Organized Behavior Classification of Tweet Sets using Supervised Learning Methods

Erdem Beğenilmiş
Bogazici University
Istanbul, Turkey
erdem.begenilmis@gmail.com

Suzan Uskudarli
Bogazici University
Istanbul, Turkey
suzan.uskudarli@boun.edu.tr

ABSTRACT

There is an increasing incidence in negative propaganda and fake news, which has recently gained lots of attention during the 2016 elections in United States, France, and United Kingdom. Bots and hired users collaborate to make messages seen and persist so they may spread and gain support. Assuming that most Twitter users post without predetermined, malicious intent, there is a need for automated detection of organized behavior to protect users from manipulation. This work proposes an automated approach to classify tweets with organized behavior. Supervised learning methods are used to classify the tweets by using a training data set with 850 records based on the analysis of over 200 million tweets. Our model gave promising results for detection of organized behavior and this motivated us to proceed with the generation of two more classifiers such as [“political”, “non-political”] and [“pro-Trump”, “pro-Hillary”,“neither"]. In each cases, the random forest algorithm consistently results in high scores with an average accuracy and f-measure above 0.95.

CCS CONCEPTS

• **Information systems** → **Decision support systems**; **Social networks**; • **Computer systems organization** → *Real-time systems*;

KEYWORDS

Political propaganda, 2016 US presidential elections, organized behavior detection, supervised learning, social media analysis, big data, Twitter

ACM Reference Format:
Erdem Beğenilmiş and Suzan Uskudarli. 2018. Organized Behavior Classification of Tweet Sets using Supervised Learning Methods. In *WIMS '18: 8th International Conference on Web Intelligence, Mining and Semantics, June 25–27, 2018, Novi Sad, Serbia.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3227609.3227665

1 INTRODUCTION

Politics is one of the most prominent uses of social media platforms due to their facilitation of reaching the masses. The 2016 US presidential election demonstrated the effectiveness of using Twitter[13]. During the election, approximately 400,000 bots generated around 3.8 million tweets corresponding to 19 percent of all campaign related posts [18]. An investigation of fake news during the 2016 US election revealed that Veles (a Macedonian town with a population of 45,000) was the source of many pro-Trump fake news [22]. Studies also show that illegal groups like ISIS and White Supremacy Extremists[1] strategically use social media to recruit new members and to disseminate their propaganda [6, 7]. These events suggest that there is a need for malicious activity detection in social media.

This work proposes an approach for the near real time automatic detection of organized behavior. The term organized behavior is used to refer to a collaborative and coordinated posting behavior involving multiple users, who serve an agenda. On the other hand, tweets that are posted spontaneously without any predetermined agenda are referred to organic behavior[2]. The proposed method is implemented analyzing more than 200 million tweets, which are mainly posted during the 2016 US presidential election. Our approach gave promising results for organized behavior detection and this motivated us to generate two more classifiers such as [“political”, “non-political”] and [“pro-Trump”, “pro-Hillary”,“neither"]. In each cases, the random forest algorithm consistently results in high scores with an average accuracy and f-measure above 0.95. During the generation of classification models, all the features are extracted by analyzing user characteristics & temporal tweeting patterns, which are independent of the content and graph related features. This is akin to sensing that *something is up* without knowing *what is up*. Content and graph related features are not used, because extraction of these features are not cheap and would be problematic for real time detection.

The main contributions of this study are a classification model to detect organized behavior in tweet sets, a prototype implementation for feature extraction & classification, and a data set consists of more than 200 million tweets and features. The remainder of the paper is structured as follows: Section 2 describes background information; Section 3 presents related work; Section 4 describes the overall approach and used features; Section 5 presents the proposed model; Section 6 presents the experiments and results; Section 7 discusses the results and future work; Section 8 concludes.

WIMS '18, June 25–27, 2018, Novi Sad, Serbia

ACM ISBN 978-1-4503-5489-9/18/06…$15.00
https://doi.org/10.1145/3227609.3227665

[1] FBI known violent extreme groups: https://www.fbi.gov/cve508/teen-website/what-are-known-violent-extremist-groups.

[2] The organic term is borrowed from [10]

2 BACKGROUND

In this study, Twitter Rest API[3] (User Timeline API and Search API) is used for data retrieval. User Timeline API returns tweets of users, while the Search API supports queries subject to various criterias. Queries return results from a sampling of tweets in the recent 7 days.

In supervised learning methods, a model is trained with datasets consisting of labeled data samples that identify the class of the data. This work examines three supervised learning methods: random forest, support vector machine (SVM), and logistic regression. In the random forest, a forest of random decorrelated decision trees are created from a trained feature set [27]. The resulting forest is used to predict a classification based on the most predicted class by its decision trees. The support vector machine aims to identify a hyperplane that best divides a dataset into its classes[3]. Logistic regression is a statistical approach that is well suited for binary classification problems[27].

Principal component analysis (PCA) is a statistical method for explaining data with large number of variable using a smaller number of variables. PCA aims to find the minimum number of uncorrelated features with the highest variances to reduce dimension in the data [3]. In the prototype implementation, classification experiments have relied on Apache Spark MLlib[23] while PCA processes are done by Weka[1].

3 RELATED WORK

Related works can be categorized into behavior detection, spam detection, content and group detection.

Cao [10] examines the URL sharing behavior in Twitter and distinguishes the sharing behaviors as organic and organized. For the detection of organized urls, a graph is generated with nodes representing users and edges representing the use of the same URLs in posts. Then, URL and posting time based features of those users are extracted and supervised classifications methods are applied on manually labeled training data with 406 organic and 406 organized records. Random Forest gave the best result with F measure(0.84) and ROC area(0.92).

Ratkiewicz[21] studied to detect astroturf contents in political campaigns, which are run by politically-motivated individuals. Topological, content-based and crowdsourced features of information diffusion networks on Twitter are extracted using a composed directed graph whose nodes represent individual users and edges represent the retweet, mention, reply events between users. Supervised learning methods are used to detect astroturf content resulting in an accuracy better than 96%.

In another study about the classification of group behaviors [19], it is aimed to detect criminal and anti-social activities in social media. Graph matching algorithms are applied to explore consistent social interactions.

Also, in order to disclose spam URLs, Cao [9] analyzed behavioral features in three categories : click-based, posting based, clicking statistics. With these features, a training data set is created with 1,049 spam and benign urls by checking the labels of urls in the tweets from a category website. Another training data set is created by manual labelling of 219 benign and 79 spam urls. For the both behavioral features in two training data sets, random forest algorithm is used as supervised learning method by using 10-fold cross validation. The algorithm is trained by using all feature sets together and separately such as click-based, posting based, clicking statistics. With this approach, 86% of accuracy is found by using all features for the training.

In another research, political bot accounts, who take place in Brexit Referendum and play strategic role in referendum conversations, are analyzed [16]. It is found that these bots use excessively the family of hashtags associated with the argument for leaving the European Union, and utilize different levels of automation. It is stated that these bots, which compose one percent of sampled accounts, generate almost a third of all the messages for the Brexit referendum contents in Twitter.

When Cao's study about organized urls [10] is compared to our study, the main focus of this study [10] is to detect organized URL behaviors, while our study aims to detect organized behaviors. Also, the extracted features are different from each other. However, in both study, same classification approaches are used and random forest algorithm gave the most promising results. Furthermore, when Ratkiewicz's study about astroturf political campaigns [21] is compared with our study, same supervised classification methods are used in both studies, while feature extraction phases are different from each other. Also, in the study about group classification [19], group behaviors are identified as a result of graph matching algorithms. In our study no graph related algorithm or feature are used due to performance concerns for real time detection. In the work about [9] identifying spam urls, similar works have been done considering feature extraction and classification methods. However, our work focuses on detection of organized behavior patterns and features of user & temporal tweets.

There are also studies about content and group detection which are in parallel with our study [4, 14, 24]. In all studies either underlying content or group is detected. However, in all of them the detection is done by using language or topic related features. On the other hand, in our study this detection has been made with features which do not contain network and content related features.

4 APPROACH & FEATURES

In Twitter approximately 6, 000 tweets are posted per second [4]. In organized behaviors, it is more likely that users use strategies to increase the likelihood of their posts' observations. To understand these behaviors better, we studied on tweets with hashtags, because hashtags are used to increase engagement. For each traced hashtag (*tracedHT*), we collected a tweet set and extracted features of it. Studies about social media activism[12, 15, 20] report some characteristics of organized behaviors as sharing a common goal, temporal synchronization among users, and the dissemination of messages. Based on these characteristics, we identified two kinds of main feature types, user & temporal features, to detect the presence of organized behavior. The user features capture information about the characteristics of users, while the temporal features are used to detect the presence of a synchronization. Table 1 defines a set of functions used to formulate these features.

[3] Twitter REST API: https://developer.twitter.com/en/docs (accessed:2018-02-21).

[4] http://www.internetlivestats.com/twitter-statistics

Table 1: Descriptions of user and tweet functions, where T is set of tweets, t is a tweet, h is a hashtag, u is a user, D is a set of Days, Δ is a duration

Function	Type	Description
User functions		
reg-date(u)	*Date*	date u registered
following#(u)	*Integer*	number of users that u follows
follower#(u)	$Integer_s$	number of users who follow u
favorite#(u)	*Integer*	number of tweets favorited by u
tweets(u, T)	$Tweet_s$	set of tweets posted by u in T
tweet#(u)	*Integer*	number of tweets posted by u
tweetsD(u, T, D)	$Tweet_{ss}$	set of tweet sets (posted by u on $d \in D$)
users(T)	$User_s$	set of users in T
Tweet functions		
entity#(t)	*Integer*	number of entities in t (hashtag, url, mention, media)
hashtag?(t, h)	(0|1)	1 if $h \in$ hashtags(t), 0 otherwise
mention#(t)	*Integer*	number of mentions that occur in t
retweeted?(t)	(0|1)	1 if t is retweeted, 0 otherwise
mentions(t)	$Mention_s$	the set of mentions that occur in t
temporalTweets(T, Δ)	$Tweet_s$	set of temporal tweets in T based on Δ

4.1 User Features

User features are computed for each user in the tweet set of each *tracedHT*. The majority of these features represent how active and effective the user is. Besides, these features may be used to differentiate normal Twitter users from Bot/Cyborg/Hired Twitter users[11].

Average tweets/day: Higher values might indicate the behavior of an automated account.

$$\text{tweet\#}_{\mu/d}(u) = \frac{tweet\#(u)}{\text{today}() - \text{regDate}(u)} \tag{1}$$

Follower degree: Approaching 1 indicates a high degree of followers suggesting popularity, while values approaching to 0 indicate the opposite. Newly created bots tend to follow numerous users and have very few followers.

$$\text{follower-degree}(u) = \frac{\text{follower\#}(u)}{\text{follower\#}(u) + \text{following\#(u)}} \tag{2}$$

Entity use: Entities are used to gain attention. A higher rate would be expected in an act of spreading messages[5]. This value is calculated separately for hashtag, mention, url, and media entities.

$$\text{entity-use}(u, T) = \frac{\sum\limits_{t \in tweets(u,T)} entity\#(t)}{\mid tweets(u, T) \mid} \tag{3}$$

Traced hashtag use: Represents how focused the user is to the *tracedHT*. It is more likely that organized users concentrate on several hashtags.

$$\text{user-hashtag-use}(u, T) = \sum_{t \in tweets(u,T)} hashtag?(t, tracedHT) \tag{4}$$

Average daily tweets of *tracedHT*: Focuses on the daily use of the *tracedHT* based on user.

$$\text{tweet-hashtag\#}_{\mu/d}(u, D, T) = \frac{\sum\limits_{t \in \text{tweetsD}(u,T,D)} hashtag?(t, tracedHT)}{\mid D \mid} \tag{5}$$

Average Tweets/Day vs. Average daily tweets of *tracedHT*: Used in order to compare the user's general daily behavior to the days with *tracedHT*.

$$\text{user-daily-tweet-comparison}(u, D, ET) = \frac{\text{tweet-hashtag\#}_{\mu/d}(u, D, ET)}{\text{tweet\#}_{\mu/d}(u)} \tag{6}$$

User creation date: *userReg*(u), which can be useful in understanding of collective behavior of users (see Figure 2).

4.2 Temporal Features

Temporal features can be used to detect the synchronization since they focus on the characteristics of tweets which are posted in the same time interval I where :

$$T = \bigcup_{\substack{t \in allTweets \\ \text{hashtag?}(t,\, tracedHT) = 1}} \text{temporalTweets}(allTweets, I) \tag{7}$$

The majority of temporal features are calculated based on Twitter entities(hashtags, urls, images/videos), retweets, unretweets because these characteristics can be thought as indicators of synchronous behaviors. For example, features based on retweets can be sign of bot account existence and collective behavior, because in case of retweets there is no need to generate a content, which can be challenging for the automated accounts. Similarly, existence of too much unretweets may suggest organic behavior due to difficulty of creating an original tweet. Also, in order to propagate messages, bot users may use mentions to get attention of other users.

Entity use :

$$\text{entity-use}(T) = \frac{\sum_{t \in T} \text{entity\#}(t)}{\mid T \mid} \tag{8}$$

Temporal Tweet Per User (TPU): Higher values of TPU in T can be an indicator of organized behavior.

$$\text{temporal-tpu\#}(T) = \frac{\sum_{t \in T} 1}{\mid \text{users}(T) \mid} \tag{9}$$

[5] For more information about Twitter entities see: https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/entities-object1

Retweet frequency:

$$\text{retweet\#}(T) = \sum_{t \in T} retweeted?(t) \tag{10}$$

$$\text{retweet\%}(T) = \frac{\text{retweet\#}(T)}{|\ T\ |} \tag{11}$$

Unique retweeted frequency: The percentage of distinct retweets to see the diversity of retweets.

$$\text{original-retweeted-tweet\%}(T) = \frac{\left| \bigcup_{\substack{t \in T \\ retweeted?=1}} \{t\} \right|}{\sum_{t \in T} retweeted?(t)} \tag{12}$$

Retweeting user frequency: Show how many of the users in the temporal tweet set are participated in retweets.

$$\text{retweeted-users\#}(T) = \sum_{u \in users(T)} \left\lceil \frac{\sum_{t \in tweets(u,T)} retweeted?(t)}{|\ tweets\#(u,T)\ |} \right\rceil \tag{13}$$

$$\text{retweeted-users\%}(T) = \frac{\text{retweeted-users\#}(T)}{|\ \text{users}(T)\ |} \tag{14}$$

Unretweeted tweet frequency:

$$\text{unretweeted\%}(T) = 1 - \text{retweet\%}(T) \tag{15}$$

Users with no retweets frequency:

$$\text{unretweeted-users\%}(T) = 1 - \text{retweeted-users\%}(T) \tag{16}$$

Unretweeted tweet count:

$$\text{unretweeted\#}(T) =|\ T\ | - \text{retweet\#}(T) \tag{17}$$

Count of users with no retweets frequency:

$$\text{unretweeted-users\#}(T) =|\ \text{users}(T)\ | - \text{retweeted-users\%}(T) \tag{18}$$

Ratio of unretweets and users with no retweets:

$$\text{unretweeted-tweet\_user\_ratio}(T) = \frac{unretweeted\#(T)}{unretweeted - users\#(T)} \tag{19}$$

Mention Ratio: Frequency of mentions to distinct mentions:

$$\text{mention-ratio}(T) = \frac{|\ \bigcup_{t \in T} mentions(t)\ |}{\sum_{t \in T} mention\#(t)} \tag{20}$$

Ratio of mentions in retweets: Frequency of mentions that occur in retweeted tweets.

$$\text{mention-RT}(T) = \frac{|\ \bigcup_{\substack{t \in T \\ \text{retweeted?}(t)=1}} mentions(t)\ |}{\sum_{t \in T} mention\#(t)\ \text{retweeted?}(t)} \tag{21}$$

Ratio of mentions in unretweeted tweets: Frequency of mentions in tweets that are not retweeted.

$$\text{mention-notRT}(T) = \frac{|\ \bigcup_{\substack{t \in T \\ \text{retweeted?}(t)=0}} mentions(t)\ |}{\sum_{t \in T} mention\#(t) - \sum_{t \in T} mention\#(t)\text{retweeted?}(t)} \tag{22}$$

5 AN ORGANIZED BEHAVIOR DETECTION MODEL

The proposed model consists of two main phases: feature extraction and model generation (Figure 1). In the feature extraction phase, preparation of a collection and extracting its features (Algorithm 1) are performed. In the model generation phase, the random forest, SVM, and logistic regression algorithms are used to train models. In the training data set, each row represented a collection. In our study, a *collection* stands for a tweet set of interest. Tweets of interest are chosen to be those that contain a *tracedHT*. Feature extraction is performed on each collection, and these features are used to train classifiers.

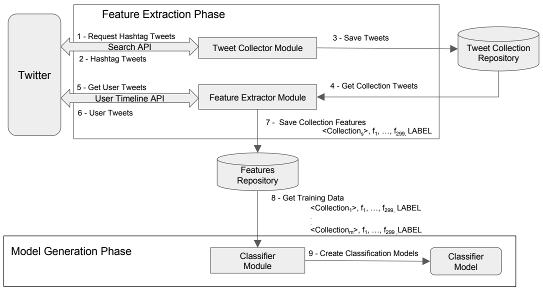
Feature Extraction Phase
Twitter
1 - Request Hashtag Tweets
Search API
2 - Hashtag Tweets
Tweet Collector Module
3 - Save Tweets
Tweet Collection Repository
4 - Get Collection Tweets
5 - Get User Tweets
User Timeline API
6 - User Tweets
Feature Extractor Module
7 - Save Collection Features
<Collection$_s$>, f_1, …, f_{299}, LABEL
Features Repository
8 - Get Training Data
<Collection$_1$>, f_1, …, f_{299}, LABEL
<Collection$_m$>, f_1, …, f_{299}, LABEL
Model Generation Phase
Classifier Module
9 - Create Classification Models
Classifier Model

Figure 1: General overview of implemented model.

To sum up Algorithm 1, in the feature extraction phase a collection is created by fetching tweets with *tracedHT* (hashtag of interest). Additional tweets of the users who posted in the collection are fetched to expand the collection in order to get more information about them. The resulting collection is referred to with a hashtag (i.e. #lockHerUp collection). Collection expansion consists of fetching additional tweets of users that were posted within a given time before and after the time of a post captured in *seed tweets*(ST) (line 4 in Algorithm 1). The aim of this is to understand whether there is a significant difference in the behavior of a user before and after the time of their post in *seed tweets*. In our experiments we chose this duration as one week. The expansion of *seed tweets* with the user tweets is referred to as the *expanded tweet set*(ET). After composition of *expanded tweet set*(ET), all the user and temporal features are extracted for the collection and stored in the *Features Repository*. In order to have an overall view of collections, the mean, variance, standard deviation, minimum, and maximum values for all the features are also computed and stored in the *Features Repository*. In our experiments time interval for temporal features was

Algorithm 1: Feature extraction algorithm applied to each tweet set.

```
Input: Hashtag tracedHashtag, Interval analysisInterval, Integer numDays
Output: List seedTweets, List allUserFeatures = [], List allTempFeatures = [], List expandedTweets = []
$seedTweets \leftarrow getTweets(hashtag = tracedHashtag)$
$users \leftarrow getUsers(tweets)$
for $u \in users$ do
    $expandedTweets.add(getTweets(u, numDays, seedTweets)$
end
for $u \in users$ do
    $userFeatures \leftarrow$ extractUserFeatures(u,tracedHashtag,expandedTweets)
    $allUserFeatures.add(< u, userFeatures >)$
end
$timeIntervals \leftarrow getTimeIntervals(tweets, analysisInterval)$
for $ti \in timeIntervals$ do
    $tweetsTI \leftarrow getTweets(expandedTweets, ti)$
    $temporalFeatures \leftarrow extractTemporalFeatures(tweetsTI)$
    $allTempFeatures.add(< u, temporalFeatures >))$
end
$featureStats \leftarrow$ computeFeatureStats(allUserFeatures,allTempFeatures)
$trainingDataRecord \leftarrow$ {allUserFeatures,allTempFeatures,featureStats}
return trainingDataRecord
```

chosen as one hour. Table 2 shows the sizes of *ST*, ET, the number of tweets used for temporal features extraction(#TFT)(Equation 7) and number of users for some collections [6].

Table 2: The number of tweets and users in some collections. The expansion of the seed collections result in significant increases in number of tweets.

Hashtag	#ST	#ET	#TFT	#Users
#imwithher	35,362	4,411,703	75,681	15,792
#maga	65,643	4,193,718	171,991	13,773
#crookedhillary	18,999	3,773,531	34,190	8,159
#oscarfail	2,111	749,748	2,152	1,507
#thanksgiving	4,575	539,081	4,461	2,367
#unitedairlines	1,345	276,228	2,240	898

6 EXPERIMENTS AND RESULTS

In this section, training data set and classification results are explained.

6.1 Data: Tweet Sets

In ["organic" vs."organized"] classification, a training data set with 851 records[7] (625 organized and 226 organic) is used, while in [political vs non-political] classification, a training data set with 879 records (231 non-political and 648 political) is used. In [pro-Trump vs pro-Hillary] classification, 854 records (311 pro-Trump, 171 pro-Hillary, and 371 None) are used for training[8].

6.1.1 Labeling Organized Tweet Sets. A hashtag is labeled as organized if there exists studies and news informing organized activity within the hashtag (i.e. #tcot & #pjnet [6]). Also, hashtags are labeled as organized, if they are used by large amount of bots [9] and there are studies proving it [8, 18]. Those hashtags formed our ground truth data sets for organized behavior[10]. After having collections of our ground truth hashtags, we inspected their *seed tweets* (*ST*) in order to gain insight about organized behaviors. For each *ST* of an organized hashtag, the following values are computed in order to gain overall characteristics of organized hashtags : the percentage of distinct words ($DW(\%)$), the average tweet count per user ($TPU(\mu)$), the percentage of retweets($RT(\%)$), the variance and standard deviation of hashtags ($HT(\sigma^2)$ and $HT(\sigma)$). By this way, externally discovered hashtags are compared to the ground truth collections and labeled as organized if their values coincide with the values of ground truth collections. Example comparisons are given in Table 3.

In Table 3, a low value of $DW(\%)$ indicates a lack of diversity in vocabulary, while a high value of $TPU(\mu)$ shows that the users who posted in the collection tend to repeatedly post the same hashtag. High values of $HT(\sigma^2)$ & $HT(\sigma)$ indicate the use of multiple hashtags in the collection, which can be observed in the so-called viral activity. Similarly, high $RT(\%)$ values indicate less original content.

[6] Sizes of the all collections in data set: https://github.com/Meddre5911/DirenajToolkitService/blob/master/organizedBehaviorDataSets/OrganizedBehaviorDataSetSizes.csv

[7] Each record reflects the features of one collection

[8] Training data sets can be found in https://github.com/Meddre5911/DirenajToolkitService/tree/master/organizedBehaviorDataSets/trainingDataSets

[9] In the first presidential debate, 32.7 percent of pro-Trump hashtags and 22.3 percent of pro-Hillary hashtags were posted by bots[18]

[10] Some of the hashtags in our ground truth sets are #benghazi, #obamacarefail, #imvotingbecause, #draintheswamp, #trumpwon, #clintonemails, #auditthevote, and #hillaryemails.

Table 3: Tweet sets characteristics inspected during manual inspection.

Hashtag	tweets	DW	TPU	RT	HT	HT
	(#)	(%)	(μ)	(%)	(σ^2)	(σ)
#podestaemails15	17,890	4.62	1.89	92.28	2.40	1.54
#BoycottHamilton	56,523	3.92	1.54	2.12	2.72	1.65
#StrongerTogether	13,581	11.9	1.64	77.06	0.46	0.68
#unitedairlines	54,506	8.13	1.28	71.86	0.75	0.86
#womansDay	416,350	7.57	1.38	78.76	0.13	0.36

Note that, these parameters are only used for inspection of tweet sets. In order to prevent overfitting in classification results, they are not used in the training data set.

Besides from using the parameters in Table 3, same user amount in different tweet sets with similar tweets are also taken into account in labeling process. In this step, tweet sets are thought similar if inspection of their tweets reveal that they have a common point based on content and targeted audience. Table 4 shows the users who participated in multiple pro-Trump hashtags in different times. An externally discovered hashtag is labeled as organized, if at least 20% of its users also participated in one of the similar collections. This is due to the observation that colluding users are active over time and use several persisting hashtags to manipulate others. Increasing of users' percentage after July 2015[11] in Figure 2 is an example to this.

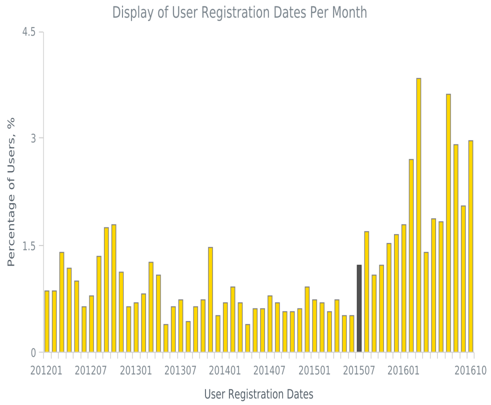

Figure 2: Percentage of same users in Table 4 based on Twitter registration date (users in #benghazi & #crookedHillary)

6.1.2 *Labeling Organic Tweet Sets .* Hashtags are labeled as organic when they are deemed to be spontaneously posted and they emerge due to events like holidays, natural disaster, and news about popular people. Based on these two criteria, the labeling process is done after manually inspection of hashtag tweets. For example, #unitedairlines, #boycottUnitedAirlines hashtags instantly became viral after a video of passenger, who was forcibly removed from a plane due to over booking. Similarly, in the 2017 Oscars, the #oscarsfail hashtag became top trending after the best picture award was accidentally given to wrong movie.

Table 4: Same User Comparison of an organized collection.

Traced hashtag	#benghazi
Time Interval	24 Oct - 1 Nov 2016
User#	7, 854

Collections with mutual users with #benghazi				
Traced Hashtag	**Interval**	**User#**	$\bigcap$ (#)	$\bigcap$ (%)
podestaemails	17-24 Oct	32, 794	3, 232	41.15
crookedhillary	04-10 Nov	16, 854	2, 291	29.17
boycotthamilton	17-22 Nov	36, 561	1, 580	20.12
maga	20-28 Nov	29, 130	3, 063	39.00

Table 5: Same User Comparison of an organic collection.

Traced hashtag	#internationalwomensday
Time Interval	07 – 09 March 2017
User#	267, 695

Collections with mutual users with #internationalwomensday				
Traced Hashtag	**Interval**	**User#**	$\bigcap$ (#)	$\bigcap$ (%)
Thanksgiving	20-28 Nov	104, 060	6, 247	6
Oscars	25 Feb-01 Mar	134, 868	12, 656	9.38
NationalPetday	09-14 Apr	39,506	3, 095	7.83
MothersDay	13-15 May	113, 225	8, 207	7.25

Before labeling a hashtag as organic, we inspected *seed tweets* of the hashtag suspecting the incidence of bots hijacking[2]. Recall that during the labeling of organized hashtags, we labeled hashtags as organized in case of excessive bot existence. In order to understand the bot account existence in the candidate organic hashtags, we examined characteristics of their *seed tweets* by checking statistical values in Table 3. This control was necessary since hashtags like #Thanksgiving, #LaborDay, #WomansDay, etc. are used by many users and it would be possible that they might be bots. Table 5 shows overlapping users between organic hashtags, and their percentage is very low compared to those in organized hashtags in Table 4. Therefore, hashtags like #Thanksgiving, #LaborDay, #NationalSiblingsDay, and #WomansDay are tagged as *organic* in our study.

6.1.3 *Tweet Sets of Other Categories.* Besides from organic and organized classification, additional two different classifications such as *pro-Trump* vs. *pro-Hillary*, and *political* vs. *non-political* are also applied on tweets of *tracedHTs* . The labeling of political vs. non-political and pro-Trump vs. pro-Hillary hashtags are done either by using given pro-Trump and pro-Hillary hashtags in the studies [17, 18] or by manually inspecting *ST*. The determination of the label was fairly straightforward in comparison of labeling organic vs. organized behavior.

[11] The official nomination date of Trump for presidency.

Table 6: Training Data Set Explanations

Training Data Set	Explanation
Training Data Set 1	All features
Training Data Set 2	Principal components of features in Data Set 1
Training Data Set 3	All features except those based on traced Hashtag (see Equations 4,5,6)
Training Data Set 4	Principal components of features in Data Set 3

Table 7: Organic vs. Organized Classifications

Features	Method	A	F	ROC
Data Set 1	**Random Forest**	0.99	0.99	0.99
	Logistic Regression	0.99	0.99	0.99
	SVM	0.75	0.64	0.66
Data Set 2	**Random Forest**	0.98	0.97	0.96
	Logistic Regression	0.98	0.98	0.98
	SVM	0.99	0.99	1.00
Data Set 3	**Random Forest**	0.99	0.99	0.99
	Logistic Regression	0.99	0.98	0.97
	SVM	0.75	0.64	0.66
Data Set 4	**Random Forest**	0.97	0.96	0.95
	Logistic Regression	0.98	0.98	0.97
	SVM	0.99	0.99	0.99

6.2 Results of Three Classifers

For each classification, four different training data sets are generated(Table 6). In Data Set 3 and 4, it is aimed to test performance of proposed approach in case of tweets are not collected based on hashtags and *tracedHT* related features (Section 4) are not used. Each classifier model is evaluated using 10-fold cross validation. Tables 7, 8, and 9 show the results for each category[12]. In the tables, column F stands for F-Measure, column A stands for Accuracy, and ROC stands for Receiver Operating Characteristic. Results show that the random forest algorithm results in high scores with full features, while logistic regression and SVM algorithms give better results when PCA is applied.

The most important five features[13] are given in Table 10 and 11 [14]: It is surprising that none of the temporal features are in top five, suggesting that organized behavior detection could be achieved by extracting user features only. These top user-based features are also reported in other studies[9, 10], and this is encouraging. Compared to temporal features, extracting user features is easier. This suggests that a classifier based on only user features might be useful.

Table 8: Political vs. Non-Political Classifications.

Features	Method	A	F	ROC
Data Set 1	**Random Forest**	0.99	0.99	0.99
	Logistic Regression	0.99	0.99	0.99
	SVM	0.77	0.65	0.64
Data Set 2	**Random Forest**	0.99	0.98	0.97
	Logistic Regression	0.98	0.98	0.98
	SVM	1.00	0.99	1.00
Data Set 3	**Random Forest**	0.99	0.99	0.99
	Logistic Regression	0.99	0.99	0.99
	SVM	0.77	0.65	0.64
Data Set 4	**Random Forest**	0.98	0.98	0.98
	Logistic Regression	0.99	0.98	0.99
	SVM	1.00	0.99	1.00

Table 9: Pro-Hillary vs. Pro-Trump vs None classification. P is precision, and R is recall.

Features	Method	A	F	P	R
Data Set 1	**Random Forest**	0.97	0.96	0.97	0.96
	Logistic Regression	0.44	0.20	0.15	0.33
Data Set 2	**Random Forest**	0.92	0.90	0.93	0.88
	Logistic Regression	0.94	0.93	0.93	0.93
Data Set 3	**Random Forest**	0.96	0.95	0.96	0.94
	Logistic Regression	0.44	0.20	0.15	0.33
Data Set 4	**Random Forest**	0.92	0.89	0.93	0.87
	Logistic Regression	0.93	0.91	0.92	0.91

Table 10: Top 5 features in Data Set 1 Classifications

Organic vs. Organized	1) σ^2 of media-use(u, C) [Eq. 3] 2) Maximum of mention-use(u, C) of all users [Eq. 3] 3) μ of follower-degree(u) [Eq. 2] 4) μ of $10.001 \le favorite\#(u) \le 20.000$ 5) Maximum of $tweet\#(u)$
Political vs. Non-Political	1) σ^2 of media-use(u, C) [Eq. 3] 2) Maximum of mention-use(u, C) of all users [Eq. 3] 3) μ of follower-degree(u) [Eq. 2] 4) μ of $10.001 \le favorite\#(u) \le 20.000$ 5) Maximum of $tweet\#(u)$
pro-Hillary vs. pro-Trump vs. None	1) % of users with $0.0001 \le$ media-use$(u, C) \le 0.5$ 2) % of users with url-use$(u, C) = 1$ [Eq. 3] 3) % of users with mention-use$(u, C) = 7$ [Eq. 3] 4) σ^2 of url-use(u, C) [Eq. 3] 5) % of users with $\frac{\text{mention-use}(u, C)}{\text{tweet\#}_{\mu/d}(u)} = 1$ [Eq. 6]

7 DISCUSSION

The aim of the present work was to develop a basic model for detecting organized behavior on Twitter. Limited resources also played a role in keeping it simple, however, it is interesting to learn how simple approaches perform. All three classification models (organized vs organic; political vs non-political; and pro-Trump vs pro-Hillary vs None) are trained with the same features and all

[12] Since Spark MLlib does not support SVM multi-classification,SVM results are not provided for pro-Trump vs pro-Hillary classifications.

[13] Features are selected using the *ClassifierSubsetEval* attribute evaluator with the Random Forest classifier and the Best First search method in the Weka[1] by applying 10 fold cross validation

[14] C refers to a collection and $U = users(C)$

Table 11: Top 5 features in Data Set 3 Classifications

Organic vs. Organized	1) # of users with $0.6 \leq$ media-use$(u, C) \leq 0.9$[Eq. 3] 2) # of users with hashtag-use$(u, C) = 7$ [Eq. 3] 3) # of users with mention-use$(u, C) = 10$ [Eq. 3] 4) Minimum of follower-degree(u) [Eq. 2] 5) # of users such that $1 \leq favorite\#(u) \leq 100$
Political vs. Non-Political	1) σ^2 of media-use(u, C) [Eq. 3] 2) μ of follower-degree(u) [Eq. 2] 3) % of users with $10.001 \leq favorite\#(u) \leq 20.000$ 4) % of users with $10.001 \leq tweet\#(u) \leq 20.000$ 5) Maximum of $tweet\#(u)$
pro-Hillary vs. pro-Trump vs. None	1) % of users with mention-use$(u, C) = 0$ [Eq. 3] 2) % of users with $0.0001 \leq$ media-use$(u, C) =\leq 0.5$ 3) % of users with $0.0001 \leq$ mention-use$(u, C) \leq 0.5$ 4) % of users with url-use$(u, C) = 1$ [Eq. 3] 5) % of users with $1 \leq favorite\#(u) \leq 100$

yield promising results. Based on these results, we speculate that the proposed features may fingerprint tweet collections. However, to be sure, more work needs to be done. Also, results taken for Data Set 3 and 4 show that collection of tweets based on a *tracedHT* is not essential. Other mechanisms such as community detection[5] algorithms or topic detection[25] methods may be also used for tweet collection.

It could be that our data set is too small even though it contains millions of tweets[15]. On the other hand, user features might indeed be most significant ones by shining light on the characteristic of those who are in collusion. A classifier model generated solely based on user features may be worth developing, since it is fairly easy and cost effective, which would be beneficial for real-time classification.

Besides from the user and temporal features in the model, we also examined other features that are related to tweet content, user relations, and information flow patterns. We observed the presence of unusually similar content posted by the same people or those connected to them during the same interval, presumably serving a shared agenda. However, similarity computation among all tweets set is very costly (complexity of $O(n^2)$)– a task that exhausted our resources. Efficient approaches to compare large sets of posts is an interesting research direction.

Community detection and closeness centralities of users can be also used to understand organized behavior. However, the computation of closeness centrality becomes a challenge for real time detection when the followers of followers of thousands of people are considered In popular context, such as political campaigns, follow relations exceeding 50K are not uncommon.

Another observation is information flow patterns, for example recurring message paths, such as user A tweets t_x, which is retweeted by B, which is retweeted by C. Here the users A, B, C remain the same, wheras the message t_i may vary. Such pattern suggests presence of coordination, automated behavior, however it has an overhead in computation. Because of these reasons, we did not use features related to content similarity, user closeness centrality, information flow of tweets, although they remain of great interest. The features we did include are those we considered to be significant and whose computation was in the realm of our resources.

For real time detection systems, there is a need for in-memory big data tools like Apache Spark[26], which supports terabyte-scale data processing. Our repository is more than 480 gigabytes with over 200 million tweets, summary tables, indexes. A Spark cluster can process all that data in memory.

8 CONCLUSIONS

This work proposes a supervised learning model for automatically classifying organized behavior patterns. Models are trained with user and temporal features extracted from over 200 million tweets that were mostly gathered during the 2016 US presidential election. Three types of classifications were performed among the categories: [organic,organized], [political, non-political], and [pro-Trump,pro-Hillary,None]. In each case, the random forest algorithm consistently resulted in high accuracy and f-measure scores with an average of 0.95. The results of classifying tweet sets suggest that neither content nor user relation features are required to successfully classify them. Furthermore, that user features are the most significant regarding our classification tasks. Further investigated with larger training data sets should be perform to further validation.

ACKNOWLEDGMENTS

This work is partially supported by the Turkish State Planning Organization (DPT) under the TAM Project, number 2007K120610. The members of the Complex Systems Lab (SosLab) at the Computer Engineering Department of Bogazici University are thanked for their constructive comments and technical support during this work.

REFERENCES

[1] [n. d.]. Weka Description. http://www.cs.waikato.ac.nz/ml/weka/. ([n. d.]). Accessed: 2018-02-21.

[2] Norah Abokhodair, Daisy Yoo, and David W McDonald. 2015. Dissecting a social botnet: Growth, content and influence in Twitter. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing (CSCW '15)*. ACM, ACM, New York, NY, USA, 839–851. 2675133.2675208.

[3] Ethem Alpaydin. 2010. *Introduction to Machine Learning* (2nd ed.). The MIT Press.

[4] Michael Ashcroft, Ali Fisher, Lisa Kaati, Enghin Omer, and Nico Prucha. 2015. Detecting Jihadist Messages on Twitter. In *Proceedings of the 2015 European Intelligence and Security Informatics Conference (EISIC)*. IEEE Computer Society, Washington, DC, USA, 161–164.

[5] Mohamed Bakillah, Ren-Yu Li, and Steve HL Liang. 2015. Geo-located community detection in Twitter with enhanced fast-greedy optimization of modularity: the case study of typhoon Haiyan. *International Journal of Geographical Information Science* 29, 2 (2015), 258–279.

[6] J.M. Berger. 2016. Nazis vs. ISIS on Twitter: A Comparative Study of White Nationalist and ISIS Online Social Media Networks. (2016).

[7] J.M. Berger and Jonathon Morgan. 2015. The ISIS Twitter Census, Defining and describing the population of ISIS supporters on Twitter. (2015). https://www.brookings.edu/wp-content/uploads/2016/06/isis_twitter_census_berger_morgan.pdf.

[8] Alessandro Bessi and Emilio Ferrara. 2016. Social bots distort the 2016 U.S. Presidential election online discussion. *First Monday* 21, 11 (2016). fm.v21i11.7090.

[9] Cheng Cao and James Caverlee. 2015. *Detecting Spam URLs in Social Media via Behavioral Analysis.* Springer International Publishing, 703–714.

[10] Cheng Cao, James Caverlee, Kyumin Lee, Hancheng Ge, and Jinwook Chung. 2015. Organic or Organized?: Exploring URL Sharing Behavior. In *Proceedings of the*

[15] For details see: https://github.com/Meddre5911/DirenajToolkitService/blob/master/organizedBehaviorDataSets/OrganizedBehaviorDataSetSizes.csv

24th ACM International on Conference on Information and Knowledge Management. ACM, New York, NY, USA, 513–522.

[11] Zi Chu, Steven Gianvecchio, Haining Wang, and Sushil Jajodia. 2012. Detecting automation of twitter accounts: Are you a human, bot, or cyborg? *IEEE Transactions on Dependable and Secure Computing* 9, 6 (2012), 811–824.

[12] Frank Edwards and Philip N. Joyce Howard. 2013. Digital Activism and Non Violent Conflict. (2013). Available at SSRN: https://ssrn.com/abstract=2595115.

[13] Emilio Ferrara, Onur Varol, Clayton Davis, Filippo Menczer, and Alessandro Flammini. 2016. The Rise of Social Bots. *Commun. ACM* 59, 7 (June 2016), 96–104. http://doi.acm.org/10.1145/2818717 10.1145/2818717.

[14] Nicolas Foucault and Antoine Courtin. 2016. Automatic Classification of Tweets for Analyzing Communication Behavior of Museums. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis (Eds.). European Language Resources Association (ELRA), Paris, France.

[15] Shahla Ghobadi and Stewart Clegg. 2015. "These Days Will Never Be Forgotten ...": A Critical Mass Approach to Online Activism. *Inf. Organ.* 25, 1 (Jan. 2015), 20. j.infoandorg.2014.12.002.

[16] Philip N. Howard and Bence Kollanyi. 2016. Bots, #StrongerIn, and #Brexit: Computational Propaganda during the UK-EU Referendum. (2016). http://arxiv.org/abs/1606.06356

[17] Bence Kollanyi and Philip N. Howard. 2016. Bots and Automation over Twitter during the Second U.S. Presidential Debate. (2016). http://politicalbots.org/wp-content/uploads/2016/10/Data-Memo-Second-Presidential-Debate.pdf.

[18] Bence Kollanyi, Philip N. Howard, and Samuel C. Wooley. 2016. Bots and Automation over Twitter during the First U.S. Presidential Debate. (2016). https://www.oii.ox.ac.uk/blog/bots-and-automation-over-twitter-during-the-first-u-s-presidential-debate

[19] Georgiy Levchuk, Lise Getoor, and Marc Smith. 2014. Classification of group behaviors in social media via social behavior grammars. In *SPIE Defense+ Security*. International Society for Optics and Photonics, 909707–909707. 12.2050823.

[20] Jonathan A Obar. 2013. Canadian Advocacy 2.0: An Analysis of Social Media Adoption and Perceived Affordances by Advocacy Groups Looking to Advance Activism in Canada. (2013). Available at SSRN: https://ssrn.com/abstract=2254742.

[21] Jacob Ratkiewicz, Michael Conover, Mark R Meiss, Bruno Gonçalves, Alessandro Flammini, and Filippo Menczer. 2011. Detecting and Tracking Political Abuse in Social Media. *ICWSM* 11 (2011), 297–304.

[22] Craig Silverman and Lawrence Alexander. 2016. How Teens In The Balkans Are Duping Trump Supporters With Fake News. https://www.buzzfeed.com/craigsilverman/how-macedonia-became-a-global-hub-for-pro-trump-misinfo. (November 2016). Accessed: 2018-02-21.

[23] Apache Spark. [n. d.]. Machine Learning Library (MLlib). https://spark.apache.org/docs/1.1.0/mllib-guide.html. ([n. d.]). Accessed: 2018-02-21.

[24] Prashanth Vijayaraghavan, Soroush Vosoughi, and Deb Roy. 2016. Automatic Detection and Categorization of Election-Related Tweets. In *Tenth International AAAI Conference on Web and Social Media*.

[25] Ahmet Yıldırım, Suzan Üsküdarlı, and Arzucan Özgür. 2016. Identifying topics in microblogs using Wikipedia. *PloS one* 11, 3 (2016), e0151885. journal.pone.0151885.

[26] Matei Zaharia, Reynold S. Xin, Patrick Wendell, Tathagata Das, Michael Armbrust, Ankur Dave, Xiangrui Meng, Josh Rosen, Shivaram Venkataraman, Michael J. Franklin, Ali Ghodsi, Joseph Gonzalez, Scott Shenker, and Ion Stoica. 2016. Apache Spark: A Unified Engine for Big Data Processing. *Commun. ACM* 59, 11 (Oct. 2016), 56–65. 10.1145/2934664.

[27] Nina Zumel and John Mount. 2014. *Practical Data Science with R* (1st ed.). Manning Publications Co., Greenwich, CT, USA.