IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 3, 2020, accepted April 2, 2020, date of publication April 7, 2020, date of current version April 27, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2986400

Leveraging Phase Transition of Topics for Event Detection in Social Media

PEDRO H. BARROS[1], ISADORA CARDOSO-PEREIRA[1], HÉCTOR ALLENDE-CID[2], OSVALDO A. ROSSO[3], AND HEITOR S. RAMOS[1], (Member, IEEE)

[1]Departamento de Ciência da Computaçãò, Universidade Federal de Minas Gerais, Belo Horizonte 31270901, Brazil
[2]Escuela de Ingeniería Informática, Pontificia Universidad Catolica de Valparaãso, Valparaíso 2340025, Chile
[3]Instituto de Física, Universidade Federal de Alagoas, Maceió 57072900, Brazil

Corresponding author: Pedro H. Barros (pedro.barros@dcc.ufmg.br)

This work was supported in part by Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), in part by Fundação de Amparo à Pesquisa do Estado de Alagoas (FAPEAL), in part by Comisión Nacional de Investigación Científica y Tecnológica (CONICYT), and in part by Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

ABSTRACT With the advancement of technology, many processes in our world have been reformulated, updated, and digitized. Therefore, interpersonal relationships have also been following this trend so that social networks have become increasingly present in our lives. Given this context, social network users create and share a large amount of data, from content about their daily lives, funny facts, as well as information about traffic, weather, and various subjects. The problem of event detection in social media, such as Twitter, is related to the identification of the first story on a topic of interest. In this work, we propose a novel approach based on the observation that tweets are subjected to a continuous phase transition when an event takes place, i.e., its underlying dynamic changes. Our proposal consists of a formal characterization of the phase transition that occurs when an event takes place, and the use of this characterization to devise a new method to detect events in Twitter, based on calculating the entropy of the keywords extracted from the content of tweets (regardless of the language used). We evaluated the performance of our approach using seven data sets, and we outperformed nine different techniques present in the literature. Unlike the work found in the literature, we present a theoretical rationale about the existence of phase transitions. For this, we characterize a model, already existing in the literature, of phase transitions described by differential equations, where we find correspondence between the model used in the study and the real data. The experimental results show that our proposal significantly improves the learning performance for the metrics used.

INDEX TERMS Event detection, information-theoretic metrics, phase transition, social media analysis

I. INTRODUCTION

Social networks have become an essential venue for social communication, information sharing, and other activities. There are different types of Social networks serving various purposes, such as relationship networks (Facebook, Twitter, Instagram), professional networks (Linkedin, Classroom 2.0), multimedia sharing networks (Youtube, Flickr), among others.

Such social networks typically produce rich amounts of user-generated information related to situation reports and can be massively used for different applications, for instance, data aggregation [1], source identification of rumors [2], social network analysis [3], recommendation systems [4], event stream dissemination [5], networks modeling [6].

The associate editor coordinating the review of this manuscript and approving it for publication was Santhosh Kumar Gopalan.

One of the most known and used social networks is Twitter. Twitter is a microblogging social network, which allows users to send and receive personal updates to/from other contacts in texts of up to 280 characters known as ''tweets''. While some of the shared topics are personal status updates, a significant fraction is a response induced by events, such as natural disasters [7] (e.g., earthquakes and tsunami [8]), political events (e.g., protests [9], elections [10]), and others.

Event Detection can be defined as finding a specific or a new pattern within a data collection (usually a stream of data). It can be a new pattern or an abnormality within the data. In the social networks context, it is related to the identification of the first story on a topic of interest by constantly monitoring news streams. The goal is to identify the first story to discuss a particular event that happened at a specific time and place. Dou *et al.* [11] define an event as ''an occurrence causing a change in the volume of text data that discusses the

associated topic at a specific time''. Therefore, Twitter can be seen as a valuable source of data useful to understand a wide range of events happening around the world, with each user being a potential informant.

Despite its value as an information source when compared to traditional media news, the analysis of Twitter data presents some challenges, of which we can mention:

- Tweets generate a large amount of data that require scalable approaches to analyze their content;
- Tweets are time-sensitive, i.e., they are shared in a real-time fashion, having a great relationship with the time they are posted;
- Due to its length restriction, a tweet typically presents a content that is brief and informal, containing unstructured sentences, typos, and abbreviations;
- More often than not, tweets present no useful information. In fact, according to Parikh and Karlapalem [12], ''half of the tweets are pointless and do not convey any valuable information''. They say that these tweets are mainly related to personal updates of users, spam, and self-promotion. Processing such tweets not only increases the processing time but also degrades the quality of results.

Detecting an event is a challenging task due to the intrinsic characteristics of tweets. Most approaches consider to find outliers (anomalies), and further, investigate whether these outliers are related to events [13]–[15].

We observed that current approaches are prone to fail when the underlying model changes, and the previous predicted model is not valid anymore. Hence, events can be related to a changing of the dynamic of the underlying model. It means that tweets related to an event have characteristics of a new model and, hence, can not be considered as an outlier. Assuming a model changing, the task of detecting an event can be seen as a phase transition detection.

One of the challenges in this type of approach is to try to differentiate a change in system dynamics from an outlier or noise. In the case of random deviations or anomalies (outlier detection) [16] the detection techniques consider the system stationary. In the case of a dynamic change, the system can not be regarded as stationary and must be treated using adaptive techniques [17].

The change of dynamics in a non-stationary environment consists of an effect called concept drift. Adaptive learning refers to updating predictive models online during their operation to react to concept drifts [17]. The change in the dynamics of data distribution over time can be observed in different ways. It can occur abruptly (for example, when we switch from one sensor to another with a different calibration) or incremental; a sensor naturally has its accuracy slowly diminished over time.

This work proposes a novel method, which benefits from phase-transition detection techniques to detect events on Twitter. Our method is based on the calculation of the entropy of the keywords extracted from the content of tweets to classify the most shared topic as an event or not.

This proposal is an extension of the previous work presented in [18]. In comparison to this previous work, we improved the topic extraction steps by proposing the summary graph partitioning technique described in Section III-B. This addition improved the topic of representation results. Moreover, we added six data sets, one of them previously labeled, and the rest unlabeled, yielding a more comprehensive performance analysis. Finally, we proposed an analytical model of the dynamics of rumor spread in social networks to validate our method. In this work, the main contributions as:

- we modeled the event detection problem as a phase transition discovery problem, and we gave empirical evidence of such phase transition and showed a formal characterization of this behavior. With this, we can capture the changing of the system dynamics. By using such characterization, we were able to tune the parameters of our event detection model, shrinking the search space of these parameters.
- we present evidence that phase transitions are carried out continuously, i.e., they can be classified as second-order transitions. To the best of our knowledge, we do not know any work that presents such formal evidence about this hypothesis.
- we used a model of the dynamic of rumor spreading in social networks. This model was characterized by 3 differential equations presented in the SIR model. The system of equations described in this manuscript was numerically solved via the Monte Carlo method. By using the Monte Carlos results, we were able to characterize the phase transition present in the SIR model, and we found that the transitions in real data sets have similar characteristics as the transitions found in the synthetic model, hence, evidencing the hypotheses raised in our work.
- we propose a graph to summarize candidate keywords related to an event, and applied a graph partitioning technique to extract the keywords of a given event. To do so, we form clusters of keywords that are more related to each other, and we also remove irrelevant words that eventually can be assigned to a cluster.

This work is organized as follows: Section II presents the related work to event detection using Twitter; Section III describes the methodology used to analyze the data; Section IV presents the main results, in addition to some comments on phase transitions; and Section V concludes this work.

II. RELATED WORK

Sakaki *et al.* [19] analyzed tweets by extracting features that represent their context. They designed a probabilistic spatiotemporal model, considering every user as a device and applied a particle filtering to find the center and the trajectory of the event location. They detected an earthquake with a high likelihood.

Proposed by [20], TEDAS is a system that detects and analyzes events on Twitter. The system has three functionalities:

detection of new events, ranking of events according to importance, and temporal (or spatial) generation of a pattern for the event.

D'Andrea *et al.* [21] proposed a system based on text mining and machine learning algorithms to detect real-time events related to traffic on Twitter. Similarly, Giridhar *et al.* [22] proposed an anomaly clarification service, which can explain sensor anomalies using social network feeds, such as vehicular traffic accidents.

The above-mentioned studies [19]–[22] belong to a class of techniques that considers a specific context of the data. As in many natural language processing applications, approaches that are specific to a certain domain generally perform better than the approaches that are open-domain or generic, however, their use is limited only to the proposed context such as traffic accidents, weather conditions, among others. Unlike this type of approach, our work proposes a new event detection technique on Twitter that considers any event regardless of its context.

EDCoW [23] (Event Detection with Clustering of Wavelet-based Signals) is a proposal that represents words as signals, based on [24], by applying wavelet analysis of the words. It then filters away trivial words by looking at their corresponding signal auto-correlations. The remaining words are then clustered to form events with a modularity-based graph partitioning technique.

Even considering any type of event, due to the need to use a transformation based on Wavelets together with modularization of the graph, EDCoW requires a massive amount of computation, making it not a scalable option. In addition, due to the cross-correlation matrix, it only considers pairs of events, the proposal cannot distinguish between different events that happen in the same period of time, for example, two football matches that occur at the same time. Our proposal considers that simultaneous events can occur simultaneously, so we hypothesize that the summary graph partitioning technique used in our approach can distinguish the events from each other.

Mathioudakis and Koudas [25] proposed the TwitterMonitor. Besides detecting events on Twitter in real-time, they provided meaningful analytics that synthesizes a description of every topic. They identified trend keywords and grouped them according to their co-occurrences, employing a context extraction algorithm based on [26].

Authors in [27] presented Tweetvent, an approach that detects bursty tweets segments within a fixed time window as events. In addition, they cluster event segments into events considering both their frequency distribution and content similarity using a variant of the Jarvis-Patrick algorithm [28]. Each cluster is compared to Wikipedia articles to identify real-life events.

Dang *et al.* [29] proposed a method based on Dynamic Bayesian Networks [30]. Their model uses the knowledge regarding tweets and analyze the topic diffusion process and finds two main characteristics of an emerging topic, which are attractiveness and key-node. The authors grouped them based on their co-occurrence using DBN-based model.

In Alsaedi *et al.* [31], authors predicted riots from Twitter using some features derived from Twitter posts (temporal, spatial, and textual content). They assumed that events could occur in the same location over a specific time or multiple events in different locations.

Jung and Nguyen [32] proposed a new model of real-time event detection, considering each user of a social network as a sensor. Therefore, because they considered that the sensors are independent, the authors have developed a measurement score that associates the unusual phenomena in a social data stream.

Aiello *et al.* [33] proposed a technique that uses a metric based on the frequency-inverse document frequency (TF-IDF) of the bigram in a given time and creates a ranking of the most probable events. Their method assumes as prior information that a fixed given number of events are happening in an interval of time and provides a ranking of the most probable events with their correspondent keywords. We see in [33], that the author proposes a model of possible events, i.e., the number of topics is defined, so that the proposals return the most likely k topics for a given time window. This type of approach requires manual intervention to later label which of the returned topics are events or not. Unlike this approach, we propose a classifier that automatically decides whether an event occurred.

Anbalagan and Valliyammai [34] proposed an information entropy-based event detection framework to identify events and their location by clustering the high-density ratio of tweets using Twitter data. The Shannon entropy of target users, location, time intervals, and hashtags are estimated to quantify the dissemination of events using the entropy maximization inference model. The geo-tagged (spatial) tweets are extracted for a specified time (temporal) to identify the location of an event and visualizes the event in geo-maps. To evaluate the proposal, they used Entropy, Cluster Score, Event Detection Hit, and False Panic Rate during four major disaster events, which are identified to illustrate the effectiveness of the proposal. The experimental outcome determines the scope and significant dissemination direction of finding events from a new perspective, which demonstrates 96% of improved event detection accuracy.

We see that entropy-based methods for detecting events on twitter as in [34] are extensively used in natural language processing applied to event detection in twitter [35]–[38]. However, we have found no formalization (in the context of phase transitions) about this fact. In this work, we conjecture that twitter has a second-order phase transition. It is important to note that the phase transition assumption is stronger than just identifying outliers. By using this characterization, we are able to fine-tune the parameters of our approach more efficiently.

Studies, as mentioned above, indicate that detecting Twitter events using emerging topics is feasible. The present work is inspired by the use of several methods mentioned

above, such as bigrams and their probabilities, but, differently from all previous work, we take advantage of the fact that entropy is a measure of a quantity of information to model the occurrence of an event, and, with this, devise a novel approach that detects the phase transition of the entropy of topics. Therefore, we try to detect the change in system dynamics and properly classify a topic trend as an event.

III. METHODOLOGY

A. DATA SET

To validate our proposal, we used two data sets collected by Aiello *et al.* [33], where authors gathered tweets related to worldwide events that happened in 2012:

– **The FA Cup Final data set**: This data set contains tweets about The Football Association Challenge Cup (FA Cup), the peak of the English football season. FA Cup is the main men's competition in English football and belongs to the oldest football association in the world. Figure 1 shows the time series corresponding to the number of collected tweets about the FA Cup. Each sample contains the number of tweets in one minute. In 2012, Chelsea and Liverpool played the final match, with goals from Ramirez (11') and Drogba (52') for Chelsea, and Carrol (62') for Liverpool. Hence, Chelsea won the match of $2-1$. It lasted 90 minutes, plus 15 minutes of half-time break;

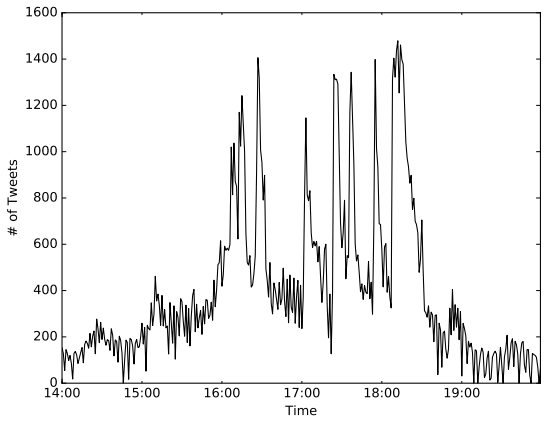

FIGURE 1. Tweets about the FA Cup.

– **The Super Tuesday Primaries data set**: In the United States, the president is elected in an indirect election, with the winner being determined by electors of the Electoral College. Super Tuesday refers informally to Tuesdays early in a US presidential primary season when the highest number of states hold primary elections. Super Tuesday in 2012 happened on March 6, 2012, with 419 delegates (18.3% of the total). We show in Figure 2 the time series corresponding to the number of collected tweets about the Super Tuesday, where each sample contains the number of tweets in five minutes.

The authors in [33] created these data sets using the official event hashtags. They constructed the ground truth by checking the mainstream media reports to identify significant topics for each data. They identified 13 and 22 important topics for FA Cup and Super Tuesday data sets, respectively. The details of the data set construction can be found in their paper.

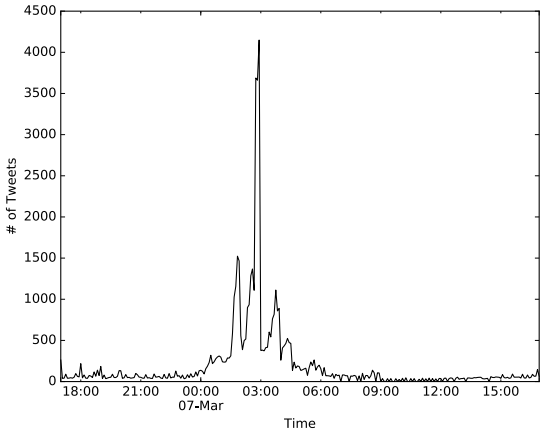

FIGURE 2. Tweets about the Super Tuesday.

In addition to the data sets mentioned above, some non-labeled data sets collected by Zubiaga [39] were used. It includes tweets for five real-world events

– **Cyprus hijacked airplane (Cyprus data set - 2016)**: An Egyptian man abducted a domestic flight, forcing the pilot to land at Larnaca International Airport in Cyprus;
– **Ebola outbreak (Ebola data set - 2014)**: was the most widespread outbreak of Ebola virus disease (EVD) in history, causing significant loss of life and socioeconomic disruption at the region of Mano river in Africa. The focus was mainly in the countries of Guinea, Liberia, and Sierra Leone;
– **Mexican election (Mexican data set - 2012)**: General elections were held in Mexico on Sunday, July 1, 2012. Voters went to the polls to elect Deputies, Senators, and the President;
– **Sismo Ecuador (Sismo data set - 2016)**: An severe earthquake happened in Ecuador on April 16, 2016, with an epicenter of 27 km from the city of Muisne, capital of the province of Esmeraldas. The earthquake had a moment magnitude of [7.8]M_w. Structures that were close to the epicenter collapsed;
– **SXSW (SXSW - 2012)**: South by Southwest is a set of music, film, and technology festivals held every spring (usually in March) in Austin, Texas, United States.

We chose these five data sets to make our evaluation as heterogeneous as possible, such as presidential elections and music festivals, regardless of the language used (Spanish or English).

Following Aiello *et al.* [33], we performed a data sanitation in the dataset, removing stopwords (such as prepositions and pronouns), punctuation marks, mentions and URLs. We also lower all the letters, in order to normalize the writing.

B. PHASE TRANSITION-BASED ENTROPY (PTb-ENTROPY)

We proposed a method to identify real-time events on Twitter by detecting the emerging topics presented in a given time, where this proposal is based on the calculation of the entropy of the time series that aggregates the frequency of the keywords extracted from tweets through a co-occurrence algorithm. This proposal can be used to detect any event because we do not consider any prior information about the event.

The notion of tweet segment was proposed in [40] for named entity recognition, not related to event detection. Similarly to [40], we consider a tweet to be a set of bigrams. A bigram is a sequence of two adjacent elements from a string of tokens, which are typically letters, syllables, or words. Therefore, we segment each tweet into words (tokenize). Then, after this transformation, we built the bigrams for each tweet and applied our proposal. Our method consists of six steps described in the following sections.

1) CALCULATING BIGRAMS

Let $f : \boldsymbol{B} \rightarrow \boldsymbol{X}$ be the function that maps the bigram $b \in \boldsymbol{B}$ into the count set $\boldsymbol{X}$ for all time slots in the data set. $\boldsymbol{B}$ is the set of all bigrams. The function f is given by $f(b) = \{x_k \in \boldsymbol{X} \mid x_k = \#b_k\}, \forall k \in \{t_0, t_1, \ldots, t_N\}$, where k is the set of all time slots in a set of observations of size N, and $\#b_k$ is the number of occurrences of the bigram b at each time k in the data set.

2) CREATING THE TIME SERIES

As the method is devised for real-time processing, we firstly collect a set of observations inside a window and then move the window forward to analyze more data. Hence, we firstly determine $\boldsymbol{B}^W$, $\boldsymbol{X}^W$, and $f^W(b)$, for the first window and continue to determine these sets for the subsequent windows. The window is denoted by $W^{i:n} = \{\boldsymbol{X}_p\}$, where $p \in \{t_i, t_{i+1}, \ldots, t_{i+n-1}\}$ is a time slot inside W, t_i is the initial time, and n is the number of elements of W. $\boldsymbol{X}_p$ denotes the set of counts for all bigrams at time p. The window, with size n, slides one-time unit, in a way that the difference between consecutive windows is exactly one observation. Note that, throughout our proposal, the value of n is fixed.

Furthermore, for each obtained bigram, we calculate the probability of its occurrence in a given sliding window. We then construct the time series of frequencies $\boldsymbol{S}^W(b) = \{s_p(b)\}$ restricted to a window W as

$$s_p(b) = \frac{x_p(b)}{\sum_{j \in p} x_j(b)}, \tag{1}$$

where $s_p(b)$ represents an observation of the time series of frequencies associated with each bigram in the corresponding window at time p. We denoted as the probability of occurrence of the bigram b for a given sliding window p as being $\hat{s_p} = s_p$. Equation (1) converts $\boldsymbol{X}^W \subset \boldsymbol{X}$, the set of counts restricted to W, into a vector of frequencies $\boldsymbol{S}^W$.

3) CALCULATING THE ENTROPY

To calculate the entropy of the bigram time series we firstly consider $\hat{s_p}$ as the estimate of the probability of the occurrence of the bigram b at each time slot p in W. Therefore, let $\hat{\boldsymbol{S}}^W$ be the vector of the estimate of probabilities of occurrence of each bigram in W. Hence, the Shannon entropy for each W is defined as

$$H_b^W = \sum_{j \in p} -\hat{s_j}(b) \log\left(\hat{s_j}(b)\right). \tag{2}$$

4) ANALYZING THE ENTROPY

The entropy H_b^W provides a measure of the quantity of information at each window W. Hence, windows that present high entropies are likely to represent the occurrence of an event. Based on the entropy value, we can infer if a window presents an anomaly behavior, i.e., an event, for a given bigram b. In such a situation, we are interested in detecting a phase transition between a window that does not present an event to a window that presents an event. In Section IV, we discuss how we detect the phase transition in the context of this work.

If an event is detected in the sliding window W, then the event is assigned to the time t_{i+n-1}, the last observation in W.

5) CREATING THE SUMMARY GRAPH

Considering that if we find a phase transition, i.e., an event, for a given bigram b_i at time t, we expand the representation of the bigram using a technique commonly applied in frequent itemset mining. For this, we proposed a variation of the FP-growth (Frequent pattern-growth) Algorithm originally introduced by [41] to adapt the technique to use bigrams. The FP-growth algorithm is an efficient and scalable method for mining, which uses an extension of the prefix-tree structure for compressed data storage called the frequent-pattern tree (FP-Tree).

The algorithm returns the co-occurrence information of the items, but, in our proposal, we select only the occurrences on the second level of the tree, that is, two words (bigram). With this, we find sets of bigrams that are related to b_i, i.e., they are likely to appear in the same tweet as b_i.

Levenshtein distance [42] can be viewed as a measurement metric to quantify the difference between two sequences. More specifically, for textual elements, we can define this distance as the minimum number of single-character edits (insertion, deletion, or substitution) required to transform one textual element into another.

Moreover, to reduce the number of selected bigrams to a smaller set with bigrams that are closer to each other, i.e., are more relevant, we rule out the bigrams that are more distant to b_i. To do so, let **lev** be the Levenshtein distance, we define the distance between the bigrams $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)$ and $\boldsymbol{\gamma} = (\gamma_1, \gamma_2)$ as

$$\begin{aligned} \text{Dist}(\boldsymbol{\alpha}, \boldsymbol{\gamma}) = \min\{&lev(\alpha_1, \gamma_1) + lev(\alpha_2, \gamma_2), \\ &lev(\alpha_1, \gamma_2) + lev(\alpha_2, \gamma_1)\}, \end{aligned} \tag{3}$$

where α_1, α_2, γ_1, and γ_2 are words of the bigrams $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$. However, we apply this distance defined in Equation (3) to the bigrams obtained by the variation of FP-growth in the sliding window W, and we obtain all sets of bigrams that have a

distance to bigram b_i with a value smaller than η, an arbitrary threshold. In this work, we used $\eta = 8$, as it will be explained in Section IV.

With this set of bigrams, we construct a graph where the vertices are the words, and two words are connected by an edge whenever they belong to the same bigram. Then, in the end, the algorithm returns a graph corresponding to the summation of the events associated with the time t.

6) PARTITIONING OF THE SUMMARY GRAPH

We split the summary graph to clean undesirable words that eventually appears in the previous step. This operation helps to identify the events correctly. After the partitioning of the graph, we consider that each resulting connected component with more than two elements is associated with an event. Therefore, we can identify more than one event inside the same window with our proposal.

For the partitioning, we used the Markov Cluster Process (MCL) [43]. The technique defines a sequence of stochastic matrix processes (inflation and expansion) called operators. The main idea of the MCL algorithm is to simulate the flow in a normalized graph. To do so, it uses a random walker and counts the number of times the walker passes by each edge (current). The algorithm increases the flow where the current is strong (the vertex received many visits) and decreases the flow where the current is weak (the vertex received few visits). The algorithm uses the operators to increase/decrease the flow. The expansion operator favors the shortest paths, that is, random walks with few steps, fostering the visit to new groups. This operator associates new probabilities to all pairs of nodes, decreasing the probability for long paths and increasing for short paths. Thus, the expansion operator is responsible for allowing the flow to connect different regions of the graph.

The inflation operator is responsible for both strengthening and weakening the current flow. Inflation will then have the effect of increasing the odds of rides inside the groups and will lower the rides between groups. This is accomplished without any prior knowledge of the grouping structure.

We decided to use this algorithm because it allows two nodes to be in two groups separately. In other words, even if two distinct events have terms in common, the algorithm can satisfactorily segment them.

C. THE PTb-ENTROPY ALGORITHM

Algorithm 1 presents the pseudocode of our proposal. The algorithm calculates the entropy of a window and decides whether this window belongs to a phase transition between the absence and the presence of an event. To do so, it receives as input the index of the initial time (i_0) of W and has access to the tweets that correspond to each time slot, returning a list of detected keywords for each event detected in W.

Firstly, it initializes the set $\boldsymbol{B}^W$ (Line 2) extracting all bigrams in W. For each time slot, it calculates the counts of bigrams in W, and sorts in decreasing order (Lines 3–8). In Lines 10 to 23, it traverses $\boldsymbol{B}$ to calculate the probability vector and the entropy of W. We limit to the 200 more representative bigrams (more frequents) to rule out clutter.

Algorithm 1 Our Proposal for Event Detection in Social Media Based on Phase Transitions

Data: i_0, the index indicating the initial time in W
Result: List of detected keywords for each event detected

1 **for** $i \leftarrow i_0$ **to** $i_0 + n - 1$ **do**
2 initialize $\boldsymbol{B}^W$
3 **for** $b \in \boldsymbol{B}^W$ **do**
4 **for** $p \in \{t_i, t_{i+1}, \ldots, t_{i+n-1}\}$ **do**
   /* constructing $\boldsymbol{X}^W$ */
5 $x_p \leftarrow f^W(b_p)$
6 **end**
7 **end**
8 sort($\boldsymbol{B}^W$, $type$ = 'decrease', by = 'x_p')
9 $\text{list}_{bigrams} = \emptyset$
10 **for** $b \in \boldsymbol{B}^W_{0:199}$ **do** /* for the first 200 bigrams in $\boldsymbol{B}^W$ */
11 **for** $p \in \{t_i, t_{i+1}, \ldots, t_{i+n-1}\}$ **do**
   /* Calculating the probability vector $S^W(b)$ */
12 $\hat{s_p} \leftarrow \frac{x_p}{\sum_{j \in p} x_j}$
13 **end**
14 $H_b^W = 0$
15 **for** $p \in \{t_i, t_{i+1}, \ldots, t_{i+n-1}\}$ **do**
   /* calculating the entropy of W */
16 $H_b^W \leftarrow H_b^W + \left(-\hat{s_p} \log\left(\hat{s_p}\right)\right)$
17 **end**
18 **if** $0.1 < H_b^W < 0.7$ ***and*** $\boldsymbol{x}_{i+n-1} > 10$ ***and*** $\max(\hat{\boldsymbol{S}}^W) == \hat{s}_{i+n-1}$ **then**
19 $\text{list}_{bigrams} \leftarrow \text{add}(\text{list}_{bigrams}, b)$
20 $\text{list}_{return} \leftarrow \text{FP-growth}(b, B^W, W, \eta = 8)$
21 $\text{list}_{bigrams} \leftarrow \text{add}(\text{list}_{bigrams}, \text{list}_{return})$
22 **end**
23 **end**
24 graph $\leftarrow$ constructGraph($\text{list}_{bigrams}$)
25 graph $\leftarrow$ MCL(graph)
26 saveEvent(graph)
27 **end**

For each bigram in $\boldsymbol{B}^W_{0:199}$, it constructs the set of the probability (frequency) of the bigram at each time slot in W (Lines 11 to 13), and calculates the subsequent window entropy between Lines 15 and 17. From Lines 18 to 22, it detects whether a window presents a transition to an event or not. In the case of detection, we consider for the graph construction, the analyzed bigram along with those associated with it, using the algorithm described in section III-B.5.

We chose the entropy detection interval of $0.1 < H_b^W < 0.7$ to detect the event by analyzing the ROC curve of our detection system, as will be explained in Section IV. The rule

$\max(\hat{\boldsymbol{S}}^W) == \hat{s}_{i+n-1}$ ensures that the entropy of the last time slot inside the window has the largest value, indicating the occurrence of a phase transition. This term avoids false positive detection for events that were identified previously.

Finally, Line 24 consists of the creation of the summary graph, which is created based on the lists of bigrams returned by the FP-growth algorithm in W. In Line 25, we use the MCL in the summary graph.

The time complexity of our algorithm is $O(|B^W|^3|W| + |B^W||W||b|^2 + T(2|B^W|^{2.807}))$, where $|B^W|$ is the amount of bigram for a respective window W, $|W|$ is the number of time slots, T is the number of convergence steps for the graph clustering algorithm (MCL) and $|b|$ is the number of letters of the largest bigram b in time window W. However, in practice that T is typically small (less than 30), and $|b| < |B^W|$, therefore, we can conclude that our algorithm has time complexity equal to $O(|B^W|^3|W|)$. Although the time complexity is cubic on $|B^W|$, i.e., the number of bigrams in W, as mentioned earlier, we consider at most 200 bigrams, as we empirically observed that the frequencies of bigrams are small (typically less than five) when the number of considered bigrams is higher than 150, for all datasets herein studied. Such bigrams are not likely to be related to events.

D. EVALUATION

Our proposal returns a set of keywords that are compared with the ground truth. The ground truth has a list of keywords that define a topic. To analyze our results, we use the same set of metrics presented in [33]. We chose these metrics because we used the same data set, so our results are directly comparable to their results. The metrics are:

- Topic recall (T-Rec): percentage of ground truth events successfully detected, i.e., the true positive rate for event detection

$$\text{T-Rec} = \frac{\text{ground truth Topic} \cap \text{detected Topic events}}{\text{g. truth Topic events}}.$$

- Keyword Precision (K-Prec): Percentage of correctly detected keywords over the total of keywords for a given ground truth event, i.e., the true negative rate for keyword detection

$$\text{K-Prec} = \frac{\text{ground truth Keywords} \cap \text{detected Keywords}}{\text{detected Keywords}}.$$

- Keyword Recall (K-Rec): Percentage of correctly detected keywords over the total of keywords for a given ground truth event, i.e., the true positive rate for keyword detection

$$\text{K-Rec} = \frac{\text{ground truth Keywords} \cap \text{detected Keywords}}{\text{g. truth Keywords}}.$$

- F_1-Score (K-Score): For a better comparison between the technique, we adopted the F_1-score for keywords metrics

$$\text{K-Score} = 2 \cdot \frac{\text{K-Rec} \cdot \text{K-Prec}}{\text{K-Rec} + \text{K-Prec}}.$$

It worths mentioning that we calculate these metrics for each time slot.

IV. RESULTS AND DISCUSSION

A. PHASE TRANSITION

Initially, to analyze the phase transition, we will model the information dissemination behavior in a social network. In this model, we will use a metaphor of an epidemic infection with immunity, which means that when an individual is infected by a disease, after a certain time, it is cured and becomes immune. This characteristic is captured by SIR model [44]. Henceforth, we consider that individuals in a social network have three states related to a piece of information, i.e., a representation of the event:

- State 0 (S): The individual does not know any information about the event that occurred (susceptible);
- State 1 (I): The individual has gained knowledge about the event, and she/he were impacted by it (infected), hence, she/he starts to make posts about it, yielding the information propagation about the event in the network (infection).
- State 2 (R): The individual knows the information, but because of a temporal impact, the information no longer influences the individual's behavior (it has become obsolete or unattractive), and she/he do not divulge it in the network (recovered).

The model includes the following set of reactions:

$$S \xrightarrow{b} I \xrightarrow{\gamma} R, \tag{4}$$

where b is the contagious rate, which takes into account the probability of getting a disease when a susceptible individual has contact with an infected subject, and γ is the recovery rate, i. e., if the duration of the infection is D, then $\gamma = 1/D$, since an individual gets recovery in D units of time.

The SIR model can be described by 3 differential equations:

$$\frac{dS}{dt} = -\frac{bIS}{N} \tag{5}$$

$$\frac{dI}{dt} = \frac{bIS}{N} - \gamma I \tag{6}$$

$$\frac{dR}{dt} = \gamma I, \tag{7}$$

where N is a number of individual in the network.

Since we want to study transitions in social networks, we use the small-world network model proposed by Watts and Strogatz [45], which consists of a complex topology abstraction model for social interactions. This model for rumor analysis has already been used in Zanette [46], but the authors did not analyze correspondences with real data.

Therefore, we studied the phase transitions present in the SIR model described above, applied to many instances of small-world networks that represent the social interactions. Hence, we can better understand how these phase transitions behave in a real data sets. For this, we characterize a relevant critical stationary exponent.

The set of equations (5), (6), and (7) models the system but can not be directly solved for one specific dimension. Hence, we used Monte Carlo simulation to estimate the function R, which accounts for the number of recovered subjects when there are no longer subjects that can be infected (stationary regime). In the Monte Carlo simulation, we model the system as a small-world network, where each vertex represents an individual, and the edges represent contacts between individuals. In this way, each subject can have one of the three states described by the SIR model, and the simulation was executed until the stationary regime is reached, i.e., when there are no more infected individuals. We performed the program multiple times for the sake of numerical convergence and statistical relevance of results.

With the results of the Monte Carlo simulation, we estimate the order parameter, the stationary density of recovery individuals, as $\Psi(b, L) = \langle N_R(b, L)\rangle / L$, where $\langle N_R(b, L)\rangle$ is the mean of the number of individuals in state 2 (R) in a small-world network of size L in the stationary regime (when the number of individuals in state 1(I) equals zero). In this work, we adopted $\gamma = 0.25/25$ as in previous work [47]. Furthermore, we see in the literature that the choice of different values for γ makes a change at the critical point, but the characteristics of the system (critical exponents) tend to keep constant, i. e., belong to the same class of universality (directed percolation) [48].

We see the evolution of the system through Monte Carlo simulation in Figure 3, where we use network sizes $L = \{103, 203, 503, 1003, 2003\}$. The experiment was replicated 15000, 10000, 7000, 5000, 2000 times respectively. We chose these values of L to represent three infected individuals randomly selected among 103 individuals.

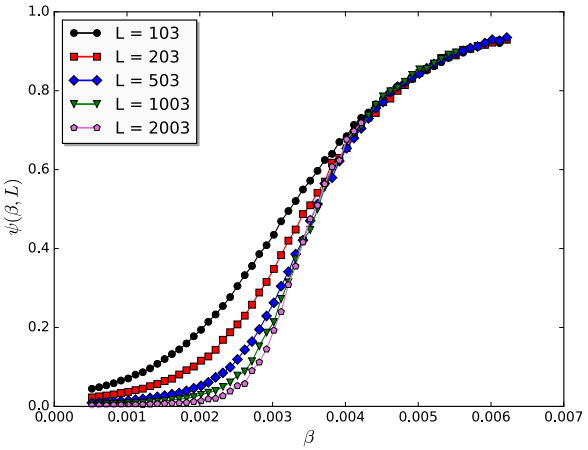

FIGURE 3. Density of recovered individuals $\Psi(b, L)$ versus b for distinct L size for Monte Carlo simulation.

We observe that this model consists of a continuous (or second-order) phase transition since the density evolves continuously from the absorbent state (low density) to the active state (high density) [49].

This feature is commonly described in the literature, and it is possible to make characterizations about the phase transition, such as divergent order parameter fluctuation, an infinite correlation length, and a power-law decay near criticality (critical exponents). In Figure 4 we report our data for the order parameter fluctuations, $\Delta\Psi(b, L) = \left[\langle N_R(b, L)^2\rangle - \langle N_R(b, L)\rangle^2\right] \cdot L$, estimated with Monte Carlo simulations, where it shows the divergent order parameter fluctuation at the critical point.

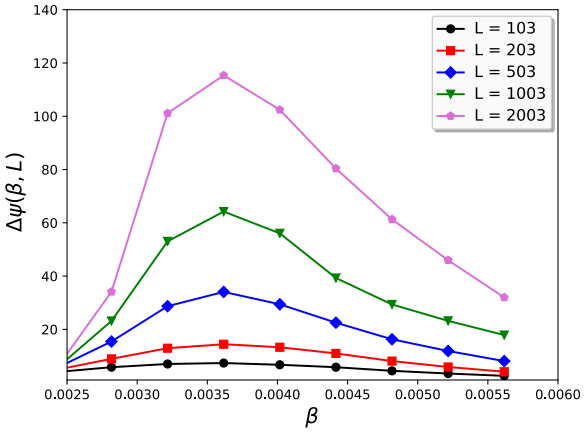

FIGURE 4. Order parameter fluctuations $\Delta\Psi(b, L)$ versus b for distinct L. The peak of the order-parameter fluctuations signalizes a phase transition.

For determining the precise location of the critical point b_c, we measure the ratio between the second moment and the square of the first moment of the number of individuals in state 2 (R), as can be seen in [50], defined as $m_L = \langle N_R(b, L)^2\rangle / \langle N_R(b, L)\rangle^2$. We can see m_L versus b in Figure 5.

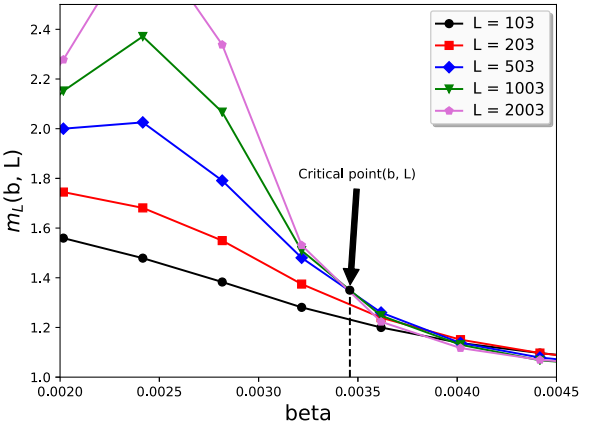

FIGURE 5. The moment ratio m_L as a function of the infection probability p for distinct L. The scale-invariant at the critical point allowed us to precisely estimate the critical contact rate $b_c = 0.003459 \pm 0.00003$.

The singular behavior near the critical point can be described as a power series. The critical exponent β can be defined as $\Psi(b, L)_{b \to b_c} \propto (b - b_c)^{\beta}$. Figure 6 shows the log-log plot of the $\Psi(b, L)$ versus $(b - b_c)$, where we find $\beta = 0.471$ with L= 5003 and 1000 repetitions.

With this, we characterize the phase transition modeled by the SIR with the numerical solution obtained by the Monte

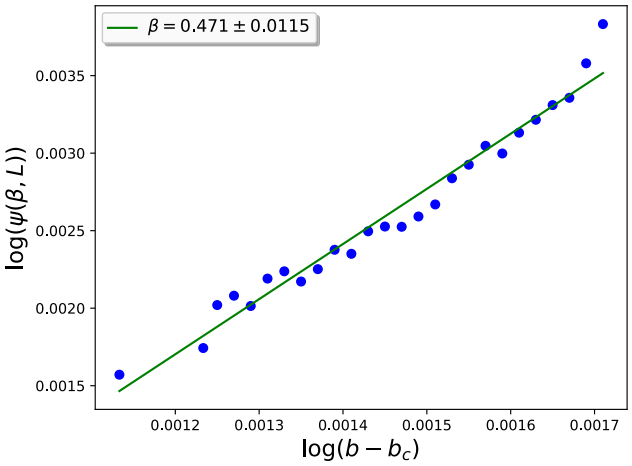

FIGURE 6. Log-log plot of the order-parameter Ψ versus $b - b_c$ above the critical point.

Carlo method. Now, we analyzed how the entropy of the bigram with the sliding window behaves with the data of the tweets.

As stated in Section III-B, the goal is to detect the phase transitions of the entropy between consecutive windows. Figure 7 shows the dynamic of the entropy of the most representative bigram (the one with more occurrences) of the FA cup data set. As we calculated the entropy of a window, we consider that the entropy of all time slots inside a window is constant. Observe that after the time 5000, the system starts to present another dynamic where the entropy is higher than before.

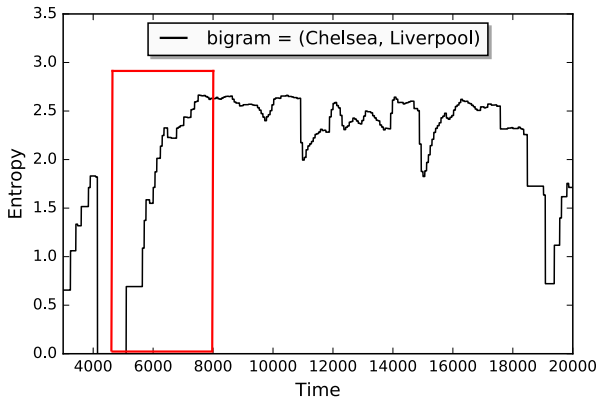

FIGURE 7. Dynamic of the entropy for a representative bigram in the evaluated data set.

We adopt as order parameter Ψ equal to H_b^W. We note the similarity between Figures 3 and 7 (for time 4000 to time 8000 in red rectangle), being this sigmoidal format (S-shape), one of the characteristics of second-order phase transitions. We observed some discontinuities in Figure 7 since the proposal needs to discretize the time that tweets are posted to create the time series of bigram frequency.

Figure 8 estimates the scale exponent β in the vicinity of the critical point $t_c = 5000$ as the slope of the fitted

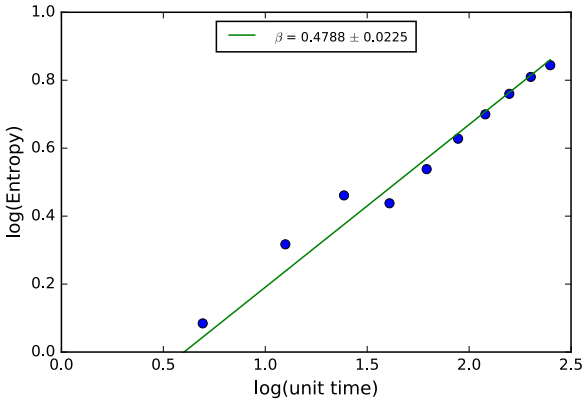

FIGURE 8. Log-log plot of $H_b^W(t)$ versus $|t - t_c|$ above the critical point. The critical exponent $\beta = 0.4788$ suggests that this is a continuous phase transition.

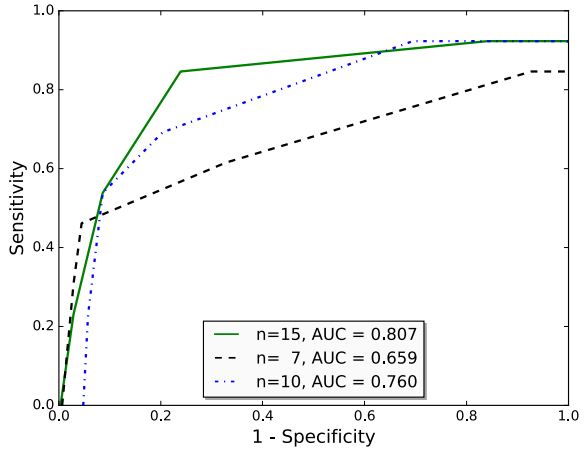

FIGURE 9. ROC curve for different values of n (length of the time window) to our proposal.

line $H_b^W(t) \propto (t - t_c)^\beta$ in a log-log plot of $H_b^W(t)$ versus $(t - t_c)$. The critical exponent $\beta = 0.4788$ suggests that this is a continuous phase transition, as stated in [51]. The β value is close to the result obtained in our Monte Carlo simulation. Therefore, the phase transitions present similar characteristics.

Such a critical exponent is characterized by a slow transient that can be a result of the intrinsic dynamic of the social media posting for circumstances like live sports. In such circumstances, some users post immediately after an occurrence of an event, and some others take some time before posting, hence, the system changes almost continuously and slowly.

To detect slow transients, we assume that whenever the entropy moves from low to high values between the minimum and maximum values (second-order transition property), we can detect a transition. Observe that we do not want to detect the transient only when the entropy reaches its maximum value. Thus, the entropy detection interval should not contain the maximum value.

Therefore, since the model depends on a few parameters (sliding window size n, entropy range, summary graph

TABLE 1. Events detected from FA Cup dataset.

#	Detected topic	Corresponding story	Sample tweet
1	goal chelsea ramires scores first liverpool 1-0	Ramires scores	#Ramires scores in the 11th minute #Chelsea lead at #Wembley
2	mikel gets yellow card agger	Mikel gets a yellow card.	@:Yellow Card to Mikel for tackling Gerrard... 37" #FACup
3	super cech line carroll goal liverpool claiming	Liverpool nearly score Andy Carroll takes a shot. Petr Cech makes a fantastic save.	@: Great Save by Petr Cech After All! #FACupFinal

TABLE 2. Events detected from SuperTuesday dataset.

#	Detected topic	Corresponding story	Sample tweet
1	newt gingrich win project georgia primar cnn abc idaho tennessee	Newt Gingrich has won the Georgia primary. #BBC	@: Newt Gingrich has won the Georgia primary.
2	mitt romney unemployment economy not fast race gingrich ron paul santorum ohio america obamacare health	The partial result of Ohio votes is released: Santorum has 39 %, Romney 36 %, Gingrich 15 % and Paul 8 % with 24 % of the voting reports.	@: Holy Crap, Sanatorium is up in Ohio...Still too close too call.

TABLE 3. Comparison of methods using FA Cup data set.

Method	T-Rec	K-Prec	K-Rec	K-Score
Petrović et al. [53] (Doc-p)	0.692	0.346	0.503	0.410
Aiello et al. [33] (FPM)	0.308	**0.694**	0.512	0.589
Aiello et al. [33] (SFPM)	0.615	0.241	0.608	0.345
Aiello et al. [33] (BNGran)	0.846	0.310	0.567	0.401
Aiello et al. [33] (GFeat-p)	0.238	0.120	0.471	0.191
Nguyen and Jung [32]	0.769	0.453	0.548	0.496
Blei et al. [54] (LDA)	0.538	0.204	**0.643**	0.310
Weng and Li [23] (EDCoW)	0.384	0.312	0.357	0.333
Choi and Park [55] (HUPM)	**0.923**	0.320	0.600	0.417
Our proposal (PTb-Entropy)	0.846	0.687	0.618	**0.651**

TABLE 4. Comparison of methods using SuperTuesday data set.

Method	T-Rec	K-Prec	K-Rec	K-Score
Petrović et al. [53] (Doc-p)	0.182	0.351	0.437	0.389
Aiello et al. [33] (FPM)	0.136	**0.698**	0.372	0.485
Aiello et al. [33] (SFPM)	0.273	0.617	0.593	0.605
Aiello et al. [33] (BNGran)	0.364	0.522	0.613	0.564
Aiello et al. [33] (GFeat-p)	0.091	0.108	0.294	0.158
Nguyen and Jung [32]	0.455	0.612	**0.714**	0.659
Blei et al. [54] (LDA)	0.136	0.101	0.212	0.137
Weng and Li [23] (EDCoW)	0.273	0.345	0.381	0.362
Choi and Park [55] (HUPM)	0.455	0.420	0.678	0.519
Our proposal (PTb-Entropy)	**0.545**	0.673	0.691	**0.682**

threshold η), we did an investigation to determine which value of this variable would maximize accuracy. In [52], the author proposed a method called Bayesian Optimization, which consists of optimizing functions such as a "black box".

The method consists of, with some known points, determining the shape of the function by regression. Usually, this prediction is made through a Gaussian process due to some characteristics (scalable to a few points and not parametric).

Thus, based on the regression of the Gaussian process it is defined a utility function that consists in finding the next candidate for the parameters aiming at the optimization of some specific metric. In this work, a discretization is performed for the parameters n and η. We used a random starting point and then 5 rounds of the algorithm, where we found the values $n = 15$, entropy interval $[0.1, 0.7]$, and $\eta = 8$. These values are used in the rest of this work.

In addition, the analysis of the ROC curve for FA Cup dataset shown in Figure 9 corroborates with the aforementioned interval of the entropy found by the Bayesian Optimization. This ROC was constructed as the Sensitivity (true positive rate) versus 1-Specificity (false positive rate) for different values of $n \in \{7, 10, 15\}$. Each point of the ROC was calculated varying the entropy detecting interval from $[0.1, 0.4]$ to $[0.1, 1.7]$ in steps of 0.1 units. The best result, with Area Under Curve AUC $= 0.807$, was obtained with $n = 15$ and entropy detecting interval $[0.1, 0.7]$. With our characterization of event detection as a phase transition problem, we are more confident that the event lies inside the window, as we tune the window size parameter with the awareness of the phase transition characteristics. Moreover, we simultaneously tuned the entropy interval as we know a priori, due to our characterization, that the event will likely happen inside the window.

B. EVENT DETECTION EVALUATION

Our method consists of capturing the phase transition of the entropy when it changes from near 0 to an intermediate value, i.e., the moment a bigram arises to a moment we consider

TABLE 5. Events detected from unlabelled data sets.

#	Detected topic	Corresponding story	time
		Cypirus data set	
1	dramatic, jumping, hostage, daring, window, pilot	A hostage jumped from the cockpit of the pilot, during the kidnapping.	29/03/2016 - 11:00
2	custody, egypt, situation, surrenders	**Unidentified Event.**	29/03/2016 - 12:00
3	wiped, embracing, condemned , apologized	The Egyptian government misnamed the suspect, and later apologized.	29/03/2016 - 20:00
		Identified Event = 13/14 = 92.86%	
		Ebola data set	
#	Detected topic	Corresponding story	time
1	deadliest, ebola, nations, stop,virus	**Unidentified Event.**	03/07/2014 - 12:00
2	virus, contracted, doctor, disease, fight, chief, lagos	A Liberian official dies after landing in Lagos, Nigeria.	23/07/2014 - 18:00
3	doctor, us, tests, liberia, positive, working, american	An American doctor was infected with the deadly Ebola virus.	27/07/2014 - 04:00
		Identified Event = 21/27 = 77.78%	
		Mexican data set	
#	Detected topic	Corresponding story	time
1	encuestas, pm, resultados, salida, si, denuncias, 8	**Unidentified Event.**	01/07/2012 - 22:00
2	12, elecciones, presidente, desde, nacional	Mexicans decide a presidential alternation after 12 years.	02/07/2012 - 04:00
3	trending, topic, ser, felicidades, lugar, primer	The users reporting the happiness of their candidate's victory.	02/07/2012 - 01:00
		Identified Event = 6/8 = 75%	
		Sismo data set	
#	Detected topic	Corresponding story	time
1	fallecidos, emergencia, say, 472	Announcement of the number of partial deaths.	18/04/2016 - 14:00
2	llega, diferentes, afectados, colombia	**Unidentified Event.**	18/04/2016 - 19:00
3	apoyo, movilizan, abiertas, para, 1979	Twitter users comment that this earthquake was the strongest since 1979.	19/04/2016 - 01:00
		Identified Event = 9/11 = 81.82%	
		SXSW data set	
#	Detected topic	Corresponding story	time
1	location, sxsw, apps, buzz	The great highlights at the event was the large use of social localization app.	11/03/2012 - 03:00
2	morello's, sxsw, cops, tom, occupy, shut	Morello did a presentation on the street outside the event because of a protest.	17/03/2012 - 12:00
3	plays, sxsw, new	**Unidentified Event.**	18/03/2012 - 06:00
		Identified Event = 19/24 = 79.17%	

it as an event, as evidenced by Figure 10. For the sake of illustration, this Figure shows some detected bigrams using the proposed method in the analyzed data set. The bigrams ({chelsea, goal}, {yellow, card}, {liverpool, goal}) depicts the three goals and two yellow cards, showing the approximate time and importance of each event as well.

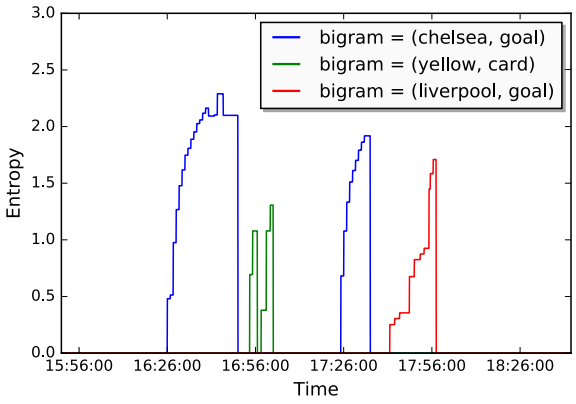

FIGURE 10. Entropy related to three detected bigrams in the analyzed data set.

These results also provide evidence that our proposal has good sensibility to time: the yellow cards occurred in a close period, but the proposal managed to differentiate them.

Note that the identified time is not the exact time in which the events occurred or lasted since users need a time (of unknown duration) to respond to the event. Comparing to the ground truth, we can see that the detected events are close to the real event, with about 4 seconds of difference between them. For the sake of illustration, Table 1 and 2 shows some detected topics along with the corresponding ground truth.

To evaluate our results, we used the metrics described in section III-D, which can be seen in Table 3 and Table 4. We compared the obtained results with some techniques from the literature. Only Choi and Park [55] and Nguyen and Jung [32] results were collected from the original articles and copied into the table of results. The best results are presented in bold.

We can observe that our approach is the second-best when in terms of the T-Rec metric, K-Prec and K-Rec for the FA Cup data set, but it obtains the best result in terms of K-score. In Table 4, we can observe that our approach is the second-best in terms of the K-Prec and K-Rec metrics, and obtains the best result in the T-Rec metric. Again our proposed model gets the best results for the K-score.

We expanded the evaluation of our approach by using these five unlabelled data sets proposed by [39]. For these cases, we had our algorithm already trained (see the parameters set by using the ROC curve in Figure 9), and we used the unlabelled data sets as a test. In this evaluation, we are mostly interested in identifying the recall metric of relevant events. To do so, we calculate the fraction of real-life events and events that were detected by our approach. For each unlabelled data set, we consider that each sample contains the number of tweets corresponding to one hour.

To calculate the recall, we collected all events detected by our algorithm, and later, made a manual association of events. For this, we perform a manual search to find if the keywords returned from the proposal had any relation to the real-life events.

To do so, we checked in specialized news websites the keywords identified by the proposal in order to construct a description of the identified event. This construction, as well as the keywords and the time that our proposal identified the event, can be seen in Table 5. For the sake of illustration, Lines labeled as 1, 2, and 3 shows three samples of events detected by our proposal, two of them correspond to events that we were able to find a correspondence in real-life events, and one of them shows an unidentified event. Besides, we calculated the percentage of events identified for each data set, as we can see at the end of each table.

V. CONCLUSION

In this work, we used entropy to model the occurrence of an event in social media. We discovered that during the occurrence of an event, the entropy of the bigrams extracted from the social media changes its dynamics, and we observed a continuous phase transition of the entropy dynamics.

For this, we model the dynamics of rumor propagation in social networks, and with this, we find and study the behavior of the phase transition found in this synthetic model. We found evidence that the synthetic phase transition (solved with Monte Carlo) has the same characteristics of the transition observed in the Twitter data.

Therefore, we proposed a novel method to detect events in Twitter based on time series formed by the probabilities of the keywords extracted from the content of the tweets. Our proposal, although not taking into account any previous information about the content of the tweet, can identify any event (even in different languages).

Considering the phase transitions, we present strong theoretical and practical evidence of the existence of tweet transitions, as well as some characteristics about them. By using such characterization, we were able to tune the parameters of our event detection model, shrinking the search space of these parameters.

The proposed method presents satisfactory results when compared to state-of-the-art and presented an overall best results compared to some models in the literature for labeled and unlabeled data sets. Furthermore, we provide some evidence that our method is sensitive to detect events that take place close in time.

As future work, we intend to perform a characterization of more critical exponents (γ', ν, and others), besides performing an analysis on the dynamic behavior of the SIR model. With this, we hope to be possible a better understanding of the phase transition as well as the behavior of the dynamics system. Moreover, We plan to differentiate first-order

and second-order phase transitions to apply more suitable techniques to detect each type of event.

REFERENCES

[1] X. Wu, X. Zhu, G.-Q. Wu, and W. Ding, "Data mining with big data," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 1, pp. 97–107, Jan. 2014.

[2] J. Jiang, S. Wen, S. Yu, Y. Xiang, and W. Zhou, "Rumor source identification in social networks with time-varying topology," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 1, pp. 166–179, Feb. 2018.

[3] J. Kim and M. Hastak, "Social network analysis: Characteristics of online social networks after a disaster," *Int. J. Inf. Manage.*, vol. 38, no. 1, pp. 86–96, Feb. 2018.

[4] C. Bothorel, N. Lathia, R. Picot-Clemente, and A. Noulas, "Location recommendation with social media data," in *Social Information Access*. Springer, 2018, pp. 624–653.

[5] H. Jin, C. Lin, H. Chen, and J. Liu, "QuickPoint: Efficiently identifying densest sub-graphs in online social networks for event stream dissemination," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 2, pp. 332–346, Feb. 2020.

[6] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Rev. Modern Phys.*, vol. 74, no. 1, p. 47, 2002.

[7] K. Zahra, M. Imran, and F. O. Ostermann, "Automatic identification of eyewitness messages on Twitter during disasters," *Inf. Process. Manage.*, vol. 57, no. 1, Jan. 2020, Art. no. 102107.

[8] B. D. M. Peary, R. Shaw, and Y. Takeuchi, "Utilization of social media in the east japan earthquake and tsunami and its effectiveness," *J. Natural Disaster Sci.*, vol. 34, no. 1, pp. 3–18, 2012.

[9] Z. Tufekci and C. Wilson, "Social media and the decision to participate in political protest: Observations from tahrir square," *J. Commun.*, vol. 62, no. 2, pp. 363–379, Apr. 2012.

[10] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe, "Predicting elections with twitter: What 140 characters reveal about political sentiment," in *Proc. 4th Int. AAAI Conf. Weblogs Social Media*, vol. 10, no. 1, 2010, , pp. 178–185.

[11] W. Dou, X. Wang, W. Ribarsky, and M. Zhou, "Event detection in social media data," in *Proc. IEEE VisWeek Workshop Interact. Vis. Text Anal.-Task Driven Anal. Social Media Content*, Oct. 2012, pp. 971–980.

[12] R. Parikh and K. Karlapalem, "Et: Events from tweets," in *Proc. 22nd Int. Conf. World Wide Web*, New York, NY, USA: ACM, 2013, pp. 613–620

[13] R. Lee and K. Sumiya, "Measuring geographical regularities of crowd behaviors for Twitter-based geo-social event detection," in *Proc. 2nd ACM SIGSPATIAL Int. Workshop Location Based Social Netw. LBSN*, New York, NY, USA: ACM, 2010, pp. 1–10.

[14] S. Motoi, T. Misu, Y. Nakada, T. Yazaki, G. Kobayashi, T. Matsumoto, and N. Yagi, "Bayesian event detection for sport games with hidden Markov model," *Pattern Anal. Appl.*, vol. 15, no. 1, pp. 59–72, Feb. 2012.

[15] M. Washha, A. Qaroush, M. Mezghani, and F. Sedes, "A topic-based hidden Markov model for real-time spam tweets filtering," *Procedia Comput. Sci.*, vol. 112, pp. 833–843, Jan. 2017.

[16] N. Giatrakos, A. Deligiannakis, M. Garofalakis, and Y. Kotidis, "Omnibus outlier detection in sensor networks using windowed locality sensitive hashing," *Future Gener. Comput. Syst.*, to be published, doi: https://doi.org/10.1016/j.future.2018.04.046.

[17] J. Gama, I. Žliobaitè, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," *ACM Comput. Surveys*, vol. 46, no. 4, pp. 1–37, Apr. 2014.

[18] P. H. Barros, I. Cardoso-Pereira, A. A. F. Loureiro, and H. S. Ramos, "Event detection in social media through phase transition of bigrams entropy," in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Jun. 2018, pp. 1068–1073.

[19] T. Sakaki, M. Okazaki, and Y. Matsuo, "Earthquake shakes Twitter users: Real-time event detection by social sensors," in *Proc. 19th Int. Conf. World wide Web WWW*, New York, NY, USA: ACM, 2010, pp. 851–860.

[20] R. Li, K. H. Lei, R. Khadiwala, and K. C.-C. Chang, "TEDAS: A Twitter-based event detection and analysis system," in *Proc. IEEE 28th Int. Conf. Data Eng.*, Apr. 2012, pp. 1273–1276.

[21] E. D'Andrea, P. Ducange, B. Lazzerini, and F. Marcelloni, "Real-time detection of traffic from Twitter stream analysis," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 2269–2283, Aug. 2015.

[22] P. Giridhar, M. T. Amin, T. Abdelzaher, D. Wang, L. Kaplan, J. George, and R. Ganti, "ClariSense+: An enhanced traffic anomaly explanation service using social network feeds," *Pervas. Mobile Comput.*, vol. 33, pp. 140–155, Dec. 2016.

[23] J. Weng and B.-S. Lee, "Event detection in Twitter," in *Proc. ICWSM*, vol. 11, 2011, pp. 401–408.

[24] O. A. Rosso, H. Craig, and P. Moscato, "Shakespeare and other english renaissance authors as characterized by information theory complexity quantifiers," *Phys. A, Stat. Mech. Appl.*, vol. 388, no. 6, pp. 916–926, Mar. 2009.

[25] M. Mathioudakis and N. Koudas, "TwitterMonitor: Trend detection over the Twitter stream," in *Proc. Int. Conf. Manage. Data SIGMOD*, New York, NY, USA: ACM, 2010, pp. 1155–1158.

[26] S. Deerwester, S. Duais, G. Furnas, T. Landauer, and R. Harshman, "Indexing by latent semantics analysis," *J. Amer. Soc. Inf. Sci.*, vol. 41, no. 6, pp. 391–407, 09 1990.

[27] C. Li, A. Sun, and A. Datta, "Twevent: Segment-based event detection from tweets," in *Proc. 21st ACM Int. Conf. Inf. Knowl. Manage. CIKM*, 2012, pp. 155–164.

[28] R. A. Jarvis and E. A. Patrick, "Clustering using a similarity measure based on shared near neighbors," *IEEE Trans. Comput.*, vols. C–22, no. 11, pp. 1025–1034, Nov. 1973.

[29] Q. Dang, F. Gao, and Y. Zhou, "Early detection method for emerging topics based on dynamic Bayesian networks in micro-blogging networks," *Expert Syst. Appl.*, vol. 57, pp. 285–295, Sep. 2016.

[30] K. P. Murphy and S. Russell, "Dynamic Bayesian networks: Representation, inference and learning," Ph.D. dissertation, Dept. Elect. Eng., Univ. California, Berkeley, CA, USA, 2002.

[31] N. Alsaedi, P. Burnap, and O. Rana, "Can we predict a riot? Disruptive event detection using Twitter," *ACM Trans. Internet Technol.*, vol. 17, no. 2, pp. 1–26, May 2017.

[32] D. T. Nguyen and J. E. Jung, "Real-time event detection for online behavioral analysis of big social data," *Future Gener. Comput. Syst.*, vol. 66, pp. 137–145, Jan. 2017.

[33] L. M. Aiello, G. Petkos, C. Martin, D. Corney, S. Papadopoulos, R. Skraba, A. Goker, I. Kompatsiaris, and A. Jaimes, "Sensing trending topics in Twitter," *IEEE Trans. Multimedia*, vol. 15, no. 6, pp. 1268–1282, Oct. 2013.

[34] A. Bhuvaneswari and C. Valliyammai, "Information entropy based event detection during disaster in cyber-social networks," *J. Intell. Fuzzy Syst.*, vol. 36, no. 5, pp. 3981–3992, May 2019.

[35] J. Lin, "Divergence measures based on the Shannon entropy," *IEEE Trans. Inf. Theory*, vol. 37, no. 1, pp. 145–151, 1991.

[36] A. Guille and C. Favre, "Event detection, tracking, and visualization in Twitter: A mention-anomaly-based approach," *Social Netw. Anal. Mining*, vol. 5, no. 1, p. 18, Dec. 2015.

[37] J. Benhardus and J. Kalita, "Streaming trend detection in Twitter," *Int. J. Web Based Communities*, vol. 9, no. 1, p. 122, 2013.

[38] J. Shetty and J. Adibi, "Discovering important nodes through graph entropy the case of enron email database," in *Proc. 3rd Int. Workshop Link Discovery LinkKDD*, 2005, pp. 74–81.

[39] A. Zubiaga, "A longitudinal assessment of the persistence of Twitter datasets," *J. Assoc. Inf. Sci. Technol.*, vol. 69, no. 8, pp. 974–984, Aug. 2018.

[40] J. Piskorski and M. Ehrmann, "On named entity recognition in targeted twitter streams in polish," in *Proc. 4th Biennial Int. Workshop Balto-Slavic Natural Lang. Process.*, 2013, pp. 84–93.

[41] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," *ACM SIGMOD Rec.*, vol. 29, no. 2, pp. 1–12, Jun. 2000.

[42] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," in *Sov. Phys. doklady*, vol. 10, no. 8, 1966, pp. 707–710.

[43] A. J. Enright, , S. Van Dongen, and C. A. Ouzounis, "An efficient algorithm for large-scale detection of protein families," *Nucleic Acids Res.*, vol. 30, no. 7, pp. 1575–1584, Apr. 2002.

[44] M. Kuperman and G. Abramson, "Small world effect in an epidemiological model," *Phys. Rev. Lett.*, vol. 86, no. 13, pp. 2909–2912, Mar. 2001.

[45] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, p. 440, 1998.

[46] D. H. Zanette, "Critical behavior of propagation on small-world networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 64, no. 5, Oct. 2001, Art. no. 050901.

[47] L. Stone, B. Shulgin, and Z. Agur, "Theoretical examination of the pulse vaccination policy in the SIR epidemic model," *Math. Comput. Model.*, vol. 31, nos. 4–5, pp. 207–215, Feb. 2000.

[48] K. Chung, Y. Baek, M. Ha, and H. Jeong, "Universality classes of the generalized epidemic process on random networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 93, no. 5, May 2016, Art. no. 052304.

[49] L. D. Landau, "On the theory of phase transitions," *Ukr. J. Phys.*, vol. 11, pp. 19–32, 1937.
[50] R. Dickman and J. K. L. da Silva, "Moment ratios for absorbing-state phase transitions," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 58, no. 4, pp. 4266–4270, Oct. 1998.
[51] C. Argolo, P. Barros, T. Tomé, I. Gleria, and M. L. Lyra, "Stationary and dynamic critical behavior of the contact process on the sierpinski carpet," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 91, no. 5, May 2015, Art. no. 052137.
[52] J. Mockus, "On the Bayes methods for seeking the extremal point," *IFAC Proc. Volumes*, vol. 8, no. 1, pp. 428–431, Aug. 1975.
[53] S. Petrović, M. Osborne, and V. Lavrenko, "Streaming first story detection with application to twitter," in *Proc. Hum. Lang. Technol., Annu. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2010, pp. 181–189.
[54] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003.
[55] H.-J. Choi and C. H. Park, "Emerging topic detection in Twitter stream based on high utility pattern mining," *Expert Syst. Appl.*, vol. 115, pp. 27–36, Jan. 2019.

PEDRO H. BARROS received the degree in computer engineering from the Federal University of Alagoas. He is currently pursuing the master's degree in computer science with the Federal University of Minas Gerais. His research interests include urban computing, mobile computing, state equation, phase equilibrium, and phase transitions.

ISADORA CARDOSO-PEREIRA received the degree in computer engineering from the Federal University of Alagoas, Maceió, Brazil. She is currently pursuing the master's degree in computer science with the Federal University of Minas Gerais. Her research interests include data science, artificial intelligence, the Internet of Things, and urban computing.

HÉCTOR ALLENDE-CID received the Ph.D. degree from Universidad Técnica Federico Santa María, Chile, in 2015. He is currently an Assistant Professor with the Escuela de Ingeniería Informática, Pontificia Universidad Católica de Valparaíso. His research interests include supervised algorithms, distributed regression methods, and image processing.

OSVALDO A. ROSSO was born in Rojas, Buenos Aires, Argentina, in 15 October 1954. He received the M.Sc. and Ph.D. degrees in physics from the Universidad Nacional de La Plata, La Plata, Buenos Aires, in 1978 and 1984, respectively. He held a Postdoctoral position with Forschung Zentrum Jülich GmBH, Jülich, Germany, from 1988 to 1990. He was a Visiting Researcher with the Istituto Lamel, Sezione di Cinematografia Scientifica, Consiglio Nazionale Delle Ricerche, Bologna, Italy, from 1990 to 1992. He was a Research Academician with the School of Electrical Engineering and Computer Science, University of Newcastle, Newcastle, Australia, from 2007 to 2009. He was a Professor Visitante do Exterior, a Categoria Doutor Senior, and a Coordenacao de Aperfeicoamento de Pessoal de Nivel Superior with the Departamento de Física, Instituto de Ciências Exatas, Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, Brazil, from 2010 to 2012. He was a Pesquisador Visitante do Exterior with the Conselho Nacional de Desenvolvimento Científico e Tecnológico CNPq, Brazil, Instituto de Computação, Universidade Federal de Alagoas, Maceió, Alagoas, Brazil, from 2012 to 2014. From 2014 to 2016, he was a Professor Visitante with the Instituto de Física, Universidade Federal de Alagoas (UFAL), Maceió, Alagoas, Brazil, where he has been an Adjoint Professor, since January 2017. Since March 1985, he has held a permanent research position with the Argentinean Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), Argentina, being actually a Principal Researcher. He has authored or coauthored over 291 research publications including about 156 articles published in international journals (H-index = 38, total of citations = 5085). His current research interests include time series analysis, nonlinear dynamics, information theory, time frequency analysis, complex networks and their applications to physics, biological, and medical sciences.

HEITOR S. RAMOS (Member, IEEE) received the B.Sc. degree in electrical engineering from the Universidade Federal da Paraíba (UFPB), in 1992, the M.Sc. degree in computing modeling from the Universidade Federal de Alagoas (UFAL), in 2004, and the Ph.D. degree in computer science from the Universidade Federal de Minas Gerais (UFMG), in 2012. He is currently a Professor with the Department of Computer Science, Federal University of Minas Gerais, DCC/UFMG, Brazil. His research interests include wireless networks, sensors networks, mobile and ad hoc networks, and urban computing.