A Metadata-Based Event Detection Method Using Temporal Herding Factor and Social Synchrony on Twitter Data

Nirmal Kumar Sivaraman(✉), Vibhor Agarwal, Yash Vekaria, and Sakthi Balan Muthiah

The LNM Institute of Information Technology, Jaipur, India
{nirmal.sivaraman,vibhor.agarwal.y16,yash.vekaria.y16, sakthi.balan}@lnmiit.ac.in

Abstract. Detecting events from social media data is an important problem. In this paper, we propose a novel method to detect events by detecting traces of herding in the Twitter data. We analyze only the metadata for this and not the content of the tweets. We evaluate our method on a dataset of 3.3 million tweets that was collected by us. We then compared the results obtained from our method with a state of the art method called Twitinfo on the above mentioned 3.3 million dataset. Our method showed better results. To check the generality of our method, we tested it on a publicly available dataset of 1.28 million tweets and the results convey that our method can be generalised.

Keywords: Event detection · Temporal Herding Factor · Social network analysis

1 Introduction

In this work, we study the Twitter activities of the users and examine if we can find a definite behavioral trait for tweets concerning events without looking at the content of the tweets. The working definition of an *event* is as follows – something that happens and captures the attention of many people. In case of online social media like Twitter, *measuring the attention* is equivalent to measuring whether they are putting any tweet about what has happened.

We propose a novel method for event detection using a novel measure called *Temporal Herding Factor (THF)*. Any event that has a substantial impact on the society will be a talking point in the social media for at least a few days. In this work, we use one day as the granularity of time. Our approach to event detection is a term interestingness approach [5], where we consider hashtags as the terms at a granularity of time of one day. We use the idea of social synchrony [9] to detect events using THF to quantify the traces of herding in the Twitter data. When there is herding, we consider that there is a corresponding event. Importantly, we use only metadata to detect events. For evaluation of our work, we collected a

S. Cherfi et al. (Eds.): RCIS 2021, LNBIP 415, pp. 637–643, 2021.
https://doi.org/10.1007/978-3-030-75018-3_47

dataset that contains 3.3 million tweets geotagged for India. We call this dataset as 3.3M dataset[1]. We got precision, recall and $F1$ score of 0.76, 0.89 and 0.82 respectively. To check the generality of our method, we considered the generic dataset that has no geotags and is from a different time period. We call this dataset as generic dataset. We tested this dataset using the same thresholds that were calculated for the 3.3M dataset. We got a precision, recall and $F1$ score of 0.70, 0.97 and 0.81 respectively. The dataset that is available publicly contains 1.28 million tweets. Also we compared our results with a state of the art method called Twitinfo that is closest to our approach. We observed that our method has more F1 score.

2 Related Works

There are a lot of works on event detection in the literature. According to [5], the event detection methods can be broadly classified into four,

- Term-interestingness-based approaches,
- Topic-modeling-based approaches,
- Incremental-clustering-based approaches, and
- Miscellaneous approaches.

Term-interestingness-based approaches rely on tracking the terms that are likely to be related to an event [7]. Twitinfo method [7] has the best F1 score among the term interestingness approaches [5]. Topic-modeling-based approaches depend on the probabilistic topic models to detect real-world events by identifying latent topics from the Twitter data stream [3]. Incremental-clustering-based approaches follow an incremental clustering strategy to avoid having a fixed number of clusters [4]. Miscellaneous approaches are the ones that adopt hybrid techniques, which do not directly fall under the three categories [1].

3 A Model to Detect Online Events

We detect events by using the ideas of herding that is calculated as THF and social synchrony. In this section, we discuss Herding, our formulation of THF, social synchrony and our methodology.

3.1 Social Synchrony

According to [9], surge and social synchrony are defined as follows:

Surge: *A social phenomenon where many agents perform some action at the same time and the number of such agents first increases and then decreases.*

Social Synchrony: *A surge where the agents perform the same action.*

The problem of detecting the presence of events may be described as detecting social synchronies in Twitter with the following criteria:

[1] Dataset is available at https://tinyurl.com/244t7t46.

- The criteria for agents to be considered for observation – all the users tweeting with the same hashtag
- Find the surge in the number of agents by using the Algorithm given in [9].
- The criteria to measure the *sameness* of the agents' actions – tweeting with the same hashtag and the parameter that we introduce in this paper called *Temporal Herding Factor* of the surge being above a threshold value. This is discussed in detail in Sect. 3.2.

3.2 Temporal Herding Factor (THF)

At a behavioural level, the most popular form of herding behaviour is the tendency to imitate results [2]. Retweets can be taken as markers of *the tendency to imitate results* in case of Twitter [6]. To detect herding behavior in the Twitter users we observe their tweeting activity. At each time slice, we consider the new users with respect to the previous time slice and find out the fraction of them who retweets. We call this parameter *Temporal Herding Factor (THF)*.

We consider all the hashtags that are present in the dataset. We first take the list of all the hashtags and consider the set of hashtags as $H = \{h_1, h_2, h_3, ..., h_n\}$. The set of users who tweet regarding a topic h_j are represented as:

$$U^{h_j} = \{u_{j1}, u_{j2}, u_{j3}, ..., u_{jm}\}$$

A surge is the distribution of tweets regarding a hashtag where the number of tweets increases first and then decreases. Let a surge with respect to hashtag h is represented as S_h. S_h is divided into N time slices of equal-length and t_i denote the i^{th} time slice of the surge where $i \in \{1, 2, ..., N\}$.

Let $U^h_T(t_i)$ denote the set of all the unique users who posted *tweet(s)* that are not *retweets* related to the hashtag h in the time slice t_i, $U^h_{RT}(t_i)$ denotes the set of all the unique users who retweeted related to the hashtag h in the time slice t_i and $U^h_{all}(t_i)$ denotes the set of all the unique users involved in the tweeting or retweeting activity related to the hashtag h in the time slice t_i.

$$U^h_{all}(t_i) = U^h_T(t_i) \cup U^h_{RT}(t_i)$$

THF at the i^{th} time slice (t_i) of S_h is defined as follows:

$$THF(t_i) = \begin{cases} \frac{|U^h_{RT}(t_i)|}{|U^h_{all}(t_i)|} & : \text{if } t_i = t_1 \\ 0 & : \text{if } |U^h_{all}(t_i) - U^h_{all}(t_{i-1})| = 0 \\ \frac{|U^h_{RT}(t_i) - U^h_{all}(t_{i-1})|}{|U^h_{all}(t_i) - U^h_{all}(t_{i-1})|} & : \text{otherwise} \end{cases}$$

Here, $|U^h_{RT}(t_i) - U^h_{all}(t_{i-1})|$ represents the number of all the unique users who have retweeted with the hashtag h in the time slice t_i but have not tweeted or retweeted in t_{i-1}. When the combined set of all the unique users tweeting or retweeting with the hashtag h are same for two consecutive time slices t_i and t_{i-1} (i.e., $|U^h_{all}(t_i) - U^h_{all}(t_{i-1})| = 0$), then $THF(t_i)$ is considered as 0. The above formula is used for computing THF values for every time slice $t_i \in \{t_2, t_3, ..., t_N\}$.

At time slice $t_i = t_1$, we are assuming that all the users are new users (i.e. they are involved in the surge for the first time) and hence,

$$THF(t_1) = \frac{|U^h_{RT}(t_1)|}{|U^h_{all}(t_1)|}$$

Now that we have the values $THF(t_i)$ at every time slice t_i, we aggregate them by taking average.

$$THF_{avg} = \frac{1}{N} \sum\nolimits_{i=1}^{N} THF(t_i)$$

We hypothesize that the value of THF_{avg} is higher for the tweets regarding an event as compared to the tweets regarding random topics. Our method is outlined in Fig. 1.

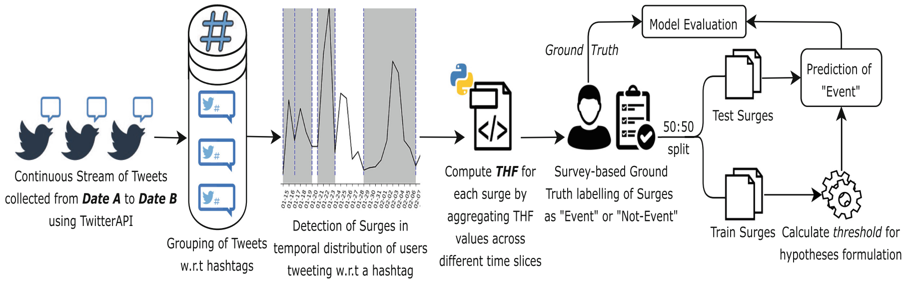

Fig. 1. Methodology for Event Detection using THF.

4 Evaluation

In this section, we evaluate our event detection model described in the previous section. The 3.3 M dataset that is used for evaluation was downloaded using the Twitter API. This dataset contains $3,360,608$ tweets from 15^{th} Jan 2018 to 4^{th} Mar 2018.

We took $28,415$ tweets randomly out of all the tweets scraped for each day. This was done so that the dataset can be uniformly distributed over all the days. The number 28415 was chosen because this was the smallest number of tweets that were captured on a single day during this time period. After this sampling, we had 1363920 tweets posted by 280286 unique users. We detected 244 surges in our dataset. Out of these 244 surges, we dropped 41 surges since they had less than 50 tweets. We computed the THF values for each one of the 203 surges.

Labelling: We conducted a comprehensive survey on the tweets in each of the Candidate surges and labelled them as events or non-events. We randomly picked 50 tweets from each surge. Each of these surges are then annotated by 3 people. The survey was conducted amongst 27 volunteers of age group 17–23 years, who were familiar with Twitter. The questions that were asked in the survey form are:

- How many tweets are talking about an event? (this is to find out all the tweets regarding events)
- How many tweets are there in the largest set of tweets that are talking about the same event? (this is to find the largest cluster among the tweets that are talking about an event)

If the answer to the second question is more than 17 (33% of 50 tweets), we label them as events. The Fleiss' Kappa coefficient is 0.73.

We labelled surges as events and non-events. Here, *Events* imply that the corresponding surge has at least one event. The THF_{avg} value was computed for all the surges in each method. We considered manual classification labelling as the ground truth. We then divided our dataset randomly into two equal parts – one for training and the other for testing. We randomly selected 50% of our data for the training set. We calculated the Mean and Standard Deviation of the THF_{avg} values corresponding to the surges in the training set that are labeled as events. We then selected threshold T as discussed further in order to define a range as our hypothesis for predicting whether a surge corresponds to an event or not. The hypothesis is as follows:

Hypothesis: If $Mean_E - (T \times SD_E) < THF_{avg} < Mean_E + (T \times SD_E)$, then there is at least one event in the corresponding surge.

In the above, $Mean_E$ and SD_E represent the mean and standard deviation of the THF_{avg} values respectively, corresponding to the surges in the training set that are labelled as events. T is the number of standard deviations we consider to detect the outliers.

Choosing Best T by Multiple Runs: In the hypothesis given above, choosing the right value of T is a crucial part. In order to select the value of T that gives the most accurate results, we evaluated our method on different values of T. Further, we carried out 10 random runs of training-testing on our dataset for each value of T. The average precision, average recall and F1 score are 0.76, 0.89 and 0.82 respectively.

4.1 Comparing with the Event Detection Method Twitinfo

In this section, we report the results of the comparison between our model and the Twitinfo model; both implemented on the 3.3M dataset. We compare our results with Twitinfo model because that is the closest to our approach.

Detecting events using Twitinfo on Our Dataset: Using the same values of T and α as Twitinfo uses, would not be an optimal decision since the granularity at which they analyze the tweets is at minute-level whereas we deal with the day-level analysis in the THF_{avg} model. As a result, we test the Twitinfo algorithm on our dataset using different values of T and α.

We applied the Twitinfo method on our dataset for differnt values of T and α. The best results are obtained for $T = 3.0$ and $\alpha = 0.225$. However, Recall in detecting Events is relatively poor in this case. The Precision, Recall and F1-score of Twitinfo method are 0.81, 0.53 and 0.64 respectively, whereas, The Precision, Recall and F1-score of our method are 0.76, 0.89 and 0.82 respectively.

5 Generality of the Hypothesis

To test the generality of our hypothesis, we verify the hypothesis on a generic dataset that contains 1280000 tweets from 14^{th} Dec 2011 to 11^{th} Jan 2012. This dataset is not restricted to any particular region and is from a different time period. We downloaded the Twitter firehose dataset that was used in [8]. This dataset is also listed in the *ICWSM* website[2] and is publicly available.

There are 77 hashtags and 547 surges in the dataset. Out of them, majority of the tweets in 423 surges are non-English. There were 14 surges that were too small – having less than 50 tweets. We discarded all such surges. Hence, we were left with 110 surges.

To test the hypothesis, we manually labeled the surges that represent events as described in previous sections. We then tested the same hypothesis that we formulated from the 33m dataset, on these surges. The precision, recall and F1 score of the method on Generic Dataset are 0.70, 0.97 and 0.81 respectively for $T = 1$.

6 Conclusion

In this paper, we proposed a method to detect events from Twitter data, based on our hypothesis that herding occurs in surges during events. Results obtained from our method show that it performs better than the state of the art method Twitinfo. Moreover, we tested our method on an openly available dataset (Generic dataset). We used the same boundary values that we calculated from 3.3M dataset and showed that our algorithm works with F1-score of 0.81 even with the Generic dataset. This indicates that herding is a distinguishing factor when it comes to the activities of the users during events versus the activities when there is no evet.

References

1. Adedoyin-Olowe, M., Gaber, M.M., Dancausa, C.M., Stahl, F., Gomes, J.B.: A rule dynamics approach to event detection in Twitter with its application to sports and politics. Expert Syst. Appl. **55**, 351–360 (2016)
2. Banerjee, A.V.: A simple model of herd behavior. Q. J. Econ. **107**(3), 797–817 (1992)
3. Cai, H., Yang, Y., Li, X., Huang, Z.: What are popular: exploring Twitter features for event detection, tracking and visualization. In: Proceedings of the 23rd ACM International Conference on Multimedia, pp. 89–98. ACM (2015)
4. Hasan, M., Orgun, M.A., Schwitter, R.: TwitterNews+: a framework for real time event detection from the Twitter data stream. In: Spiro, E., Ahn, Y.-Y. (eds.) SocInfo 2016. LNCS, vol. 10046, pp. 224–239. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-47880-7_14

[2] https://www.icwsm.org/2018/datasets/.

5. Hasan, M., Orgun, M.A., Schwitter, R.: A survey on real-time event detection from the Twitter data stream. J. Inf. Sci. (2017)
6. Li, H., Sakamoto, Y.: Re-tweet count matters: social influences on sharing of disaster-related tweets. J. Homel. Secur. Emerg. Manage. **12**(3), 737–761 (2015)
7. Marcus, A., Bernstein, M.S., Badar, O., Karger, D.R., Madden, S., Miller, R.C.: Twitinfo: aggregating and visualizing microblogs for event exploration. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 227–236. ACM (2011)
8. Morstatter, F., Pfeffer, J., Liu, H., Carley, K.M.: Is the sample good enough? comparing data from Twitter's streaming API with Twitter's firehose. In: ICWSM (2013)
9. Sivaraman, N.K., Muthiah, S.B., Agarwal, P., Todwal, L.: Social synchrony in online social networks and its application in event detection from twitter data. In: Proceedings of the 2020 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT 2020) (2020)