets and 3D object recognition. In: Proc. of SPBG, pp. 91–100. IEEE (2007)
22. Weng, J., et al.: Event detection in twitter. In: Proc. of ICWSM, pp. 401–408. AAAI (2011)