Understanding Bots on Social Media -

An Application in Disaster Response

by

Tahora Hossein Nazer

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved November 2019 by the
Graduate Supervisory Committee:

Huan Liu, Chair
Hasan Davulcu
Ross Maciejewski
Leman Akoglu

ARIZONA STATE UNIVERSITY

December 2019

ABSTRACT

Social media has become a primary platform for real-time information sharing among users. News on social media spreads faster than traditional outlets and millions of users turn to this platform to receive the latest updates on major events especially disasters. Social media bridges the gap between the people who are affected by disasters, volunteers who offer contributions, and first responders. On the other hand, social media is a fertile ground for malicious users who purposefully disturb the relief processes facilitated on social media. These malicious users take advantage of social bots to overrun social media posts with fake images, rumors, and false information. This process causes distress and prevents actionable information from reaching the affected people. Social bots are automated accounts that are controlled by a malicious user and these bots have become prevalent on social media in recent years.

In spite of existing efforts towards understanding and removing bots on social media, there are at least two drawbacks associated with the current bot detection algorithms: general-purpose bot detection methods are designed to be conservative and not label a user as a bot unless the algorithm is highly confident and they overlook the effect of users who are manipulated by bots and (unintentionally) spread their content. This study is trifold. First, I design a Machine Learning model that uses content and context of social media posts to detect actionable ones among them; it specifically focuses on tweets in which people ask for help after major disasters. Second, I focus on bots who can be a facilitator of malicious content spreading during disasters. I propose two methods for detecting bots on social media with a focus on the recall of the detection. Third, I study the characteristics of users who spread the content of malicious actors. These features have the potential to improve methods that detect malicious content such as fake news.

DEDICATION

To my family.

ACKNOWLEDGMENTS

I would like to thank my advisor, Huan Liu, for his guidance and support throughout the course of my PhD. I have been truly blessed to have his support in developing my research and facing all the challenges that I face in the past few years. His wisdom not only lightened my academic path but also helped me find my way when in life situations that seemed unbearable at the time. I would like to thank my committee members, Hasan Davulcu, Ross Maciejewski, and Leman Akoglu for their valuable feedback. They helped me discover new angles of my thesis topic and approach the problems with new tools and techniques.

Being a member of the Data Mining and Machine Learning Lab (DMML) has been a unique environment for me to learn how to collaborate, be a team player, deliver on-time, and present my work clearly and effectively. I had the opportunity to be a mentee, an independent researcher, and a mentor. I collaborated with researchers at other schools of Arizona State University (ASU) and my funding agencies and nourished the skills to turn real-world problems to Machine Learning tasks. I thank all my labmates at DMML and all my collaborators at ASU, National Science Foundation, and Charles River Analytics.

My research was not possible without the financial support of my funding agencies. Specifically, support was provided, in part, by National Science Foundation grant 1461886 and the Office of Naval Research through N000141310835 and N000141612257. I would like to thank the members of the Data Mining and Machine Learning Lab (DMML) at Arizona State University and my committee members for their valuable feedback.

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Chapter 1

INTROCUDTION

Using social media to reflect on disasters goes back to September 11th, 2001 (the 9/11 terrorist attacks on the World Trade Center towers and the Pentagon), when the families of affected people used wikis to collect information on the missing people. Additionally, FEMA and Red Cross used web technologies to inform people with status updates (Palen and Liu, 2007). Since then, there have not been any major disaster that was not widely reflected on social media (see Figure 1.1 for major events since 2011). During the destructive Haiti Earthquake, 2010, US government agencies used social media technologies as the main knowledge sharing medium between themselves, the government of Haiti, and the United Nations (Yates and Paquette, 2010). During Hurricane Sandy, 2012, more than twenty million tweets were published (Gupta *et al.*, 2013) and citizens handled activities that were unlikely to be done by official emergency responders such as recovering lost animals (White *et al.*, 2014). In the course of Hurricane Harvey, 2017, victims turned to social media to call for help. They used hashtags such #SOSHarvey and #helphouston to ask for rescuers and volunteers compiled a list of names and addresses on social media to respond to these cries for help. Devastation of victims shown in some of the social media posts resulted in thousands of re-shares and this made rescuing these victims a top priority for rescuers[1].

Social media provides a platform for emergency responders to acquire situational awareness through the content shared by the users who act as social sensors (Sakaki *et al.*, 2010). This awarness is in the form of a big picture of the event or actionable

[1]http://time.com/4921961/hurricane-harvey-twitter-facebook-social-media/

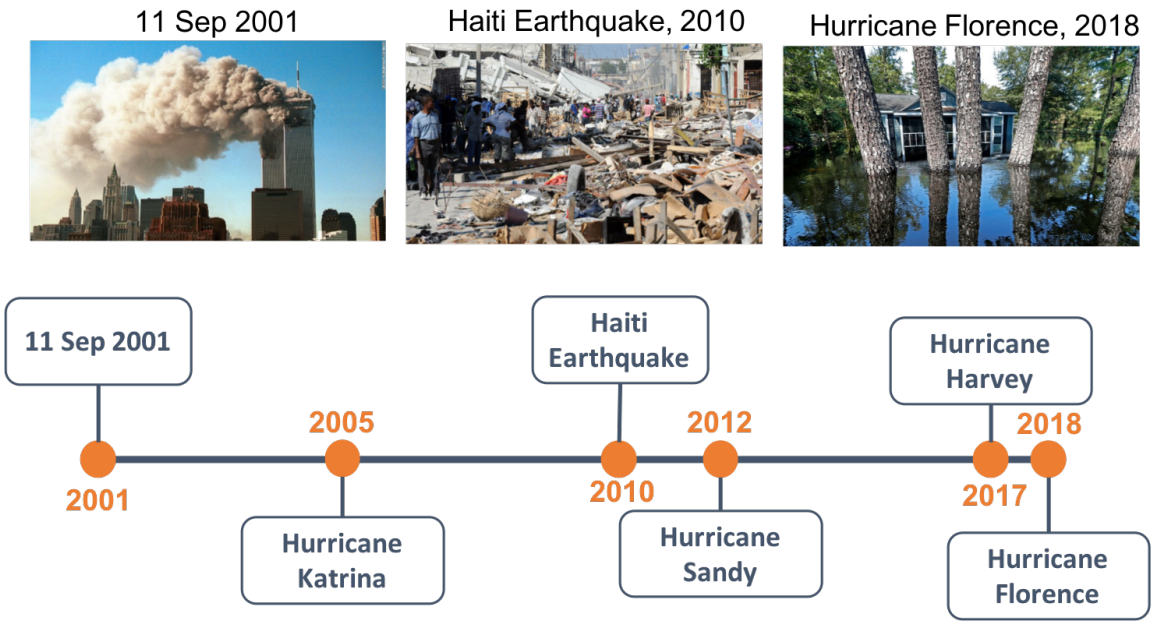


Figure 1.1: Major Disasters That Have Been Widely Reflected on Social Media Since the 9/11 Attacks in 2011.

insights (Castillo, 2016). Preliminary assessment of a disaster such as the area which was affected, the number of casualties, and the failed infrastructures are obtained in the "big picture". "Actionable insights" entail more details such as requests for resources such as water, gas, and electricity. Many systems have been developed to benefit and organize the content and volunteer efforts on social media. Ushahidi (Okolloh, 2009) is the first large-scale crowdsourcing system for data collection, visualization, and filtering in disaster relief. OpenStreetMap is one of the systems that allows volunteers contribute in generating open source maps by marking entities such as roads and buildings. These maps have been used for disasters such as Haiti earthquake, 2010 (Zook *et al.*, 2010). Artificial Intelligence for Disaster Response (AIDR) (Imran *et al.*, 2014) is another system that can be trained on a small sample of labeled tweets to detect tweets related to categories such as shelter and food. TweetTracker (Kumar *et al.*, 2011) is a system built at Arizona State University for tracking, analyzing, and understanding tweets related to a specific topic.

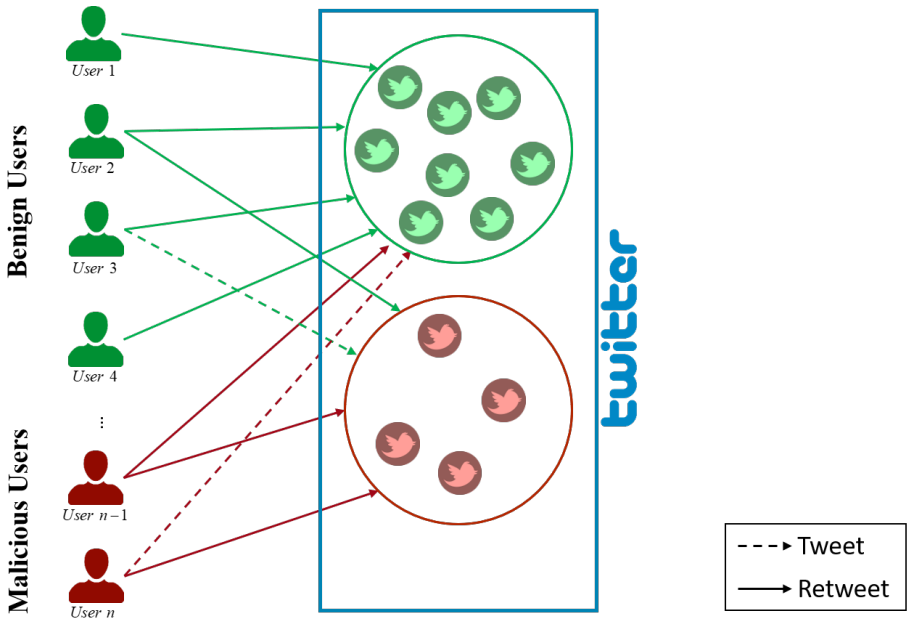


Figure 1.2: Interactions on Social Media. Both Benign and Malicious Users Generate Content (Tweet) and Share the Content of Others (Retweet). Our Focus Is Malicious Content (Red Tweets in the Figure), Malicious Actors (Red Users), and Benign Users Who Spread the Content of Malicious Actors (Green Users Who Tweet/Retweet Red Tweets).

Achieving situational awareness through the lens of social media is valuable but challenging. One challenge is that disaster related posts are indulged in huge number of irrelevant posts such as donation scams, fake news, and rumors. This issue has become more pronounced since bots have been used to facilitate the spread of these malicious content. In June 2014, 8.5% of all accounts used third-party applications to automate their activities on Twitter. However, this number is increasing as a study in 2017 estimated that between 9% and 15% of Twitter users are bots (Varol *et al.*, 2016). Bots change how information diffuses on social media; they accelerate the spread of rumors and increase the number of people exposed to rumors by 26% (Vosoughi *et al.*, 2018).

The second challenge is neutralizing the effect of social media users who (mostly unintentionally) boost the effect of malicious actors such as bots. A goal of exploiting

facilitators such as bots is increasing the reach of some target content. This goal is not achieved unless some of the social media users participate in the diffusion process. Hence, it is important to understand who these users are (their features and characteristics) and what motivates them to spread such content. A deeper understanding of these users sheds light on the factors contributing to the spread of malicious content and approaches this problem from a new angle.

In this work, we focus on three aspects of social media interactions in course of disasters, as shown in Fig.1.2. First, we study the content that becomes available and how we can extract actionable insights from them. Second, we explore the role of malicious actors, such as bots, on social media and how we can detect them. Third, we examine the users who help spread the content of malicious actors, their characteristics, and motivations.

1.1 Social Media for Disaster Response

In this section, we provide specific examples of how machine learning systems empowered by social media have been used in disaster relief and response.

- Humanitarian OpenStreetMap Team (HOT) is a non-governmental organization that helps generate maps of disastrous locations by indicating damaged and intact buildings, blocked and open roads, and the locations of key infrastructure such as hospitals. HOT was used in large scale during the Haiti Earthquake, 2010, and Typhoon Haiyan, 2013, to generate and continually update the maps of affected areas . HOT was officially tasked by UN Office for the Coordination of Humanitarian Affairs (UNOCHA) and the Philippine Red Cross before the landfall. During the typhoon, UNOCHA and also a medical aid group Médecins Sans Frontières (also known as Doctors Without Borders) used the maps generated by volunteers in HOT (Butler, 2013). Moreover, UN Volunteers

(UNV) posted opportunities for online volunteers to geotag Twitter messages and images from the affected area to map urgent needs of the population (Bonn, 2013).

- Standby Task Force is digital volunteering organization that was set up in 2010 and incorporated as a not for profit in 2014. This organization has been involved in relief and response efforts during disasters such as Hurricanes Matthew, Harvey, Irma, and Maria and the 2015 Nepal Earthquake. Standby Task Force provides crisis maps that have overlays of requests for assistance and damage reports. To generate these maps, social media posts and reports from local and national communities are collected, filtered, and categorized then verified and geo-tagged by volunteers on the map. Crisis maps generated by this organization were used by Federal Emergency Management Agency (FEMA) during Hurricane Maria providing the status of hospitals and roads in Puerto Rico (Proctor and Dalchand, 2017); supported the United States Coast Guard in the search and rescue efforts during Hurricane Irma (Proctor, 2017); collaborated with Humanity Road during the 2015 Nepal Earthquake by deploying near real-time maps of the disaster that benefited from Artificial Intelligence for Disaster Response (AIDR) (Imran *et al.*, 2014); filtered social media posts and community reports (Proctor, 2015).

- Humanity Road (HR) is a nonprofit organization formed in 2010 to help with the disaster response efforts through "social listening". This organization is one of the seven companies that were recognized by the FEMA Tech Sector in the Tech Corps program[2] which is a nationwide program of skilled technology vol-

[2]https://www.fema.gov/news-release/2015/06/17/fema-launches-innovative-national-volunteer-program-enhance-disaster

unteers who help complement the tech aspect of the relief efforts[3]. Technology organizations in the Tech Corps help federal, state, local and tribal governments incorporate innovative technologies in disaster relief and response. This organization has assisted with several disasters such as the 2015 Nepal Earthquake, the 2016 Kumamoto Earthquake in Japan, 2018 Woolsey Fire, and Hurricanes Harvey, Irma, Maria, and Michael. During the Nepal Earthquake, HR volunteers data mined social media posts to find urgent needs and situational awareness information. Moreover, HR exploited machine learning tools such as Scanigo [4] to reduce the size of the social media dataset by removing irrelevant information in order to extract situational information from tweets; in this process, Scanigo filtered 1.2 million Nepal related tweets to just 4,638 which drastically reduced the human annotation effort required. Humanity Road also collaborated with the Data Mining and machine learning Lab at Arizona State University to examine the opportunities and challenges in using social media data for disaster repose (Abbasi *et al.*, 2012).

- Kathmandu Living Labs (KLL) is a Nepali-based nonprofit technology company focused on open-source mapping founded in 2013. This organization benefits from up to 2,400 volunteers per day during disasters who use social media, satellite imaging, and observation reports to generate maps of the area hit by disaster. KLL started with detailed mapping of the Kathmandu Valley and expanded their efforts during the 2015 Nepal Earthquake by project QuakeMap.org. In this project, information is collected using the required resources about the affected people with their locations and road blocks. Misplaced people were then

[3]https://www.humanityroad.org/our-blog/humanity-road-joins-fema-tech-corps

[4]https://www.sbir.gov/sbirsearch/detail/833717

added to the maps[5]. These maps were used by the Red Cross and the Nepali Army during the Nepal Earthquake (Sinha, 2015).

1.2 Bots in the Course of Disasters

Bots, during the natural disasters of 2017, Mexico Earthquake and Hurricanes Harvey, Irma, and Maria, used disaster-related hashtags to promote political topics such as `#DACA` and `#BlackLivesMatter` (Khaund *et al.*, 2018). These bots change the network structure by forming clusters that are different from ones formed by the connections between human users. Bots form large clusters with dense cores consisting of bots and loose connections to humans. Human formed clusters, on the other hand, are smaller in size but more tightly knit. They also share fake news such as the hoax story of shark swimming on the streets after Hurricanes Harvey and Irma.

News related to natural and man-made disasters (such as terrorism and war) are the seventh and third most prominent topics among all news topics spread on Twitter from 2006 and 2017. In a study on the effect of bots on the spread of these topics (Vosoughi *et al.*, 2018), the researchers observed that bots drastically change the dynamics of diffusion. Bots accelerate the spread of rumors and increase the number users exposed to them by about 26%. Bots also change the structure of diffusion cascades by increasing their depth by 21% and maximum breadth by 26%.

In another work (Abokhodair *et al.*, 2015), a botnet that was active during the Syrian civil war in 2012 was studied. These bots used two tactics to sway away the attention from the discussions on civil war: misdirection and smoke screening. In misdirection, bots tweeted about political and natural crisis happening worldwide while including keywords related to the Syrian civil war to point attention from it. In smoke screening, bots talked about other events in Syria using relevant hashags

[5]http://www.kathmandulivinglabs.org/projects/quakemaporg

#syria (in English and Arabic) but the content was not about the civil war. Both misdirection and smoke screening tweets can shift the attention of users from the Syrian civil wars if tweeted in large numbers. Bots in this study remained active more than 6 months while having major differences from human users. For instance, they tweeted up to 5,733 tweets per week (one tweet in about 1.8 minutes), mainly tweeted about news (52.6%), as opposed to humans, and did not express personal opinion.

More than 11,000 users posted about Paris Attacks and Umpqua Community College Shootings in Fall 2015 on Twitter. Researchers (Nied *et al.*, 2017) formed the network of these users by establishing a link between every two users if they shared at least 5% of their network. Of the clusters found in this network, three botnet clusters were detected. These bots, although consisting of 10% of users (150 bots), generated more than one third of all the tweets. These bots spread alternative narratives. The prominent alternative narrative about the Umpque shooting was an image meme of four pictures of people from different crisis events who looked similar, asserting that they were the same person, a "crisis actor" hired by the government to stage the event. For the Paris attacks, bots spread the alternative narrative that the French government was staging false evidence to blame the Syrian refugees for this disaster.

Chapter 2

RELATED WORK

In this chapter we introduce the previous research on the topic of this dissertation. We will discuss the background on the problem of bot detection on social media in the first half of this chapter. The topics in this part help the readers understand the importance of studying bots on social media, categories of bots and their intents, and the methods proposed to detect bots. In the second half, we discuss how disasters are reflected on social media, the potential applications of social media for disaster response, and the systems developed towards this goal.

2.1 Prevalence of Bots on Social Media

Existence of large number of bots on social media and specially Twitter has been reported by multiple sources. In 2013, Twitter announced that about 5% of its users are fake (Elder, 2013). The Wall Street Journal reported (Koh, 2014) in March 2014 that half of the accounts created in 2014 were suspended by Twitter due to activities such as aggressive following and unfollowing behaviors which are known characteristics of bots (Lee *et al.*, 2011). In 2017, Varol et al. reported that between 9% to 15% of users on Twitter exhibit bot behaviors (Varol *et al.*, 2016). In 2018, Twitter suspended 70 million accounts (Timberg and Dwoskin, 2018) in an effort towards fighting fake news and suspicious accounts. This mass removal happened after the congressional pressure on social media companies to prevent harmful acts of organized fake accounts that affected the US presidential election in 2016 (Jane, 2016). This swarm of bots rising on social media have wide range of malicious effects. An extensive list of news articles on bots in major event is presented in Table 2.1.

Table 2.1: News Articles on Bots on Social Media During Major Events.

Outlet	Category	Event	Data Source
The Guardian[1]	Elections	US Presidential Election 2016	Twitter
Time Magazine[2]	Elections	US Presidential Election 2016	NBER
Bloomberg[3]	Elections	Italian General Election 2018	Atlantic Council
The Telegraph[4]	Referendum	Brexit	Twitter
Politico Magazine[5]	Propaganda	#ReleaseTheMemo Movement 2018	-
The Guardian[6]	Propaganda	Russia-Ukraine Conflict 2014	-
CNN[7]	Propaganda	Jamal Khashoggi's Death 2018	Atlantic Council
The Telegraph[8]	Fake News	Stock Market - FTSE 100 Index 2018	-
Forbes[9]	Ad Fraud	AFK13 Attack	White Ops Firm
The New York Times[10]	Mass Shooting	Florida School Shooting 2018	New Knowledge Co.
Fox News[11]	Mass Shooting	Texas School Shooting 2018	Bot Sentinel Co.
The Telegraph[12]	Terrorist Attack	Westminster Terror Attack 2017	-
Medium[13]	Natural Disasters	Mexico City Earthquake 2017	-

Bots remain active for a long time and affect a large number of users. In an experiment, researchers (Freitas *et al.*, 2015) created a set of 120 Twitter bots. These bots targeted different groups of users (in terms of diversity and how well they were connected) and could gain 4,999 follows from 1,952 distinct users, and 2,128 message-based interactions from 1,187 distinct users while 69% of these bots could not be detected by Twitter in 30 days. In an study of the 2010 US midterm elections on Twitter (Ratkiewicz *et al.*, 2011b), authors observed that a group of bots smearing Chris Coons, the Democratic candidate for U.S. Senate from Delaware by promoting specific URLs. These bots further targeted popular users who were active on the topic and mentioned them in their promotional tweets. When targeted users receive the same content from multiple sources the probability of involving in the cascade increases (Hasher *et al.*, 1977). Another example of political content promotion happened during the US presidential election in 2016. Bots produced 1.7 billion tweets and successfully deviated the discussion by outnumbering the tweets supporting one candidate by the factor of four (Kelion and Silva, 2016); millions

of fake stories supporting each candidate were shared on Facebook and again the balance is 4 to 1 supporting the same candidate (Allcott and Gentzkow, 2017). These findings raise the question whether fake news did (Jane, 2016) or did not (Allcott and Gentzkow, 2017) affect the election results.

2.2 Malicious Versus Benign Bots

Bots on social media have both positive and negative impact. Malicious bots are intentionally exploited to perform harmful activities on social media. Malicious bots sway the discussions (Ratkiewicz *et al.*, 2011a,b; Thomas *et al.*, 2012) and cause users to lose trust that social media platforms can deliver news honestly. Major search engines take social media into account when ranking the webpages on the results pages; more popularity on social media causes a company to appear higher in the search results. Researchers argue that some companies exploit fake likes and fake followers to increase their revenue (Clark, 2015). Furthermore, bots impinge the work researchers perform on social media as they can draw false conclusions about the populations under study. Researchers wish to understand human behavior through the lens of social media (Mejova *et al.*, 2015), and this is often impinged by the wealth of content pollution created by automated social media accounts (Wu *et al.*, 2017).

Benign bots, on the other hand, help users automate consuming and broadcasting information. These bots have benevolent purposes, self-declare as bots, and do not disguise themselves among human users. @BBCWeatherBot, @earthquakesLA, @big_ben_clock are examples of benign bots that broadcast weather, earthquakes in Los Angeles, CA, and announce time on the hour respectively. Moreover, celebrities, political figures, and businesses also use bots to publish content and respond to customers' questions. Frankfort Airport flight information bot, @FRA_Flightinfo,

and Spirit Airline Twitter account, @SpiritAirlines use bots to post real-time flight information and automatically reply to customer's comments.

2.3 Categories of Malicious Bots

Lee et al. (Lee *et al.*, 2010) studied social spammers on Twitter and found that a large portion of these spammers were bots. They observed five categories among these malicious users. *Duplicate Spammers* post almost identical tweets promoting a message. They use the mention (i.e `@`username mechanism to target specific Twitter users. *Pornographic Spammers* have graphical profiles or post adult content. *Promoters* increase the visibility of marketing messages and advertise businesses. However, they mix their message with innocuous tweets to disguise among benign users. These bots latch trending hashtags to their messages to push them higher in the social media search results. One example is using disaster-related hashtags to promote political content in during the hurricanes in 2017 (Khaund *et al.*, 2018). The effect of Promoters has been also observed in political content promotion. In an study of the 2010 US midterm elections on Twitter (Ratkiewicz *et al.*, 2011b), authors observed that a group of bots smearing Chris Coons, the Democratic candidate for U.S. Senate from Delaware by promoting specific URLs. These bots further targeted popular users who were active on the topic and mentioned them in their promotional tweets. Another example of political content promotion happened during the US presidential election in 2016. Bots produced 1.7 billion tweets and successfully deviated the discussion by outnumbering the tweets supporting one candidate by the factor of four (Kelion and Silva, 2016); millions of fake stories supporting each candidate were shared on Facebook and again the ratio was 4 to 1 supporting the same candidate (Allcott and Gentzkow, 2017). These findings raise the question whether fake news did (Jane, 2016) or did not (Allcott and Gentzkow, 2017) affect the election results. *Phishers*

deliver phishing URLs to targeted users. *Friend Infiltrators* actively connect to other users on social media hoping that they reciprocate as courtesy and remove the links if they do not. Note that aggressive following behavior is against Twitter's rules and policies[14]. Friend Infiltrators seek followers to form a large network for spreading their messages and increasing the popularity of specific social media figures. Having large number of followers and balancing the number of followers and friends help malicious bots blend in among human users (Lee *et al.*, 2011). There are many services that sell fake followers such as `buycheapfollowerslikes.org` and `audiencegain.com` for cheap prices. The New York Times reported on July 13, 2018, after Twitter removed millions of suspicious users, known as the Twitter purge, many popular figures lost thousands of their followers. Celebrities, political figures, and world leaders observed a decrease in their followers up to 6%, many of which possible to be bots (Harris *et al.*, 2018).

In another categorization (Subrahmanian *et al.*, 2016), bots are categorized into three groups: *spambots*, *paybots*, and *influence bots*. Spambots spread irrelevant and or inappropriate spam tweets. Twitter's definition of spamming includes activities such as posting links to phishing and malware sites, creating multiple accounts, and aggressive follow behaviors[15]. Paybots perform illegitimate activities in an attempt to gain financial profit. For example, they follow political figures and celebrities to boost their popularity and increase their influence on social media (Harris *et al.*, 2018). Influence bots impact users active on and conversations about specific topics. Influence bots during the 2016 US presidential election where used to tweet about one candidate and this activity changed the statistics of discussions about that candidate on Twitter by the factor of four (Kelion and Silva, 2016).

[14]https://help.twitter.com/en/safety-and-security/report-spam

[15]https://help.twitter.com/en/safety-and-security/report-spam

2.4 Characteristics of Malicious Bots

Bots are different from human users in their profile, activities, and connections. Here we enumerate a set of characteristics of malicious social bots.

- Content of posts and messages: Bots tend to use more URLs in their posts for promotional purposes (Xie *et al.*, 2008). Sentiment of tweets (Ratkiewicz *et al.*, 2011b,a), language and length of messages (Thonnard and Dacier, 2011), number of mentions and similarity between all pairs of tweets by a user (Lee *et al.*, 2011), and originality of tweets (Wang, 2010) can all differentiate bots from humans.

- Profiles and activities: Automation involved in generating bot accounts results in profiles that contain detectable patterns. Bots that were working together towards political disruption in the 2011 Russian parliamentary election were found to have similar email addresses and account creation times (Thomas *et al.*, 2012).

  Bots have a shorter life time (Lee *et al.*, 2011), short subscriber details (Cook *et al.*, 2014), and use hijacked IP addresses (Thomas *et al.*, 2012). A bot master would also spread the machines from which he launches his attacks geographically (Kanich *et al.*, 2008).

  Even when profile information is not available, the username alone can be a bot indicator. The length of the screen name (Lee *et al.*, 2011) and the distance between the distribution of n-grams extracted from verified names and bot names (Lee and Kim, 2014) can differentiate bots from normal users. Matching how well the screen name matches human typing patterns can also discriminate automated users (Zafarani and Liu, 2015).

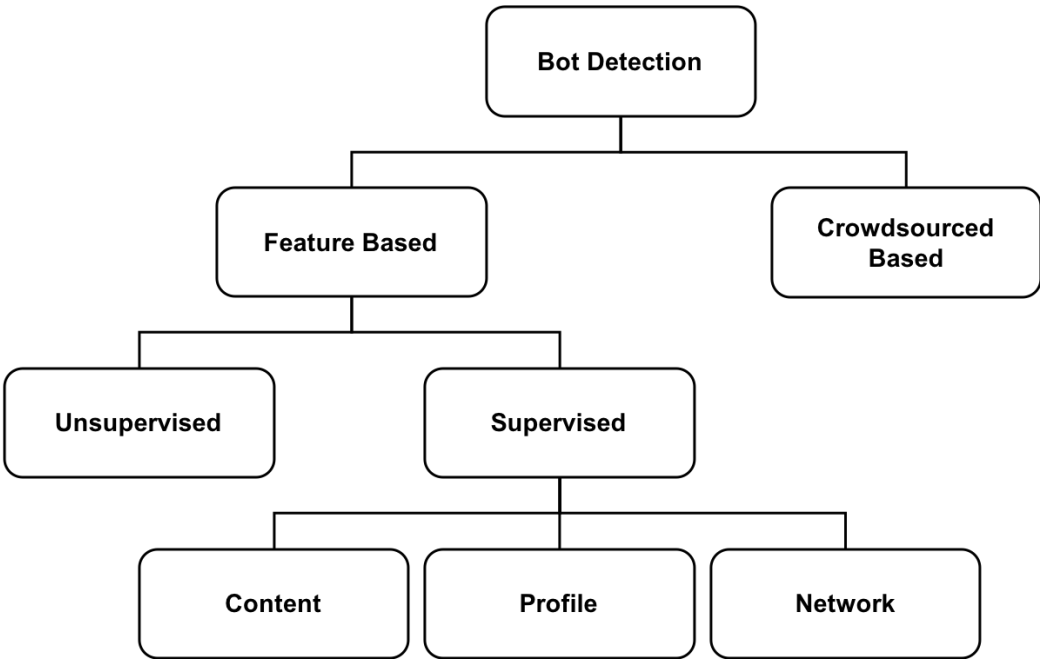


Figure 2.1: A Taxonomy of Bot Detection Methods.

- Network structure and connections: To boost their desired effect, Twitter bots follow normal users hoping to be followed back. To this aim, they exhibit mass follow and unfollow behaviors and thus their connections show different characteristics. The number of followers, number of friends, ratio of the the friends to followers, percentage of bidirectional friends, and the standard deviation of unique numerical IDs of followers and friends have been used to detect bots (Lee *et al.*, 2011).

2.5 Bot Detection Models

The prevalence of bots on social media has encouraged researchers to propose combating methods. There are two categories of methods proposed for bot detection on social media: supervised and unsupervised (shown in Figure 2.1). Supervised methods proved to be effective but require reliable ground-truth datasets for training. Unsupervised methods for bot detection rise to this challenge and propose ideas to

use unique universal characteristics of human users or the similarity between groups of users to discriminate bots from human users. In this work we introduce some of state-of-the-art unsupervised bot detection methods that are proposed in recent years, categorize them, and mention some of the challenges they face.

Researchers have studied bots on social media during numerous major events such as natural disasters, man-made disasters such as Syrian civil war and mass shootings, and political events such as US Presidential Election or referendums in other countries. An extensive list of such research papers is presented in Table 2.2.

Table 2.2: Activity Areas of Bots on Twitter

Reference	**#Tweets**	**#Users**	**#Bots**	**Event**
(Khaund *et al.*, 2018)	1,219,454	776,702	100[16]	Natural disasters
(Vosoughi *et al.*, 2018)	4.5M	3M	396K	Natural Disasters[17]
(Abokhodair *et al.*, 2015)	150K	-	-	Syrian civil war
(Starbird, 2017)	77,461	15,150	2,696	Mass Shootings
(Nied *et al.*, 2017)	32K	11K	1.1K	Paris Attacks
(Kitzie *et al.*, 2018)	36,627[18]	120K	6,051	Parkland shooting 2018
(Shao *et al.*, 2017)	11,656	915	54	Low-credibility Content
(Bessi and Ferrara, 2016)	12.6	50K	7,183	US Presidential Election 2016
(Shao *et al.*, 2018b)	27,648,423	630,368	-	US Presidential Election 2016
(Shao *et al.*, 2018a)	13,617,425	-	-	US Presidential Election 2016
(Stella *et al.*, 2018)	3.6M	523K	174K	Catalan referendum 2017
(Broniatowski *et al.*, 2018)	9,845	8,289	190	Vaccine Debate
(Allem *et al.*, 2017)	412,816	365,490	-	E-Cigarette Discussions

2.5.1 Supervised Methods

Ferrara et al. (Ferrara *et al.*, 2016) proposed a taxonomy of bot detection models which divides them into three classes: (1) graph-based, (2) crowdsourcing, and (3) feature-based social bot detection methods.

- Graph-based Methods

  Graph-based social bot detection models lie on the assumption that the connectivity of bots is different from human users on social media. *SybilRank* (Cao *et al.*, 2012) is a graph-based method proposed to efficiently detect adversary-owned bot accounts based on the links they form. The underlying assumption is that bots are mostly connected to other bots and have limited number of links to human users. Hence, if we start short random walks from a set of trusted users in the network, there is a higher probability that we land on a human user rather than a bot. Bots also show different characteristics in the communities they form. In a study on the bots that were active during the natural disasters in 2017, Khaund et al. (Khaund *et al.*, 2018) observed that bots form more hierarchical communities with cores of bots strongly connected to each other and peripheral members who are weakly connected to the core and to each other. Moreover, human users had more communities and their communities were more tightly knit.

- Crowdsourcing Methods

  Crowdsourcing social bot detection uses human annotators, expert and hire workers, to label social media users as human or bot (Wang *et al.*, 2013). This method is reliable and has near zero error when the inter annotator agreement is considered. However, it is time consuming, not cost effective, and not feasible considering millions of users on social media. Crowdsourcing and manual anno-

tation are still being used as methods for collecting gold standard datasets for feature based bot detection models, most of which use supervised classification.

- Feature-based Methods

  Feature based social bot detection methods are based on the observation that bots have different characteristics than human users. To use feature-based supervised bot detection models, one must identify differences among bot and human users in terms of features such as content or activity in a labeled dataset. Then, a classifier is trained on the features and labels to distinguish bots from humans in an unobserved dataset. Different classification methods can be used for this purpose such as Support Vector Machines (Morstatter *et al.*, 2016), Random Forests (Lee *et al.*, 2011), and Neural Networks (Kudugunta and Ferrara, 2018). We describe some common user features below:

  - Content: the measures in this category focus on the content shared by users. Words, phrases (Varol *et al.*, 2017), and topics (Morstatter *et al.*, 2016) of social media posts can be a strong indicator of bot activity. Also, bots are motivated to persuade real users into visiting external sites operated by their controller, hence, share more URLs in comparison to human users (Chu *et al.*, 2012; Ratkiewicz *et al.*, 2011a; Xie *et al.*, 2008). Bots are observed to lack originality in their tweets and have large ratio of retweets/tweets (Ratkiewicz *et al.*, 2011b).
  - Activity Patterns: Bots tweet in a "bursty" nature (Chu *et al.*, 2012; Lee and Kim, 2014), publishing many tweets in a short time and being inactive for a longer period of time. Bots also tend to have very regular (e.g. tweeting every 10 minutes) or highly irregular (randomized lapse) tweeting patterns over time (Zhang and Paxson, 2011).

– Network Connections: bots connect to a large number of users hoping to receive followers back but the majority of human users do not reciprocate. Hence, bots tend to follow more users than follow them back (Chu *et al.*, 2012).

2.5.2 Unsupervised Methods

There are two categories of unsupervised methods for bot detection on social media. In the first category bots are discovered based on the similarities they share in botnets. The idea is that members of a botnet, as much as they try to camouflage, share commonalities because to their common purpose, creator, actions, or content. In the second category, the goal is uncovering individual bots. This goal is achieved by extracting unique universal features of human users on social media in terms of their activity patterns. For example, the irregularity in their tweeting behavior; users are indifferent in the second-of-the-minute and minute-of-the-hourthey tweet at. We enumerate some of the state-of-the-art methods in the remainder of this section.

- Group-based methods

  Chavoshi (Chavoshi *et al.*, 2009) et al. proposed a criteria based on which groups of users whose activities are abnormally aligned are detected. They estimate the probability that two or more users in a group of users tweet/retweet in a window of w seconds from each other during an hour. They show that even when users are very active (they tweet every 20 seconds), the probability that two users have forty or more matching tweets in an hour is close to zero. They use this criteria to find groups of users who have at least 40 tweets in an hour and consider the users whose activities are highly correlated as bots. They evaluate their method on a dataset of Twitter users by finding clusters of similar user in every hour, merging them, and picking the top ten clusters for evaluation.

They compared the 9,134 bot accounts discovered with five other bot detection approaches including the Twitter suspension method and BotOMeter (Davis *et al.*, 2016) system. Twitter suspended 45% of the discovered bots in 12 weeks and BotOMeter agrees on the 59% of the bots.

In an extension to their previous method, Chavoshi et al. (Chavoshi *et al.*, 2016) use a lag-sensitive hashing method to cluster the suspicious users and find groups of bots based on the similarity of their activities over time. The proposed systems is constructed using four modules: (1) the steam of tweets is collected from the Twitter API using a set of keywords (2) users who have more than one tweet in T hours are selected; using the number of tweets in each second, a time series signal for each user is generated. (3) each signal is hashed into 2w+1 hash indexes (4) using a hierarchical clustering method, users are grouped based on a warped correlation distance measure; users who fall into clusters are considered bots and the singletons are considered false positives; clustering continues until the distance between the clusters to be merged is less than a threshold. In step (3), a random signal r is selected as the base and w is the maximum lag in the time series (two signals are similar if lagging one for w or -w seconds generates the other). Then, the hashed version of an activity signal is calculated as the cross-correlation between r and that signal. This hashed signal is then lagged for w seconds in each direction to generate 2w+1 hashed versions of the original signal. The users that are selected for clustering are the ones with more than w/4 occurrences in a hash bucket if that bucket contains at least w/4 such users. The evaluation is similar to their previous work.

One approach (Chen and Subramanian, 2018) exploits a clustering mechanism

on users based on the shortened URLs they include in their tweets. This method monitors all the tweets that contain a URL shortened by one of the most popular URL shortening services on Twitter. If a group with twenty or more Twitter users all post the same shortened URL, that group is marked as suspicious. The users in a suspicious group who mostly (60% of their 200 most recent tweets) post the most frequent (top 3) tweets in that group form a botnet. After the botnets are recognized, the authors further investigate the owners of the webpages that are pointed to by the shortened URLs in the botnets. If one email address is associated with multiple botnets, those botnets are considered as a spamming campaign. In another approach (Ahmed and Abulaish, 2013), three groups of features are used in a clustering based method for detecting bot campaigns: interaction features (number of Twitter followers or Facebook friends), hashtagging (total number of hashtags and hashtagging rate of a profile), and URLs features (total and unique URLs shared by a profile). Then a weighted graph of users based on the similarity between every two users is generated and is given as the input to a Markov clustering algorithm (Van Dongen, 2008). Clusters that were discovered using this method act as advertisement campaigns, hijack legitimate accounts for spamming purposes, and show the same pattern of generating tweets.

Cresci et al. (Cresci *et al.*, 2016) proposed a bio-inspired unsupervised method for detecting bots on Twitter. Each user is represented by two DNA sequence, one that encodes tweeting activity pattern (tweet, reply, or retweet) and another one encoding content type (e.g. tweets with URLs or hashtags). Ordered sequence of activities of each user is represented by these two DNA sequences. They observed that the Longest Common Subsequence (LCS) of DNA sequences in botnets is much larger than groups of legitimate users. In a dataset of bots

mixed with legitimate users, plotting the LCS for groups of different size shows breakpoints that indicate size of each group of botnet. In botnet groups, LCS remains large even when we increase the group size.

- Individual-based methods

  Both methods (Ahmed and Abulaish, 2013; Chen and Subramanian, 2018) mentioned above are unsupervised methods designed for detecting malicious botnets who are similar in terms of content or network. However, benign bots, as opposed to malicious bots are mainly individual bots designed to provide a service[19]. Hence, to discriminate benign and malicious bots we require unsupervised methods with the capability to label users based on the activity patterns as we expect benign bots to show automation characteristics.

  Another approach (Zhang and Paxson, 2011) is based on the differences between the activity patterns of humans and bots on social media. This study shows that human users are indifferent towards the minute-of-the-hour and second-of-the-minute that they tweet at. Hence, their tweeting timestamps appears to be drawn from a uniformly random distribution across minute-of-the-hour and second-of-the-minute. On the other hand, bot accounts' timestamps were observed to be either non-uniform or excessively uniform. Non-uniform patterns happen when bots tweet at specified times or intervals leading the timestamps to be concentrated at a specific minute-of-the-hour and second-of-the-minute. Excessively uniform scenarios happen when bots use a fixed delay after each tweet. This uniform pattern is too regular to be from humans. They test uniformity of tweeting timestamps using χ^2 test with 0.001 as threshold.

[19] Although there are bots that are controlled by the same organization such as Twitter bots that belong to USGS and report earthquakes on different locations.

The VolTime algorithm (Chino *et al.*, 2017) is a generative model based on the inter-arrival time and volume of activities. Action at time t_i of a user is presented using an event tuple (Δ_i, v_i) showing the time between between this action and the one on time t_{i-1} and the volume v_i. Volume can be the length in characters (for textual content) or duration (for phone calls) depending on the dataset. Once actions of users are formulated, the expected dispersion of all events is calculated. Dispersion in a set of events is the number of unique events. For example, the multiset of events (1, 1), (1, 3), (1, 1) has dispersion equal to 2. Finally, the users whose events' multiset has low dispersion (using a predefined threshold) are marked as bots.

BotWalk (Minnich *et al.*, 2017) ensembles four unsupervised methods for detecting bots. The authors generate a representation of each social media user by four feature categories: metadata-, content-, temporal-, and network-based features. Then, they ensemble the anomalous scores generated by four unsupervised anomaly detection algorithms: density-, distance-, angle-, and isolation-based methods. Anomalous sores generated by these four algorithms are scaled using a Gaussian distribution to produce a probability p(x) between 0 and 1 of the user x being an outlier. For the evaluation, they perform manually annotations on a sample of detected bots and their features and compare the detected bots with DeBot (Chavoshi *et al.*, 2016) and BotORNot (Davis *et al.*, 2016) models.

Chapter 3

EXTRACTING ACTIONABLE INFORMATION FROM SOCIAL MEDIA

Social media has become an important channel for crisis communication. Social media is a fast-paced channel for people to describe their situation and observations, specify their needs, and offer assistance; providing actionable information. Officials also use social media in disaster response; for example, as shown in Figure 3.1, Red Cross asks for donations using its Twitter account after major disasters.

(a) (b)

Figure 3.1: Two Tweets of Red Cross Asking for Donation After Hurricanes Harvey and Irma.

People connect via social networks after disasters to seek information (Palen and Vieweg, 2008) and post their requests (Vieweg *et al.*, 2010). This actionable information is useful for first responders to better distribution resources when the size and location of needs are unknown and the environment is uncertain and hard to predict (Zeimpekis *et al.*, 2013). One approach to help the posts with actionable information stand out among millions of tweets, is taking advantage of hashtags. In

the aftermath of the Chennai Rains in late November 2015, `#ChennaiRainsHelp` was used by stranded people to request for help and by volunteers to offer accommodations (Joshi, 2015). After the Paris Attacks in November 2015, volunteers in the US used `#PorteOuverte` (OpenDoor) and `#StrandedInUS` to offer a place to stay and help the French people whose flights were canceled (Goel and Ember, 2015).

Although hashtags play an important role in finding the posts which contain actionable information, not all such posts are marked with proper hashtags. In the Boston Marathon Bombings in April 2013, no consistent hashtag was offered by official organizations (Sutton *et al.*, 2014). In such cases, valuable information is neglected because of sole dependence on hashtags. A trivial way to overcome this issue is manual inspection which is not feasible due to the large number of tweets flooding in. These issues have encouraged the development of systems that automatically detect actionable information and requests of the affected people.

Multiple systems have been developed to automate the process of extracting actionable information from user-generated (crowdsourced) data. One of the first large scale attempts is Google Flu Trends in 2008. Based on the location and time of flu related searches, Google predicted the outbreak of flu seven to ten days before the Centers for Disease Control and Prevention (Graham and Zook, 2011). Other systems are Ushahidi (Okolloh, 2009), AIDR (Imran *et al.*, 2014), and TweetTracker (Kumar *et al.*, 2011) that were discussed in Section 3.1.1.

Here, we focus on a subset of actionable information shared on social media which needs immediate attention. More specifically, we propose a method to extract the request for help on Twitter (we would call them *requests* in the remainder of this section) by exploiting content and context of tweets. Our results show that context (i.e. metadata of tweets) provides a powerful signal while it is easier to collect and process.

3.1 Disaster Response Using Social Media

The International Federation of Red Cross and Red Crescent Societies (IFRC) defines disasters as sudden events that disrupt the functionality of a community such that the losses go beyond the resources of that community [1]. Disasters have eight socio-temporal stages: Pre-disaster, Warning, Threat, Impact, Inventory, Rescue, Remedy, and Recovery (Powell, 1954). The volume of social media posts varies in each stage; majority of users start posting after the disaster onsets and the frequency decreases when the disaster reaches its final stages. Availability of data is a major factor in building automatic methods for facilitating disaster management and response. Hence, we study four stages in disasters that are widely reflected on social media and we have enough data for Machine Learning methods to achieve reliable results: Warning, Impact, Response, and Recovery (see Figure 3.2).

In the warning stage, social media can be used as a complementary source of information to help increase the confidence in predicting disasters and providing warnings. Changes in the frequency of posts with specific words and topics, activity patterns of users (Asur and Huberman, 2010; Sampson *et al.*, 2015; Tumasjan *et al.*, 2010), and sentiment of posts (Mishne *et al.*, 2006) are used to predict disasters. Predicting disasters before they hit an area provides the opportunity to warn people in danger and evacuate elevators and operation rooms. Currently, USGS uses tweets to check the accuracy of sensor reports and detect earthquakes in a shorter time. Earthquakes can be detected using tweets by 60 seconds earlier than sensors; this time is valuable for warning areas in danger and starting evacuation processes (Ellis, 2015).

When disasters impact an area, social media posts show anomalies such as changes

[1] https://www.ifrc.org/en/what-we-do/disaster-management/about-disasters/what-is-a-disaster/

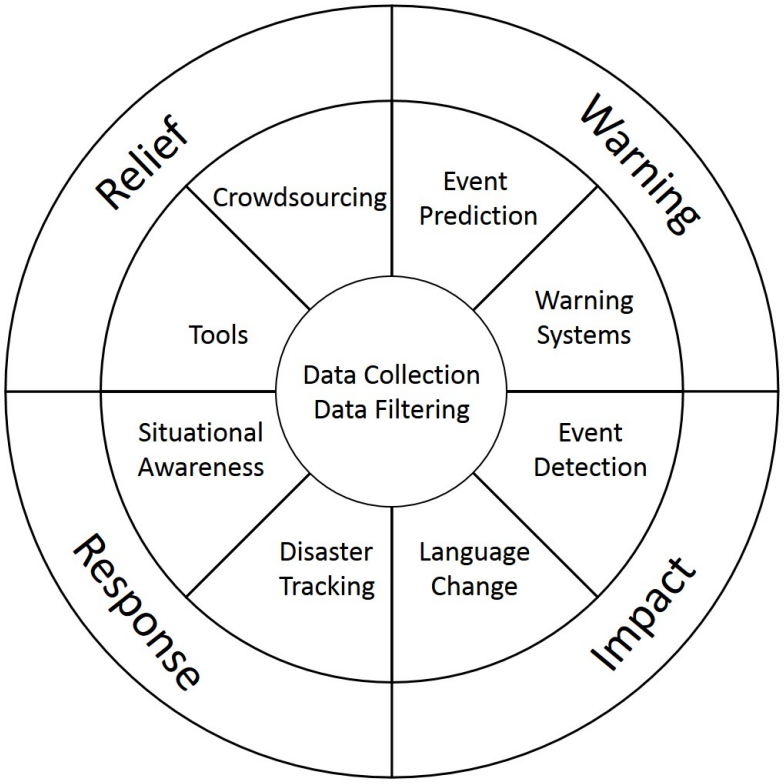


Figure 3.2: Socio-Temporal Stages of Disasters Which Are Reflected on Social Media.

in the language. A study (Cohn *et al.*, 2004) on LiveJournal after 11 September 2001 shows that emotional positivity decreased and cognitive processing, social orientation, and psychological distancing increased after the attack (Beigi *et al.*, 2016). These changes in social media posts can be quantitatively captured in sentence level or topic level. Capturing the change in real-time results in detecting disasters before they are announced by official sources, governmental websites, or major news outlets (SAMBULI, 2013).

In response to the chaotic environment caused by disasters, emergency responders want to acquire actionable insight and a big picture of the disaster (Castillo, 2016). Detecting and tracking topics, trends, and memes on social media provides information regarding the status of disasters and the affected people. Damages, casualties,

missing animals, and failed structures are some of the topics that people discuss on social media. Tracking these topics, discovering the trends, and monitoring mentioned locations help responder distribute resources more efficiently .

Volunteers are significantly important in the relief process. They post information that increases situational awareness (e.g. status of roads and damages to built structures) and provide technical support for translating social media posts and geotagging them. Some of the systems that exploit social media posts to facilitate disaster management are Ushahidi (Okolloh, 2009), AIDR (Imran *et al.*, 2014), and TweetTracker (Kumar *et al.*, 2011).

3.1.1 Machine Learning Systems for Disaster Response

Social media is a unique platform for collaboration between remote volunteers. These volunteers provide technical services such as translation, geolocating posts on the map, and generating maps of the affected area. Several Machine Learning tools have been developed to exploit crowdsourcing on social media for facilitating volunteering actions during disasters.

Ushahidi

Ushahidi (Okolloh, 2009) is the first large-scale crowdsourcing system for disaster relief. It has been initially developed to map the reports of Kenyan post-election violence in 2008 and since then has been used in many major disasters such as Hurricane Sandy and Haiti Earthquake. Ushahidi is an open source and free systems which can either be deployed on external servers or on Ushahidi's hosting system CrowdMap. When technical knowledge or hosting servers are not available, CrowdMap is a more suitable.

Ushahidi has three main sections: data collection, visualization, and filtering. As

the first step, disaster-related data is collected from several sources, web, Twitter, RSS feeds, emails, SMS, and manual comma separated files. The user-contributed information is then visualized on the map. Each point on the map shows one report and when a user zooms out, aggregated number of reports in each area is represented. As the last step, Ushahidi allows users to filter reports based on their types, e.g. supplies or shelter. An snapshot of this system is shown in Figure 3.3.

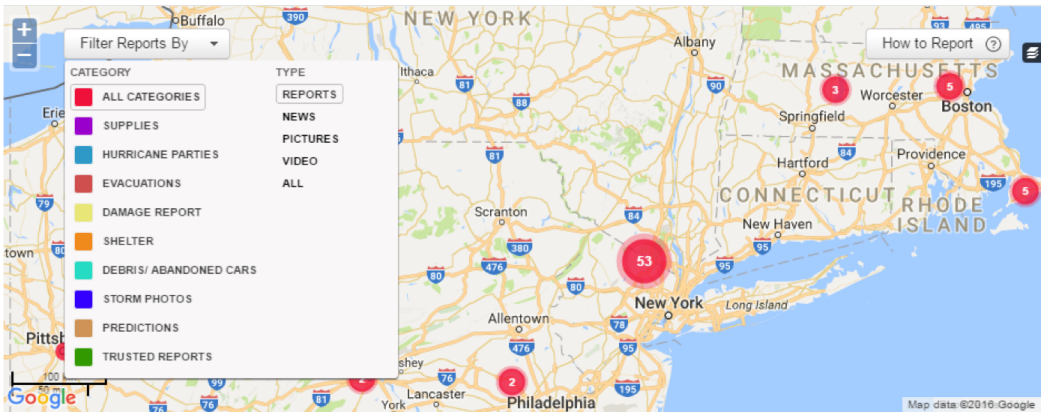


Figure 3.3: A Snapshot of Ushahidi System (Okolloh, 2009).

AIDR

Artificial Intelligence for Disaster Response (AIDR) (Imran *et al.*, 2014) is a free software platform which can be either run as a web application or created as its own instance. This system allows the detection of different categories of tweets based on a small sample of labeled tweets. The process has three steps, data collection, annotation, and classification. Tweets are collected based on a pre-selected set of keywords. A small portion of theses tweets is then labeled by volunteers as in-category or out-category. In each disaster, different categories can be considered such as status update, shelter, or food. Labeled tweets which can be as few as 200, will be used as the training set of a classifier which labels remaining set of tweets which were collected based on the keywords. In the training process, n-grams of tweets are used

as features and hence the classifier needs to be retrained for every new category and disaster. The framework of this system is shown in Figure 3.4.

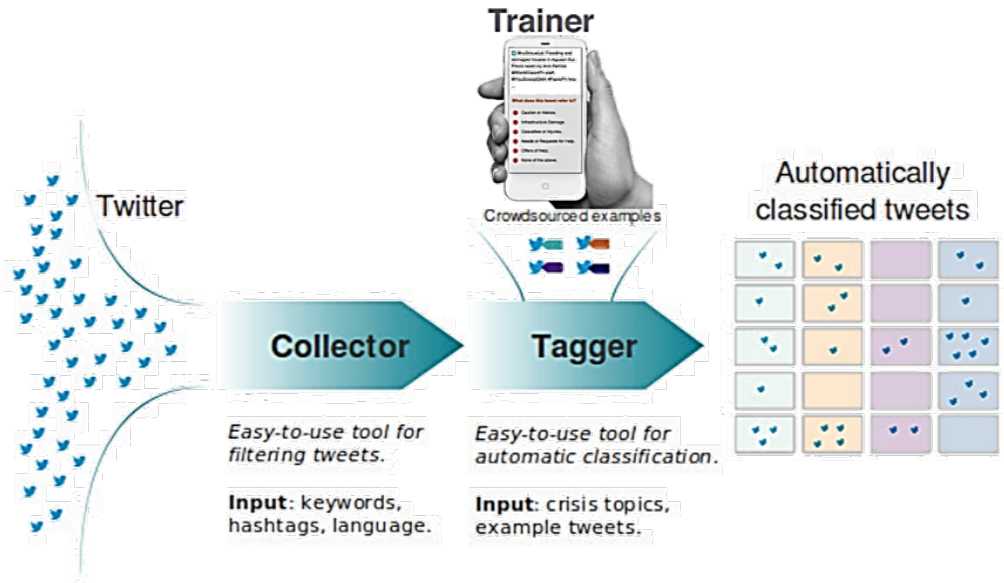


Figure 3.4: The Framework of AIDR (Imran *et al.*, 2014)

TweetTracker

TweetTracker (Kumar *et al.*, 2011) has been used by FEMA, The Red Cross, and others during crisis scenarios. In TweetTracker, tweets which are related to a job are crawled from Twitter Streaming API as they are published. Each job in TweetTracker is defined to crawl the data related to an event from Twitter, Youtube, Vcontakte, and/or Instagram. Each job is a collection of keywords, users (authors), and locations. After creating a job, if a post is published on any of the selected social media sites and matches any of the job's criteria, it will be selected. A TweetTracker job for 2016 Japan earthquake in Kumamoto is shown in Fig. 3.5. When defining a new job for the system, user can flag the job as being related to a natural or man-made crisis such as an earthquake or a bombing. Tweets of the flagged jobs will be labeled with the probability of being a request using the approach that is discussed in Section 3

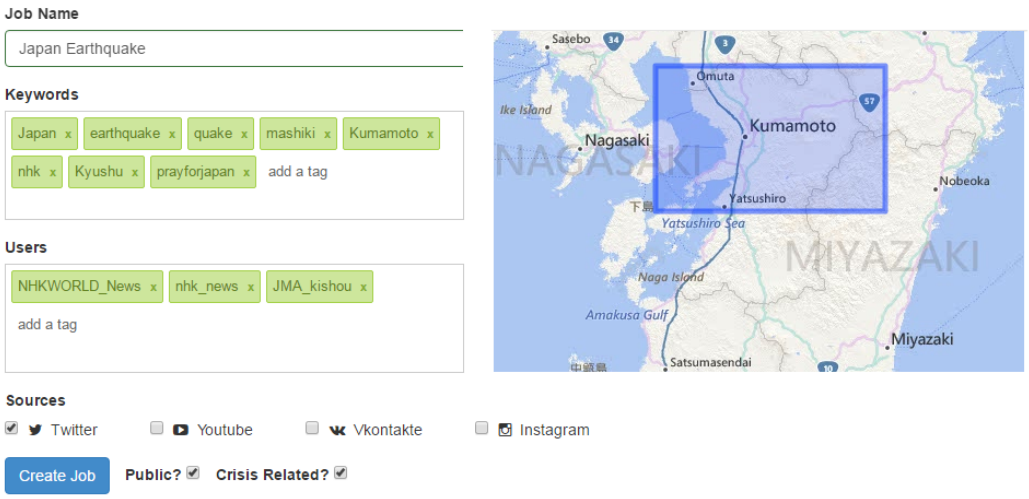


Figure 3.5: Job Creation in TweetTracker.

and the most probable ones will be show to the user. More details on TweetTracker is available in Appendix A.

3.1.2 Relief Tasks Facilitated by Machine Learning

In this section, we present some of the disaster relief tasks that are facilitated by Machine Learning techniques on the social media data.

Location Estimation

To overcome the challenge of location sparsity in social media data, several methods have been proposed to estimate the location of posts or users. Content of posts, activity characteristics, profiles, and networks of users are exploited to estimate the location in which a user is based or the post is originated from. The granularity of estimation differs from one method to another. Some approaches estimate the coordinates, some remain in the city-level, and some only focus on a disaster areas that can be limited to a neighborhood or expand to several cities or states.

Content is frequently used to estimate the origin of posts. N-grams and "crisis-

sensitive" features such as "in" prepositional phrase (such as "in Boston"), existential "there" (which usually describes an abstraction), and part-of-speech tag sequences are signals that discriminate in-region posts from out-region ones in course of a disaster (Morstatter *et al.*, 2014). Moreover, posts from a disaster area are less likely to include multiple hashtags, action words, and reference entities. Majority of such posts are original and contain URLs (Kumar *et al.*, 2014a). Posts from the same location frequently use similar words (Cheng *et al.*, 2010) and rarely use words that are used in other locations (Han *et al.*, 2013).

For locating users, the most intuitive features are geolocation or location field in their profile, the location of the websites that they linked to (which can be obtained using the IP or country code), time zone, and UTC24-Offset. These features can be combined using the stacking method (Wolpert, 1992) by considering an importance weight for each feature to find the most probable location of the user (Schulz *et al.*, 2013).

When location-indicating features are not available for a user, their location can be estimated using the location of users surrounding them. Backstrom et al. (Backstrom *et al.*, 2010) observe that there is a power law relation between physical distance and the probability of existing a social link. Based on this finding, they propose a maximum likelihood prediction method that indicates the most probable location for a user given its neighbors. Based on triadic closure, if user a is connected to users b and c, b and c are more likely to be connected to each other (Kossinets and Watts, 2006). In "Triadic heuristic" (Jurgens, 2013), the location of users is estimated as the geometric median of their neighbors who are in triadic closure with them. Moreover, users are more likely to follow users nearby and more often mention the location in which they live (Li *et al.*, 2012).

Palen and Anderson (Palen and Anderson, 2016) introduce the concept of "con-

textual streams" to combine Lexicon and location in order to overcome the issue of incompleteness. They use a set of broad terms (such as "frankenstorm" and "sandy" in the case of hurricane Sandy, 2012) to collect the first set of tweets. Then, they find the users who have geolocated tweets from their desired location. Finally, they collect their most recent 3,200 tweets and extend their previous dataset. Using this information, they can compare the activities of users who are located on the site of the disaster before, during, and after its occurrence.

Event Prediction

Social media has been used for predicting events that will happen in near future. Forecasting the popularity of products, movie box-office, election results, and trends in stock markets are examples of such predictions (Yu and Kak, 2012).

Prediction is based on features of social media posts. Increase in the number of posts which are related to a topic (Asur and Huberman, 2010) can be indicator of its future popularity. Changes in the patterns of using specific words in a area shows onset of an event (Sampson *et al.*, 2015). Also, sentiment of posts can show future status of a product (Mishne *et al.*, 2006). Crime prediction is also possible by semantic role labeling which is used for both finding the events and entities involved in them (Wang *et al.*, 2012).

Prediction method based on the extracted features can vary based on the problem. Regression method have been used for prediction popularity of posts (Szabo and Huberman, 2010) but do not perform well when sentiment data is being used (Zhang and Skiena, 2009). For predicting election results, Tumasjan et al. (Tumasjan *et al.*, 2010) use number of tweets mentioning political parties and their sentiments as indicators of popularity and political views toward them (Tumasjan *et al.*, 2010).

There is no prediction method with perfect accuracy. However, early detection

of natural disasters reduces hazards in nearby locations. For example, quakes in areas with geographic proximity are used to predict earthquakes seconds before they happen (Faulkner *et al.*, 2011).

Warning Systems

"Warning systems detect impending disaster, give that information to people at risk, and enable those in danger to make decisions and take action" (Sorensen, 2000). There has been a significant improvement in forecasting and warning systems especially for hurricane and earthquakes. Meteorologists can now forecast a hurricane 2 to 6 days before it hits an area and Global Seismic Network constantly monitors activity bellow Earth's surface. However, lack of complete data on natural hazards, monitoring instruments, and high dynamic nature of them keep accurate forecasting and warning a challenge (Reese, 2016) and "a 100% reliable warning system does not exist for any hazard (Sorensen, 2000)".

Social media facilitates is also used to deliver official and non-official warnings. Emergency managers and governmental organization post their warning messages via social media to be broadly accessed by the public (Houston *et al.*, 2015). Citizens also report warnings and advice about possible hazards (Imran *et al.*, 2013).

One source of information that can be used to improve accuracy of warnings is data of built-in accelerometers in cell phones. This data can be used for quick detection of earthquakes and estimating their intensity and effect. The measurements by these sensors which are transmitted before the loss of communication are used for estimating the degree of damage to different areas; the task that can take up to an hour when performed by helicopters (Faulkner *et al.*, 2014, 2011).

Social media is another source of information for warning systems. USGS uses tweets to check the accuracy of sensor reports and faster detection of earthquakes.

Disasters such as Sichuan earthquake in 2008 show that Twitter is faster at reporting earthquakes than USGS. Earthquakes can be detected using tweets by 60 seconds earlier than sensors which is a valuable time for warning areas under danger and start evacuation (Ellis, 2015). In another effort, Sakaki et al. (Sakaki *et al.*, 2010) consider each Twitter user as a sensor. The tweets by these sensors will be used to detect the occurrence of disasters and estimate their location.

Event Detection Methods

Events are real-world occurrences that unfold over space and time and. The goal of event detection methods is extracting events in a stream of news or social media posts (Allan *et al.*, 1998). Event detection using social media has been extensively studied and the different categorizations are available for proposed methods in this area.

When there is no information about future events available, the event detection method falls into the unspecified category. In this category, detection methods are based on bursts or trends in the stream of posts (Popescu and Pennacchiotti, 2010). In the specified event detection methods, contextual information such as time and venue are available for the anticipated event (Becker *et al.*, 2011a).

Another categorization of events is new versus retrospective. New event detection is extracting previously unseen events from a stream of posts as they come. Retrospective event detection also finds unseen events but the data source is an accumulation of historic posts (Allan *et al.*, 1998). Clustering methods are the most common in detecting both types of events (Atefeh and Khreich, 2015). But there are also supervised methods such as Naive Bayes (Becker *et al.*, 2011b) and gradient boosted decision trees that have been used for new event detection (Popescu *et al.*, 2011).

Clustering methods focus on documents and grouping them based on similarities, i.e. they are document-pivot. There is a group of feature-pivot techniques that use changes and bursts in features of documents to detect events. These features include frequency of specific keywords (Power *et al.*, 2014), surprise level of relevant keywords (Sampson *et al.*, 2015), and statistical features of posts (i.e. word frequencies) (Sakaki *et al.*, 2010).

3.2 Data

The classifier has been trained on a dataset of 13,260 tweets related to Hurricane Sandy, 2012, with 3,261 requests and 9,999 normal tweets. The requests have been provided by Purohit et al. (Purohit *et al.*, 2013) and normal tweets are a random sample of our Hurricane Sandy dataset which was collected during the same period of time.

3.2.1 Preprocessing

Several preprocessing steps have been followed to prepare the dataset for the learning step. Duplicate tweets and retweets have been removed because they will affect frequency of n-grams. Further, we have also eliminated similar tweets; the tweets which are different in at most one word (not considering the URL). As the final step, punctuation and stopwords have been removed that results in considering removing "#" from hashtags and "@" from mentions as simple words.

Before applying our model to the tweets in disaster related jobs, some preprocessing steps will be taken. Punctuations and stop words will be removed form the tweet's text and then it will be tokenized. N-grams and LDA topics will be found based on these tokens. Contextual features will be directly extracted from the meta data provided by Twitter for each tweet.

3.2.2 Feature Extraction

After disasters, based on the possible outcomes, specific keywords will be used by the affected people to report their problems and request for help. These keywords are important features to distinguish requests from other tweets. We transform keywords to n-grams in our model by considering unigrams, bigrams, and trigrams. This process results in thousands of features depending on the number of unique words in the corpus of tweets. Another content-based feature is the latent topics in tweets which can be extracted using an LDA model (Blei *et al.*, 2003). By using the LDA topics, we are reducing the dimensionality of the content-based features from several thousand n-grams to just twenty features. This process is akin to noise removal and potentially improves the results.

The second group of features are contextual. *Source* is the software by which a tweet has been published such as "web" or "iPhone". Most popular sources are the Twitter website, Twitter web clients, and Twitter applications for iPhone and Android. *Location* of a tweet is a valuable information. Users can enable the geolocation service while tweeting to help other users and first responders verify that the author is actually located in the affected area (Morstatter *et al.*, 2014). Users can *mention* each other using "@" preceding a username. This feature is used to establish conversations on Twitter. Hence, the number of mentions in a tweet can distinguish requests from conversations. In order to convey as much information as possible in the limited length of a tweet, users requesting help will use less *hashtags*. Twitter users include links (URLs) to provide further information by directing other users to external websites. Requests after a crisis will use *URLs* to introduce websites for humanitarian purposes such as collecting monetary donations (Figure 3.6(a)) or collecting blood donations (Figure 3.6(b)).

Figure 3.6: Two Request-For-Help Tweets.

3.3 Method

The goal is calculating the probability of a tweet being a request-for-help using a supervised model, i.e. a classifier. Request are a subset of actionable tweets that are "relevant to situational awareness; they contain information that provides tactical, actionable information that can aid people in making decisions, advise others on how to obtain specific information from various sources, or offer immediate postimpact help to those affected by the mass emergency"(Vieweg, 2012). The classifier uses content and context of tweets as input. Content features are directly extracted from the raw text of tweets and include n-grams (words and phrases) and topics (which are generated using an LDA model (Blei *et al.*, 2003)). Contextual features are the metadata of tweets such as entities (e.g., URLs, hashtags, and mentions), timestamp, retweet count, and author's information (e.g., username, location, number of followees, number of followers, and profile information). Content has been used in previous studies (Purohit *et al.*, 2013), however, the context has been overlooked. In the rest of this section, we will provide the details of our process to create our proposed model. The features that we use in this model are shown in Figure 3.7.

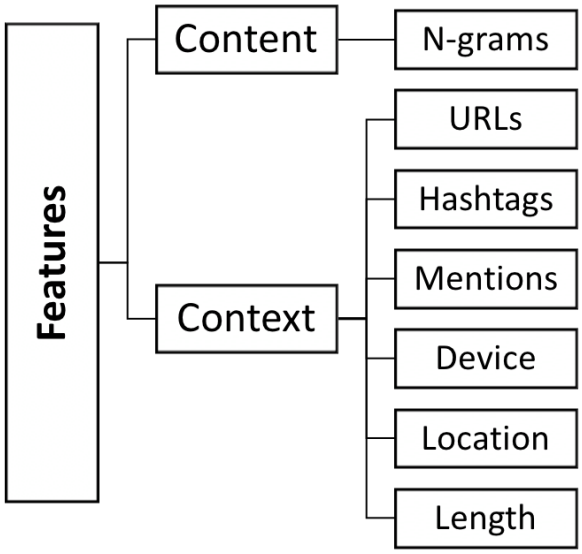


Figure 3.7: Two Types of Features Used in Detecting Help-Seeking Tweets After Major Disasters.

3.4 Experiments

To evaluate the performance of our model, we train a supervised classifier on users who tweeted about Hurricane Sandy, 2012. We chose Decision Tree classifier because it is interpretable and has the ability to show the importance of each feature in the final results. The model will label each instance in the dataset as help-seeking or other. We evaluated the model performance using precision, recall, and F_1 score. As reported in Table 3.1, using all the features results in the best performance, as expected. However, the change is minor when adding bigrams to the feature set, suggesting that unigrams contain the most information.

Table 3.1: Performance of the Proposed Model for Detecting Help-Seeking Tweets.

Feature Set	Precision	Recall	F_1 Score
Context	0.710	0.724	0.717
Context and Unigrams	0.896	0.908	0.902
Context, Unigrams, and Bigrams	0.906	0.916	0.911

3.5 Implementation in TweetTracker

Our systems has been implemented as part of TweetTracker which is a powerful tool for tracking, analyzing, and understanding activities on Twitter (Kumar *et al.*, 2011). We periodically apply our model to new tweets for crisis related jobs and update the most probable requests. An example output of our system for a job on Hurricane Sandy, 2012, is presented in Fig. 3.8. This figure shows a snapshot of TweetTracker in which there are several tabls to provide information of the selected job such as Tweets, Users, and Topics. Our system is shown in the rightmost tab, Requests. This tab is active when the selected job has been marked as crisis related. In Requests tab, text, author, and probability of the most probable requests are provided to users. Along with the probability of a tweet being a request, our system has the ability to show the geotagged most probable requests on the map. This feature is shown in Fig. 3.9.

3.6 Summary

Social media is widely used in disasters. Millions of tweets are published both during and in the aftermath of crises. Some of these tweets reflect observations, while many others contain offers, and requests for help. On the other hand, a major issue that first responders face is locating victims and distributing resources in this

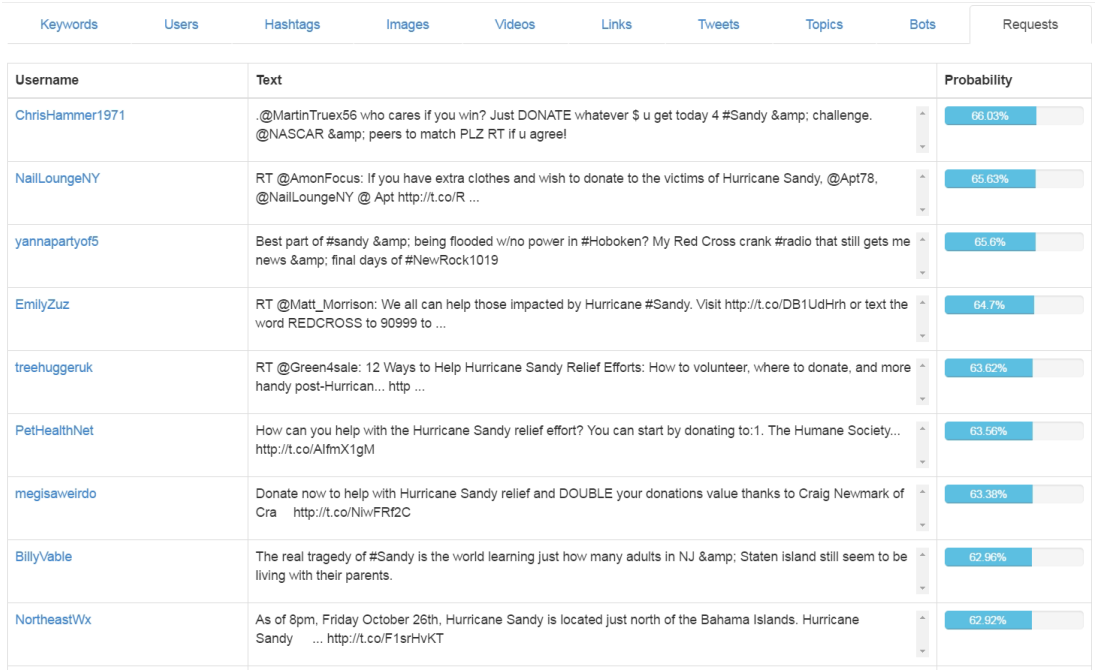


Figure 3.8: The Output of Our System, Most Probable Help-Seeking Tweets.

uncertain environment. To help with these challenges, our system has the capability to distinguish requests and show their information and location. The underlying method of the proposed systems, along with content, benefit the context which has been used in previous studies but not regarding the problem of finding social media requests in aftermath of disasters. In this paper we have both proposed and implemented a method that can help first responders connect with people who are contributing and requesting aid, for the purpose of streamlining the process of providing aid to crisis-afflicted areas.

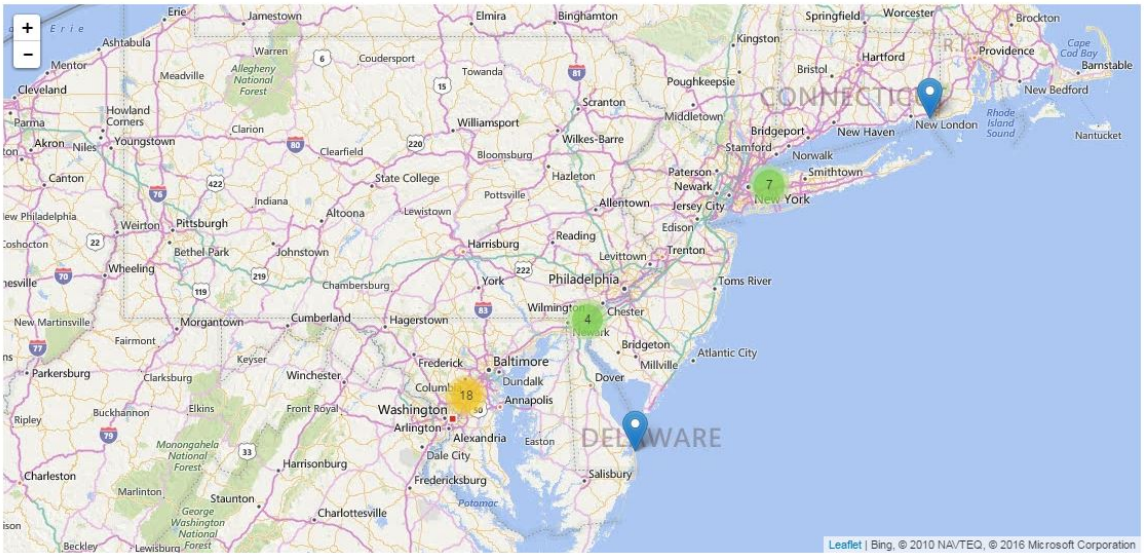


Figure 3.9: Output of Our System, Location of Most Probable Geotagged Requests on the Map of Disaster-Hit Area.

Chapter 4

DETECTING BOTS ON SOCIAL MEDIA

Bots are prevalent on social media and their malicious actions have been observed repeatedly (see Section 2.1 for more details). Thus, researchers have put great effort into understanding bots and developing methods to detect them. In supervised bot detection methods, which are the focus of this work, a labeled dataset of bots and human users is available prior to training a machine learning classifier. Using these labels, we can learn characteristics that discriminate bots from humans and use them to build classifiers that predict class labels (bot or human). The classifiers are then tested on unobserved datasets and evaluated using metrics such as precision, recall, and F_1 score.

A common theme among previous bot detection methods is attempting to maximize precision (Lee *et al.*, 2011; Chu *et al.*, 2012; Varol *et al.*, 2017). This is one extreme: the sole purpose is to minimize false positives and avoid mistakenly marking a human user as bot. By doing this, detection methods avoid removing human users from the site but leave many bots undetected. The other extreme is eliminating bots from social media at the price of removing human users. This approach is not preferable either. A method for finding a trade-off between precision and recall is optimizing for F_1 score which is the harmonic mean between precision and recall. Harmonic mean is dominated by the minimum of its arguments. Hence, F_1 cannot become arbitrarily large when either precision or recall is unchanged and the other metric is increased. This prevents bot detection algorithms from landing on trivial solutions (marking all users as bots or humans) to gain high F_1. However, considering the same weight for precision and recall in F_1 prevents us from having control over the

final values of either precision or recall. In other words, two classifiers are considered equally good if they have the same F_1 regardless of the fact that one might result in higher recall and the other one a higher precision. The ideal case is finding a solution close to optimum F_1 that allows us to focus on precision or recall depending on the application.

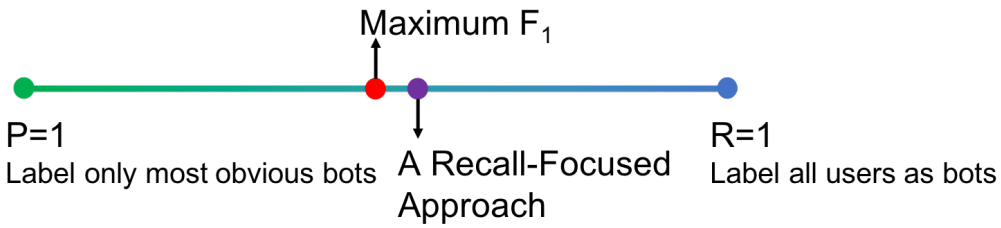


Figure 4.1: Our Goal Is Having a Recall-Focused Approach Close to the Optimal F_1.

To align with corporate goals (having a large number of active users and retaining human users by avoiding accidentally suspending their accounts), bot detection models with high precision are preferable. However, from a user's perspective, both social media users and researchers alike, the preferable situation is encountering a minimum number of bots. So, in this case, high recall is preferred. In this work, we focus on developing supervised algorithms aligned with a user's perspective: BoostOR and REFOCUS. We use multiple real-world datasets to show how we can find a sweet spot between blindly optimizing for F_1 or recall as shown in Fig. 4.1. We also compare our proposed methods with state-of-the-art bot detection models to show that focusing on recall does not necessarily result in overall performance deterioration in terms of F_1.

4.1 Data

In order to build a classifier, we need to obtain a gold standard dataset: a set of users and their bot/human label to evaluate the results. Of course, this information

does not come affixed to each user when the researcher collects the dataset. Thus, the onus is on the researcher to collect these labels. There are three main approaches to this:

- *Manual annotation:* As in most other classification problems, manual labeling can be to obtain ground truth. Although this method has been used widely in this field (Xie *et al.*, 2008; Chu *et al.*, 2010; Grier *et al.*, 2010; Ratkiewicz *et al.*, 2011a; Cook *et al.*, 2014), we still have the challenge of how to choose annotators and how many annotators are sufficient in order to achieve reliable labels. Furthermore, this method does not scale as it takes time to recruit users and funds to pay them.

- *Suspended users lists:* This method leverages the social networking site itself to obtain the labels. The researcher simply observes the users and sees which ones get suspended or removed by the site. This approach is simple, but the concern remains that the reason of suspension and deletion is not usually indicated in such lists. Thus, many suspension lists include users who violated other rules and regulations of the site. Methods proposed in (John *et al.*, 2009; Thomas *et al.*, 2012; Lee and Kim, 2014) use these lists.

- *Honeypots:* A newer ground truth acquisition method is based on using honeypots, bots created by the researcher to lure other bots. Honeypots show non-human behaviors and can be made in groups to increase their effect by connecting and interacting with each other without intervention in activities of normal users. Due to the fact that they are designed in a way that any normal user can immediately tell they are bots, any user in the network that connects to a honeypot will be considered as a bot. By using this method, a researcher can gather a large set of bots active in a network with high confidence. This

method has been applied before (Lee *et al.*, 2011; Thonnard and Dacier, 2011; Morstatter *et al.*, 2016). We provide a detailed description of characteristics of honeypots for different scenarios and applications in Appendix B.

4.1.1 Description of Datasets

To show the robustness of our models with respect to the language, topic, time, and labeling mechanism, we use the datasets represented in Table 4.1.

Table 4.1: Statistics of the Datasets Used in This Study.

Property	**Arabic Honeypot**	**Social Spambot 1**	**Social Spambot 2**
Tweets	637,435	4,449,395	4,257,918
Retweets	209,703	782,267	754,104
Human Accounts	2,317	1,083	1,083
Bot Accounts	1,978	991	464
Bot Ratio	46.05%	47.78%	29.99%
Labeling Approach	Honeypot	Manual	Manual

- Libya Dataset: from February 3^{rd}, 2011 to February 21^{st}, 2013, we collected Twitter data pertaining to Arab Spring activity in Libya. The data was collected from Twitter's Streaming API[1], a service which provides a stream of tweets matching a query (Kumar *et al.*, 2014b). The query we used to collect this data consisted of the following keywords: `#libya, #gaddafi, #benghazi, #brega, #misrata, #nalut, #nafusa, #rhaibat`, as well as a geographic bounding box around Libya[2]. Statistics on the dataset are shown in Table 4.1.

  We obtained labels of whether a user is a bot or human by observing how Twitter handled these users. In February of 2015, we crawled each user in the

[1] `https://dev.twitter.com/streaming/reference/post/statuses/filter`

[2] Southwest Lng/Lat: 23.4/10.0; Northeast Lng/Lat: 33.0,25.0.

dataset and observed the status of his Twitter account. We observed the user's account status via the `statuses/user_timeline` API endpoint[3]. The status can take on one of three values:

1. **Active:** A user whose account is still open and available on the site.
2. **Deleted:** A user whose account has been deleted. A user can be deleted by violating Twitter's policies. This is considered a permanent ban.
3. **Suspended:** A user whose account has been suspended for violating Twitter's policies. This is considered a temporary ban, where the user can petition Twitter to have his account reinstated.

These labels were obtained by using Twitter's APIs to crawl the dataset and inspecting the response code from the API. Through this process we discovered that **92.5%** of the users are active, **4.7%** are deleted, and **2.8%** are suspended. Because deleted and suspended have similar meanings, we consider both labels as a bot.

At first glance, we notice that the fraction of users identified as bots by this labeling technique is around 7.5%. This distribution gives a very conservative estimate of the number of bots on Twitter. This is a side effect of an industry approach which focuses purely on precision in order to avoid accidentally deleting some real users. We realize that this is not representative of the true distribution of bots on the site (Wei *et al.*, 2015), and that many bots may have been overlooked by Twitter. With this in mind, we introduce our next dataset which focuses on detecting bots in the wild through honeypots which tweet specific content. In response to this, we collect another dataset by focusing on

[3]https://dev.twitter.com/rest/reference/get/statuses/user_timeline

detecting bots in the wild using honeypots which tweet specific content, Arabic Honeypot Dataset.

- Arabic Honeypot Dataset (Morstatter *et al.*, 2016): this dataset consists of bots tweeting messages in Arabic. To collect this dataset, we construct a honeypot network. This network consists of 9 accounts controlled automatically by a single controller. Each account tweets messages containing Arabic phrases identified by a subject matter expert pertaining to a specific group of people. Each account also randomly follows other honeypots in our network. Since bots have a lower chance of forming social ties (Thomas *et al.*, 2011), we perform this random following process to lower the chance that our accounts are deleted by Twitter's automatic account removal algorithm. Additionally, each honeypot can randomly retweet one of the other honeypots it follows in order to give that honeypot prominence on the network and lower its probability of being deleted due to Twitter's policies.

  While the honeypot method yields a set of bot accounts, it does not give us a set of real users. This is because we only look at the bot followers to our honeypots, and we do not have ground truth labels for real users. In order to test our model, we need to collect a set of real users to use as negative training instances. To do this, we manually inspected a seed set of 10 users who also tweeted the same phrases[4]. We did this to ensure that the algorithm we train finds bot patterns in the text, and does not simply learn the difference in language distributions. Moving forward with the assumption that real users do not follow malicious bots, we use 1-link snowball sampling to collect their

[4]We did not extract verified accounts as these users are not normal users. They are often controlled by public relations firms and tweet on specific topics.

immediate network. We ensured that each user in the sample had fewer than 1,000 followers to make sure that we did not collect any celebrities (Zafarani *et al.*, 2014). We also inspected each user in the sample to ensure that he tweeted one of the phrases at some point in his last 200 tweets. This approach yielded 3,107 real-world accounts, which helped us to maintain the same class distribution reported in other approaches (Lee *et al.*, 2011). More details on the honeypot approach and the mechanism we used is provided in Appendix B.

- Social Spambot Datasets (Cresci *et al.*, 2017): Cresci et al. in their previous work on detecting social bots on Twitter namely: `t`est set #1 and `t`est set #2. We call these datasets Social Spambots 1 and 2, respectively, in our work. Each dataset is a combination of social spambots and human users on Twitter. Since the Arabic Honeypot Dataset was focused on detecting bots that may evade existing Twitter spam filters, this dataset was also chosen due to its self-proclaimed detection of evolving Twitter social bots. The subsets we chose from Cresci et al. are their two testing datasets which are a combination of the datasets labeled "Social Spambots #1","Social Spambots #3", and "Genuine Accounts" in their work (Cresci *et al.*, 2017). Specifically, in their work, "test set #1" was comprised of the entire set of "Social Spambots #1" plus an equal number of "Genuine Accounts" (Cresci *et al.*, 2017). Similarly, "test set #2" was made up of the entire set of "Social Spambots #3" and an equal number of "Genuine Accounts" from that paper (Cresci *et al.*, 2017).

  To collect the human user accounts, Cresci et al. contacted random users, asked a natural language question, and manually evaluated if the user was a human. Social Spambots 1 contains these genuine accounts plus social bots that were discovered during the 2014 Mayoral election in Rome, Italy which were used

to retweet a candidate within minutes of his original posting. Social Spambots 2 includes the genuine accounts and social bots that advertised products on *Amazon.com* by deceitfully spamming URLs which point to the products. We obtained these datasets directly from the BotOrNot Bot Repository (Davis *et al.*, 2016).

4.1.2 Feature Extraction

Bot accounts are created by malicious actors to serve specific purposes. Thus, their content can be a strong indicator to expose such potentially automated accounts. The problem with using content for bot detection is that the raw text features are of high dimensionality and sparse. Inspired by the recent advances of topic modeling, we adopt Latent Dirichlet Allocation (LDA) (Blei *et al.*, 2003) to obtain a topic representation of each user. LDA is an unsupervised and probabilistic model that has been proven useful for extracting latent semantics of documents. The principle idea behind LDA is that it treats each document as a distribution over topics, and each topic as a distribution over the vocabulary in the dataset. LDA requires one parameter[5], K, the number of topics in the corpus. From here, LDA learns two matrices:

1. Φ: the $Topic \times Word$ matrix. Each topic in LDA, Φ_i, is a probability distribution over the entire vocabulary in the corpus. Thus, Φ_i^j is the probability of word j occurring in topic i.

2. Θ: the $Document \times Topic$ matrix. Since each document is modeled as a distribution over topics, this each row, Θ_i, contains the document's distribution over all of the topics learned by LDA.

[5]We use the default hyperparameter value of $\alpha = \frac{1}{K}$, and $\beta = \frac{1}{K}$.

In this approach, we treat the user as a document. Each user's document consists of the concatenation of all of the content of his tweets. We feed these documents into LDA with K topics, obtaining Φ and Θ values for the corpus[6]. Since the Θ matrix contains the affinity for each user to each topic, we can treat these as features to be fed into a classifier. We build an SVM classifier by directly using Θ as the $Instance \times Feature$ matrix, where the instances are users and the features are their affinities for the latent dimensions discovered by LDA. We follow the assumption that, since bots are naturally more interested in certain topics, denoting each user as a distribution over different topics may help to better identify them from regular accounts (Morstatter *et al.*, 2016).

4.2 Method

In this section, we formally define what precision (P), recall (R), and F_β score. Then introduce two recall-focused methods that we proposed, BoostOR and REFOCUS.

4.2.1 Evaluation Metrics

classifiers can be evaluated using precision (P) and recall (R) defined in equation 4.1. True Postive (tp), True Negative (tn), False Positive (fp), and False Negative (fn) are shown in Fig. 4.2.

$$P = \frac{tp}{tp + fp}, \quad R = \frac{tp}{tp + fn} \tag{4.1}$$

It is common to see a fall in precision when the classifier has high recall and vise versa. To avoid this pitfall, one might use F_β score which is the *weighted* harmonic

[6]We will discuss our selection of K in the experiments section.

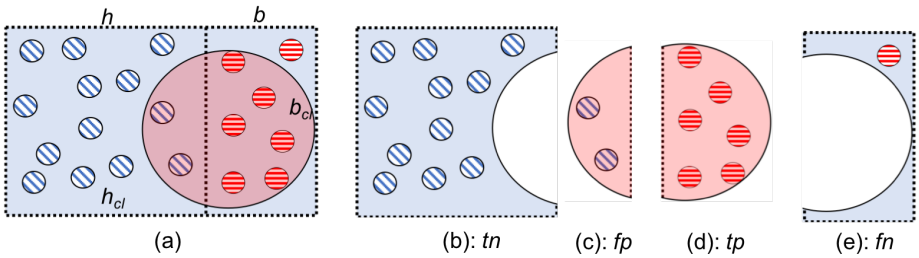


Figure 4.2: Illustration of True Negative - (b): tn, False Positive - (c): fp, True Positive - (d): tp, and False Negative - (e): fn for a Classifier Trained on Dataset (a) When the Classifier Labels a Subset of Users as Bots (Positive Class) - b_{cl} - and the Rest as Humans (Negative Class) - h_{cl}.

mean of precision and recall and is defined in equation 4.2.

$$F_\beta = \frac{(1+\beta^2)PR}{\beta P + R} \tag{4.2}$$

Mostly commonly used F_β score to evaluate the overall performance of a classifier because it is maximized when we have a trade-off between precision and recall. However, having two classifiers with the same F_1, one might have a higher recall and the other have a higher precision. Because of our focus on recall, we will use F_β scores with $\beta > 1$. With β values greater than one, we put more emphasis on recall and penalizing the classifier more when it is losing on recall. We explain the details of using this metric in designing a recall-focused approach in Section 4.2.3.

4.2.2 Method I: BoostOR

Since bots are usually generated by different parties for different purposes, the discriminant characteristics of bots from different groups are unrelated. Such heterogeneity of bots makes it challenging to come up with a classifier. To this end, we formulate the problem as a boosting task. Boosting methods aim to achieve an optimal classifier through ensembling weak classifiers, which are more proper for this problem since different weak classifiers will focus on different bots.

First, we investigate whether boosting algorithms are directly applicable for bot

detection. For generality, we selected AdaBoost (Freund and Schapire, 1997) for optimization. AdaBoost trains one weak classifier based on all training examples every iteration. In each iteration, the misclassified examples in the previous round are higher weighted in the next round. The final classifier is the weighted ensemble of all weak classifiers. Therefore, the key issues are how the weight are determined for different weak classifiers and training examples. Here we use α to denote classifier weight, the weight of classifier t can be calculated as follows:

$$\alpha_t = -\frac{1}{2} ln(\beta_t), \tag{4.3}$$

where β_t denotes the extent of the base learner deviates from the optimal solution. β_t can be directly calculated with training error ϵ as follows:

$$\beta_t = \frac{\epsilon_t}{1 - \epsilon_t}. \tag{4.4}$$

As shown in Eq. 4.5, α_t is greater than zero if ϵ_t is less than 50%; while α_t is less than zero if ϵ_t is greater than 50%. ϵ_t can be calculated through Eq. 4.5.

$$\epsilon_t = \frac{1}{\sum_{i=1}^{m} \mathbf{D}_t(i)} \sum_{i=1}^{m} \mathbf{D}_t(i) \mathbf{1}(h_i(\mathbf{v}_i) \neq y_i), \tag{4.5}$$

where the function $\mathbf{1}(\cdot)$ equals one when the condition holds, and zero otherwise.

The second issue is to determine the weight of each training instance at each iteration. The instance weight is regulated depending on whether it is correctly classified in the previous round and the performance of the weak learner. As denoted in Eq. 4.6, when the weak learner's error rate is less than 50%, weights of misclassified users will increase while those of the rest decrease.

$$\begin{aligned} \mathbf{D}_{t+1}(i) &= \frac{\mathbf{D}_t(i) exp(-\alpha_t y_i h_t(\mathbf{v}_i))}{Z_t} \\ Z_t &= \sum_{i=1}^{m} \mathbf{D}_t(i) exp(-\alpha_t y_i h_t(\mathbf{v}_i)). \end{aligned} \tag{4.6}$$

The algorithm of AdaBoost for bot detection is illustrated in Algorithm 1. Note that m is the number of all users and k is the number of features. Since LDA is adopted to represent the user, here each user vector $\mathbf{v}_i$ can be viewed as a probability distribution over k topics. The user labels are denoted by $\mathbf{Y} = \{y_1, \ldots, y_m\} \in \{-1, 1\}^{m\times 1}$, where $y_i = 1$ means that user i is a bot.

Input: The user-feature matrix $\mathbf{V} \in \mathbb{R}^{m\times k}$,

the user-label matrix $\mathbf{Y} \in \{-1, 1\}^{m\times 1}$ and a weak learner $h : \mathbf{V} \to \mathbf{Y}$

the initial user weight vector $\mathbf{D}_1 = (\mathbf{D}_1(1), \ldots, \mathbf{D}_1(m))$

the maximum number of iterations max_{iter}.

Output: The ensemble classifier $H(\mathbf{v}) = sign(\sum_{t=1}^{T} \alpha_t h_t(\mathbf{v}))$

For t = 1, ..., max_{iter}

Train weak learner using instance weight $\mathbf{D}_t$;

Get the trained classifier $h_t : \mathbf{V} \to \mathbf{Y}$ and training error ϵ_t as Eq. 4.5;

Calculate the weight $\alpha_t \in \mathbb{R}$ for h_t as Eq. 4.3;

Update user weight as Eq. 4.6;

until Convergence.

Algorithm 1: AdaBoost for Bot Detection

Since we aim to achieve a classifier which is sensitive to bots, we investigate how AdaBoost could be adapted for optimizing recall. As mentioned before, the weight of bots may be reduced if they are correctly classified. The reduction of weight leads subsequent weak classifiers to less focus on these bots, which is unfavourable. Therefore, we next investigate how the sensitivity to bots can be kept through regulating the weights of training instances. An intuitive solution is to avoid reducing weights of bots, which can be formulated as follows:

$$\mathbf{D}_{t+1}(i) = \mathbf{D}_t(i)\beta^{-y_i|h_i(\mathbf{v}_i - y_i)|}. \tag{4.7}$$

As shown in Eq. 4.7, if a regular user is predicted to be a bot, the corresponding weight will be multiplied by $\beta^{|h_i(\mathbf{v}_i - y_i)|} \in (0, 1]$, while the mislabeled bots still gain more weights. The loss between prediction and ground truth spans between the range of $[0, 1]$, and an exponential form ensemble predictor is adopted, which both enable Theorem 1 to hold. This means that the training error after T iterations is bounded. We name the new boosting algorithm as Boosting through Optimizing Recall (*BoostOR*). The detailed algorithm is shown in Algorithm 1. Figure 4.3 illustrates how the example weight is updated in AdaBoost model. The mislabeled instances will gain a larger weight in the second round of iteration. While in *BoostOR*, the weight change depends on the labels of the user. If a bot is wrongly predicted, its weight is enlarged. mislabeled regular users are more often ignored.

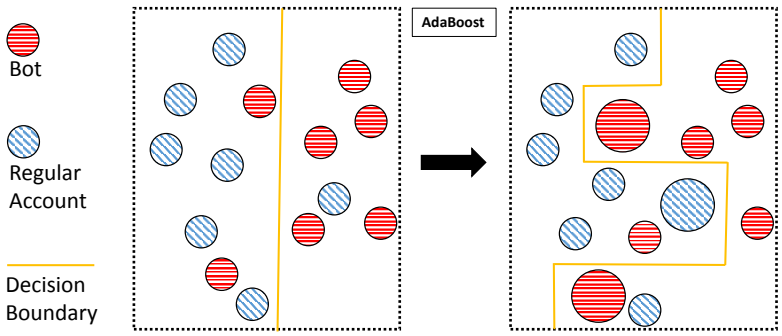


Figure 4.3: Illustration of Updating Instance Weights in AdaBoost.

However, two questions need to be answered about BoostOR: 1) will it converge by keeping weights of bots? 2) will trivial solutions be achieved by classifying all examples as bots? To answer the first question, we introduce Theorem 1 to indicate the convergence rate of the algorithm:

Theorem 1 *After T iterations, the average training loss of each iteration of BoostOR is upper bounded by:*

$$\begin{aligned} \min_{\mathbf{i}} \quad & \frac{L_i}{T} + \frac{\sqrt{2\hat{L}\ln(m)}}{T} + \frac{\ln(m)}{T} \\ \textit{where} \quad & L_i = \frac{1}{2}\sum_{j=1}^{T} |h_j(\mathbf{v}_i - y_i)| \end{aligned} \tag{4.8}$$

The corresponding proof can be found in (Freund and Schapire, 1997), where the range of convergence rate is also provided. The rate of convergence is no more than $O(\sqrt{\ln(m)/T})$ and can be as fast as $O(\ln(m)/T)$. $\hat{L}$ is the loss of optimal solution.

Since the weight of regular users are reduced instead of removed, trivial solutions will not be easily achieved in real world data, where bots are the minority group. We empirically prove this with real world Twitter datasets in Section 4.3.

Input: The user-feature matrix $\mathbf{V} \in \mathbb{R}^{m \times k}$,

the user-label matrix $\mathbf{Y} \in \{1, -1\}^{m \times 1}$ and a weak learner $h : \mathbf{v} \to y$

the initial user weight vector $\mathbf{D}_1 = (\mathbf{D}_1(1), \ldots, \mathbf{D}_1(m))$

the maximum number of iterations max_{iter}.

Output: The ensemble classifier $H(\mathbf{v}) = sign(\Pi_{i=1}^{max_{iter}}(\beta_i^{-h_i(\mathbf{v})} - \beta_i^{-\frac{1}{2}})$

For t = 1, ..., max_{iter}

Train weak learner using instance weight $\mathbf{D}_t$;

Get the trained classifier $h_t : \mathbf{V} \to \mathbf{Y}$ and training error ϵ_t as Eq. 4.5;

Calculate the weight $\alpha_t \in \mathbb{R}$ for h_t as Eq. 4.3;

Update user weight as Eq. 4.7;

until Convergence.

Algorithm 2: BoostOR for Bot Detection

4.2.3 Method II: REFOCUS

A bot detection classifier generates the probability of being a bot (belonging to the positive class) for each instance in the dataset. To assign a binary label to users, a classifier uses a threshold (commonly set to 0.5 (Pedregosa *et al.*, 2011)) to decide; if the probability of being a bot is more than the classification threshold then the user is labeled as a bot, otherwise as a human.

Precision and recall can be independently maximized easily. A trivial approach for increasing the recall is lowering the classification threshold and classifying more users as bots. Alternatively, increasing the classification threshold results in labeling most users as humans, with only the unquestionably obvious bot users labeled as bots, and causes a trivial increase in precision. However, precision and recall are not independent from each other: increasing one might result in decreasing the other. One approach for finding a trade-off between precision and recall is using the F_β score.

With β values greater than one, β times more weight is put on recall and for values less than one, β times more weight is associated with precision.

4.2.4 Searching for a Trade-off: Selecting β

Our goal is optimizing for recall, hence, we utilize F_β with $\beta > 1$ to find the best classification threshold: a sweet spot between where F_1 (overall performance) is maximized and where $R = 1$. The framework of our recall focused approach is presented in Fig. 4.4. We divide the dataset to 90% $Train$ and 10% $Test$. Then, for ten iterations, we divide Train to 90% $Train_i$ and 10% Val_i which are training and validation sets respectively; $Train_i$ is 81% of the whole data and Val_i is 9%. In each iteration, we train a classifier C_i on $Train_i$, change the classification threshold

between 0.1 and 0.9 with 0.1 steps, and find the threshold that results in the highest F_β score on Val_i; we call this threshold t_i. After the tenth iteration, we get an average of the thresholds t_1 to t_{10} to find the average threshold t. Then we train a classifier, C, on $Train$ and using t, we find the precision, recall, and F_1 score. We repeat this process ten times and report the average of precision, recall, and F_1 scores as our final results.

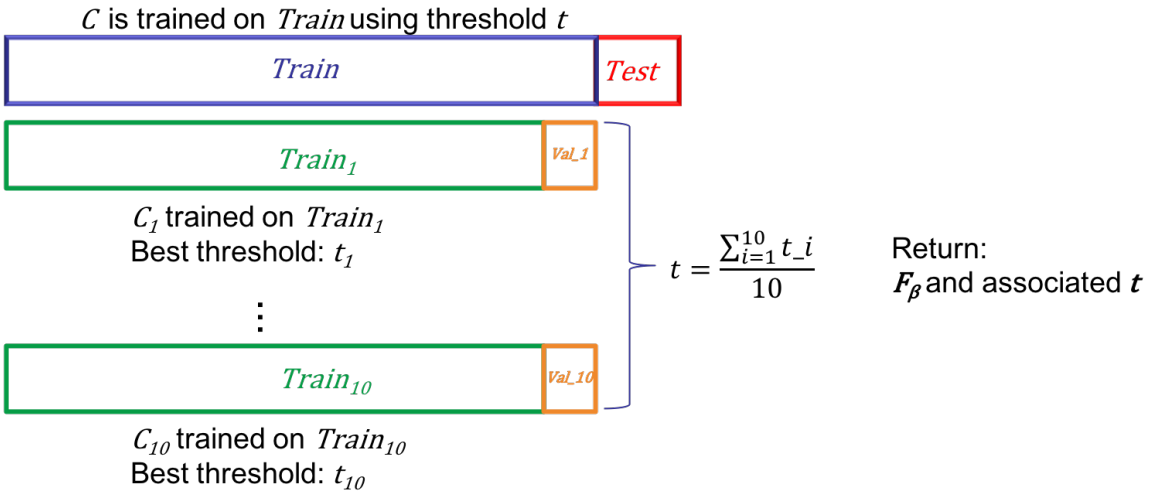


Figure 4.4: Framework for the Proposed Bot Detection Model, REFOCUS.

We need to test different values of β in the training phase to find the best classification threshold using F_β. As we increase β, precision has a non-increasing trend and recall has a non-decreasing trend. This happens because as we increase β we put more weight on recall in comparison to precision. More formally

$$R_{\beta_i} \geq R_{\beta_j} \quad and \quad P_{\beta_i} \leq P_{\beta_j} \qquad if \quad \beta_i > \beta_j \tag{4.9}$$

Due to this non-increasing pattern of precision with increase of β, we prefer to maintain a low β as long as we do not sacrifice the chance of achieving a higher recall with minor loss in precision. To find the right β, β_{opt}, we start from $\beta = 1$ and at step t

we set $\beta_{opt} = \beta_t$ if

$$(R^{\beta_t} - R^{\beta_{t-1}}) > (F_1^{\beta_{t-1}} - F_1^{\beta_t}) \tag{4.10}$$

Meaning that we choose a larger β if the gain in R is more than the loss in F_1.

4.3 Experiments

In this section we empirically investigate the performance of our proposed approaches. For each of the proposed methods, BoostOR and REFOCUS, we'll present their performance in different scenarios and in comparison with baseline bot detection models.

4.3.1 Evaluating BoostOR

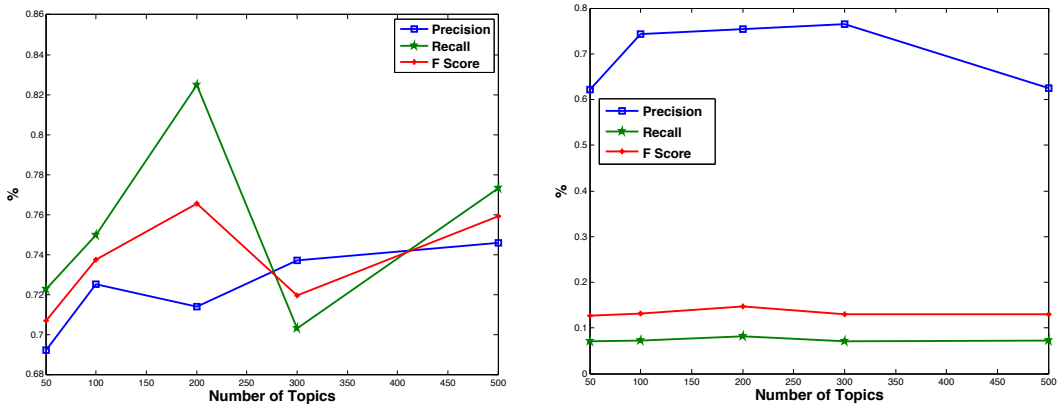
In this section we empirically investigate the performance of BoostOR with respect the ability to maximize the F_1 score on bot detection. We perform two experiments: first to show the performance of different algorithms, and second to show how the parameters used in building the AdaBoost and BoostOR models can influence the results of these methods.

Model Parameters

In this section, we study how the number of topics will influence the performance of the proposed model, BoostOR. The change of precision, recall and F_1 score as a function of K is illustrated in Figure 4.5. On both datasets, the dimensionality spans from 50 to 500 and the variations of performance are observed.

As shown in the Figure 4.5, the best performance is achieved when there are around 200 topics. When K deviates from the optimal value of 200, no matter increasing or decreasing, the performance decreases. Too many topics enable some

redundant and meaningless topics to exist, while some important topics may be neglected if the K is too small.



(a) Honeypot dataset (b) Libya dataset

Figure 4.5: Precision, Recall, and F_1 Score of BoostOR with Varying Number of Topics.

Testing the Overall Performance

We introduce a set of heuristics that are used to differentiate bot from non-bot users in social media. These heuristics are based upon state-of-the-art studies in bot detection on social media and will be used as baselines in evaluating the performance of the proposed model.

- Fraction of Retweets: this measures the number of times the user published a retweet divided by the number of tweets the user has published, calculated as:

$$Heuristic_{Retweet}(u) = \frac{|\{x|x \in tweets^u, x \text{ is retweet}\}|}{|tweets^u|}. \tag{4.11}$$

  Whether a tweet is a retweet is determined by looking at the "retweeted_status" field of the tweet's data when returned through the API. If the field contains tweet information, we consider it to be a retweet. This measure was introduced in (Ratkiewicz *et al.*, 2011b), and hypothesizes that bots are unable to produce

original content, so they rely on the retweet feature in Twitter in order to establish their presence.

- Average Tweet Length: a tweet's text is limited to 140 characters, but it is possible that bots post fewer than that as they could just be promoting a URL or a hashtag (Lee and Kim, 2014). To account for this, we introduce this heuristic which measures the average length of the user's tweets. It is the sum of the characters in all of the tweets the user has published divided by the number of tweets the user has published, formally:

$$Heuristic_{Length}(u) = \frac{\sum_{i}^{|tweets^u|} |tweets_i^u|}{|tweets^u|}, \tag{4.12}$$

  where $|tweets_i^u|$ is the length of user u's i-th tweet, measured by the number of characters in the tweet.

- Fraction of URLs: this measures the number of times the user published a tweet containing a URL divided by the number of tweets the user has published, formally:

$$Heuristic_{URL}(u) = \frac{|\{x|x \in tweets^u, x \text{ contains URL}\}|}{|tweets^u|}. \tag{4.13}$$

  Whether a tweet contains a URL is determined by looking at the "entities" field of the tweet returned by Twitter's API. This measure has been studied previously (Ratkiewicz *et al.*, 2011a; Xie *et al.*, 2008), and hypothesizes that bots are motivated to persuade real users into visiting external sites operated by their controller. In this way, this heuristic traps bots that are trying to promote URLs in their tweets.

- Average Time Between Tweets: it has been discovered that many bots tweet in a "bursty" nature (Lee and Kim, 2014; Xie *et al.*, 2008), publishing many of

their tweets within a short amount of time. This behavior is measured as:

$$Heuristic_{Time}(u) = \frac{1}{|tweets^u| - 1} \sum_{i=2}^{N} (t_i - t_{i-1}), \tag{4.14}$$

where t_i is the time stamp of the i-th tweet in u's timeline, sorted chronologically in ascending order (i.e. $t_i \geq t_{i-1}$).

We apply each heuristic, as well as AdaBoost and BoostOR to both the Libya dataset, shown in Table 4.3, and the Arabic Honeypot dataset, shown in Table 4.4. We find that both AdaBoost and BoostOR outperform the heuristics, with BoostOR performing the best on both datasets. Interestingly, we find that SVM underperforms, achieving a worse result than the heuristics when optimizing the F_1 score.

One thing to note about the heuristics is that they conform to the class distribution. That is, when they achieve their maximum performance they are simply *always* predicting the user is a bot. In other words, the results of heuristic measures indicate that we should delete our entire dataset for both datasets. While this yields what seem to be competitive results, the implication of this approach is not reasonable. While the F_1 score is useful to measure the performance, we need to dig deeper to understand the implication of these results.

To illustrate the difference in performance between the heuristics and our proposed model, we show a confusion matrix of the best-performing heuristic, $Heuristic_{Time}$, and a confusion matrix of *BoostOR* in Table 4.2 on the Arabic Honeypot dataset. First, we notice that the heuristic *never* misclassifies as a human as a bot. This is in line with the goals of the industry. However, we see in the *BoostOR* results that we achieve superior performance by lowering the false-negative rate of the model, which is in line with the goals of the researcher.

Table 4.2: Confusion Matrix for the $Heuristic_{Time}$ and BoostOR on the Arabic Honeypot Dataset.

		Prediction			
		$Heuristic_{Time}$		BoostOR	
		Bot	**Human**	**Bot**	**Human**
Truth	**Bot**	49.9%	50%	42.8%	7.7%
	Human	0%	0.1%	14.8%	34.7%

4.3.2 Evaluating REFOCUS

first, we investigate the effect of β. We evaluate the classification results using values of $\beta \in \{1, 2, 3, 4, 5\}$ and show that F_2 performs the best. Second, we explain the choice of classification algorithm and model parameters. Then, we compare our approach with a state-of-the-art bot detection model, BotOrNot (Davis *et al.*, 2016).

Table 4.3: Comparison Between BoostOR and Heuristics on the Libya Dataset.

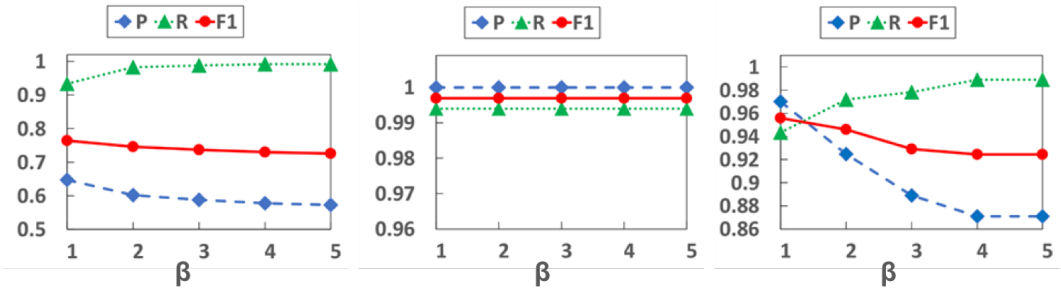
Method	Precision	Recall	F_1
$Heuristic_{URL}$	6.74%	65.12%	12.21%
$Heuristic_{Retweet\%}$	7.73%	53.63%	13.51%
$Heuristic_{Length}$	7.74%	53.63%	13.51%
$Heuristic_{Time}$	7.48%	99.89%	13.91%
SVM	29.24%	8.78%	13.53%
$AdaBoost$	75.25%	7.48%	13.61%
$BoostOR$	75.41%	8.14%	14.69%

Table 4.4: Comparison Between BoostOR and Heuristics on the Arabic Honeypot Dataset.

Method	Precision	Recall	F_1
$Heuristic_{URL}$	49.69%	96.39%	65.58%
$Heuristic_{Retweet\%}$	50.05%	99.33%	66.56%
$Heuristic_{Length}$	50.00%	99.82%	66.63%
$Heuristic_{Time}$	49.99%	99.96%	66.65%
SVM	62.41%	62.52%	62.47%
$AdaBoost$	79.76%	72.41%	75.91%
$BoostOR$	71.42%	82.48%	76.55%

Searching for the Right β

It is intuitive that using a F_β when $\beta > 1$ for training a classifier helps us find the classification threshold that results in higher recall as compared to when $\beta = 1$. However, it raises two questions: (1) what is best value of β and can we increase it indefinitely to reach the highest recall possible? (2) Does the model trained using $\beta > 1$ still perform well in terms of F_1 or we will drastically lose precision? We answer the first question here and the second one in Section 4.3.2.



(a) (b) (c)

Figure 4.6: Effect of β on Precision (P), Recall (R), and Overall Performance (F_1). In Each Dataset, We Change β from 1 to 5, Use F_β for Finding the Best Classification Threshold in the Training Phase and Report P, R, and F_1 on the Test Set.

We test our model using the framework in Fig. 4.4 on three datsets: Arabic Honeypot, Social Spambot 1, and Social Spambot 2. The resutls are shown in Fig. 4.6. In Social Spambots 1, we do not observe any change in the overall performance in terms of F_1 as we change the β. Hence, any of the F_β scores can be used to find the best classification threshold. In Arabic Honeypot and Social Spambot 2, we see some variations in precision, recall, and overall performance. $\beta = 2$ gives us the best trade-off between precision and recall because if we go from F_1 to F_2 on the x axis, we gain on recall but lose on the overall performance due to the decrease in precision. But, our focus is on recall and we are willing to lose on overall performance if we can gain significantly on recall. Hence, we prefer F_2 over F_1 because the loss in the overall performance is smaller that the gain in recall; in other words, the slope of recall line is larger than the slope of F_1 line. Further increase in β does not provide enough gain on recall in comparison to the loss in the overall performance; we do not see sharp slopes anymore.

Model Parameters

For comparing the overall performance of REFOCUS with other bot detection methods, we need to decide on the number of topics in LDA and the classification model. Due to the similarity between our feature extraction and the one by Morstatter et al. (Morstatter *et al.*, 2016) we follow their observation that 200 topics generated the highest F_1 in the Arabic Honeypot dataset and set number of topics to 200.

We test multiple classification algorithms that are observed to have high performance in the problem of bot detection (Alothali *et al.*, 2018) to find the best fit for REFOCUS: Decision Tree, Random Forest, and SVM. We use Python Scikit-learn package (Pedregosa *et al.*, 2011) for implementation with default settings except $max\_depth = 1$. As shown in Table 4.5. All classifiers achieve very similar (differ-

ence less than 0.5%) F_1 score except for Random Forest that has lower performance. We choose SVM for the rest of our experiments because it has similar or higher R and similar F_1 . Worth mentioning that our method can be built on top of any classifier to help improve recall without sacrificing the overall performance.

Table 4.5: Performance of REFOCUS When Implemented Using Different Classifiers.

	Arabic Honeypot			Social Spambot 1			Social Spambot 2		
Classifier	P	R	F_1	P	R	F_1	P	R	F_1
Decision Tree	0.726	0.771	0.745	0.995	0.996	0.996	0.995	0.995	0.995
Random Forest	0.787	0.769	0.777	0.995	0.985	0.990	0.995	0.993	0.994
SVM	0.657	0.916	0.765	1.0	0.991	0.995	0.998	0.992	0.995

Table 4.6: Comparison Between REFOCUS and Baseline Bot Detection Methods.

Dataset	**Method**	*P*	*R*	F_1	*ROC*
Arabic Honeypot	SVM	0.655	0.919	0.765	0.849
	REFOCUS	0.601	0.983	0.746	0.849
	BotOrNot	0.472	0.523	0.496	0.514
Social Spambot 1	SVM	1.0	0.991	0.995	0.997
	REFOCUS	1.0	0.993	0.996	0.997
	BotOrNot	0.963	0.961	0.962	0.969
Social Spambot 2	SVM	0.986	0.915	0.949	0.996
	REFOCUS	0.924	0.971	0.945	0.996
	BotOrNot	0.954	0.939	0.946	0.957

Testing the Overall Performance

We compare our proposed approach, REFOCUS, with two baselines:

- *SVM*: REFOCUS uses SVM to train multiple classifiers on subsamples of the dataset and learns the best recall-precision trade-off using F_β. Hence, we compare our method with SVM when its parameters are set to default and it generates the class labels (1 or -1) using 0.5 as threshold. Users are represented with 200 LDA topics and we use 10-fold cross validation.

- *BotOrNot* (Davis *et al.*, 2016; Varol *et al.*, 2017): this supervised bot detection model exploits 1150 features in six categories: user-based, friends, network, temporal, content and language, and sentiment. The model uses a Random Forest classifier and is trained on multiple publicly available datasets. BotOrNot has been used for generating ground-truth due to its performance.

We perform two sets of experiments. In the first one, we use the Arabic Honeypot dataset. We use an LDA model with 200 topics to extract features from the dataset then we apply REFOCUS and report the results. However, using this dataset raises the concern that our approach might not perform as well on non-Arabic tweets. Hence we also perform the second experiment. We follow the same procedure but use the datasets that were collected by Cresci et al. (Cresci *et al.*, 2017). These datasets (as explained in Section 4.1.1) have three advantages: they are among the most recent publicly available labeled datasets for bots, they include new and more complicated bots, and a majority of the tweets are in English. Hence, by testing our approach on Cresci's datasets, we show that our model performs well regardless of the language of tweets and is resilient to new types of bots that emerge on social media.

The results are presented in Table 4.6. For the experiments on Cresci's datasets, we do not balance the classes due to small size of the data. Hence, we also include the ROC AUC in our results. The ROC AUC for a classifier that randomly assigns labels to instances is 0.5 regardless of the class balance and is a helpful metric to

assess classifiers when the samples of one class are more than the other. Reserving the class imbalance is also helpful to mimic the real world scenario where bots are a small portion of all users on social media (Varol *et al.*, 2017).

In the first experiment, on the Arabic Honeypot dataset, SVM has higher precision and lower recall in comparison to REFOCUS. The reason is that SVM only labels a user as bot if the predicted probability of being a bot for that user is over 0.5. However, our method learns the best threshold for optimizing recall while reaching a high F_1. Hence REFOCUS chooses a lower threshold (0.35 in this case). This choice results in 2% lower F_1, however, we are willing to tolerate this loss due to 6% gain in recall. BotOrNot performs considerably worse in this dataset in comparison to the Social Spambot datasets. The reason is that Social Spambot datasets have been used in training BotOrNot and it is expected for classifiers to have lower performance on unseen datasets (e.g. Arabic Honeypot).

In the second experiment, we test our method on two non-Arabic datasets which are obtained using a manual annotation method to show that our results are robust to variations in datasets such as language. In Social Spambots 1, SVM and our proposed approach perform almost identically with an slightly better recall in REFOCUS. The reason is that the differences between instances in human and bot classes are well captured by the classifiers to the extent that the classifier (either SVM or REFOCUS) are very confident in the labeling. Hence, each instance gets a high probability of being in its actual class and changing the threshold does not change the classification results much. We also observe that our approach outperforms BotOrNot. On Social Spambots 2, SVM and BotOrNot outperform our approach in precision and have lower recall, similar to the Arabic Honeypot dataset, because they are not is not designed to optimize on recall. F_1 of our approach is similar to the baselines.

4.4 Summary

The dominant trend among the previously proposed methods for bot detection is solely focusing on precision, making sure that no human user is marked as a bot, or optimizing for F_1. In this work, we showed that we can focus on recall of a bot detection model without sacrificing the overall performance. We tested our method on three real-word datasets and observed that using F_2 score in the training phase results in finding the best classification threshold for optimizing recall and having high overall performance in terms of F_1. In the future, we wish to explore the robustness of our method on translated datasets and also measure its effectiveness in discriminating different types of bots in a dataset.

Chapter 5

DETECTING FAKE CONTENT ON SOCIAL MEDIA

Due to the increasing amount of our time spent on social media platforms, it is no surprise that people tend to receive their news content through social media sites more than before. For example, a quarter of US adults used social media as their main source of election news and campaign updates during the US presidential election in 2016 (Shearer, 2016). This high rate of engagement with online news can mainly be attributed to the nature of the social media platforms themselves, which are typically inexpensive, provide easy access, and support fast dissemination not possible through traditional media outlets. However, despite these advantages, the quality of news on social media is considered lower than that of traditional news outlets. A factor contributing to this low quality is wide-spread of fake news articles online; fake news articles are low-quality news stories with intentionally false information (Shu *et al.*, 2017). During the US presidential election in 2016, fake stories supporting one candidate were shared 30 million times on Facebook and over one million tweets related to the fake news story "Pizzagate" [1] were shared on Twitter.

Fake news has several significant negative societal effects. First, people may accept deliberate lies as truths. The likelihood of accepting fake news as true increase after observing it repeatedly (Hasher *et al.*, 1977) especially when it aligns with the user's beliefs (Weir, 2017). Second, fake news may change the way people respond to legitimate news. When people are inundated with fake news, the line between fake news and true news become more uncertain, fake news spreaders make users doubt the nature of real news at the least and leave them in the mindset that everything is

[1] `https://en.wikipedia.org/wiki/Pizzagate\_conspiracy\_theory`

biased and conflicted and it is impossible to distinguish fake from real news (Lynch, 2016). Finally, the prevalence of fake news has the potential to break the trustworthiness of the entire news ecosystem. For example, in the last three months leading to the US presidential election in 2016, fake news stories posted on Facebook gained more engagement by users in comparison to true news (Silverman, 2016). Thus, it's critical to detect fake news on social media to mitigate these negative effects, hopefully benefiting the general public and the entire news ecosystem.

There are three aspects to the problem of detecting fake news: malicious users who generate fake news, the fake news articles/posts and their content, and the users who are exposed to fake news articles. Majority of fake news detection approaches focus on the first two aspects and oversee the users who are exposed to fake news articles. For example, researchers have used the content of posts and context surrounding them to detect fake news and malicious actors. Approaches that incorporate content use linguistic and visual features and context-based approaches exploit features extracted from users, posts, and network. Recent research advancements aggregate users' profiles and engagements on news pieces to help infer which articles are fake (Castillo *et al.*, 2011; Jin *et al.*, 2016), giving some promising early results. However, no principled study is conducted on characterizing the users who are exposed to fake news articles.

In this work, we extensively study the users who are exposed to fake news through the lens of social media. As a subset of users who were exposed to fake news, we focus on the ones who share the stories of those articles on Twitter due to the fact that it is impossible to access all users who have read an online news article. We provide a deep understanding of (i) what the characteristics of users who spread fake news are; (ii) how well these features can discriminate users who spread fake news from the ones who spread true news; (iii) how we can use the discriminative features of these

users in the task of detecting fake news articles. We introduce a set of features with foundations in social psychology and behavioral studies. Then we investigate to what extent users who spread fake news and the ones who share real news are distinct in terms of these features. Finally, we extend our study to the problem of detecting fake news articles. To this end, we investigate the following hypothesis:

Do the users who spread fake news on social media show the same characteristics as the fake news spreaders in traditional media as studied in the psychological theories?

5.1 Method

To understand the characteristics of users who spread fake news, we rely on psychological theories on fake news, rumor, gossip, and conspiracies. All these concepts share similarities such as recency and connection to current events, circulating in the context of ambiguity, danger, or potential threat, and having major outbreaks during main political events and natural disasters such as presidential elections and major hurricanes (DiFonzo and Bordia, 2007). Although there is a large body of work on these psychological theories, not many of them can be (1) applied to users and their behaviors on social media and (2) quantitatively measured for fake news articles and spreaders on social media. Hence, Based on psychological theories, we enumerate four categories of features that can potentially express the differences between users who spread fake news and the ones who spread real news. Then, we propose an approach for measuring each on social media. We emphasize that the features we name in this section are obtained from a wide range of physiological studies that goes beyond fake news and we do not claim that all these factors are proven to have an effect in the spread of fake news but *potentially* can. In the remainder of this section, we explain these features and their theoretical origins.

5.1.1 Motivational Factors

Based on an extensive study by DiFonzo and Bordia (DiFonzo and Bordia, 2007), we list four factors that potentially affect the spread of fake news:

Uncertainty: spreading fake news can be a sense making activity in ambiguous situations and the frequency of fake news increases in uncertain situations, such as natural disasters or political events such as elections when people are unsure of the results. When a crisis happens, people first seek information from official sources. However, in the lack of such information, they form unofficial social networks to make predictions with their own judgment and fill the information gap (Rosnow and Fine, 1976). This might result in generating fake news such as a fake image of a shark swimming on the highways of Houston after Hurricane Harvey or millions of fake news posts that were shared on Facebook in the weeks leading to the US presidential election 2016.

Anxiety: emotional pressure can play an important role in spreading fake news and can be triggered by emotions such as frustration, irritation, and anxiety. Anxiety can make people more prone to spreading unproved claims and less accurate in transmitting information (DiFonzo and Bordia, 2007). In high anxiety situations, fake news can work as a justification process to relief emotional tension (Allport and Postman, 1947). Fake news might be used as a method of expressing emotions in anxious situations that allows people to talk about their concerns and receive feedback informally; this process results in sense making and problem solving (Waddington, 2012). For example, during the devastating time of Hurricane Harvey, 2017, a fake news story accusing Black Lives Matter supporters of blocking first responders reaching the affected area was spread by more than one million Facebook users (Grenoble, 2017). Believing and spreading such fake news stories may help the people in disaster

areas cope with the anxiety caused by delays in relief efforts (Fernandez *et al.*, 2017).

Importance or outcome-relevance: people pursue uncertainty reduction only in the areas that have personal relevance to them. For example, when a murder took place in a university campus, rumor transmission in the people from the same campus was twice the people who were from another university campus in the same city. Due to the difficulty of measuring importance, anxiety is often used as a proxy; being anxious about a fake news story shows importance (Anthony, 1973).

Lack of control: Fake news represents ways of coping with uncertain and uncontrollable situations. When people do not have primary control over their situation (action-focused coping responses), they resort to secondary control strategies which are emotional responses such as predicting the worst to avoid disappointment and attributing events to chance. Two secondary control themes are explaining the meaning of events and predicting future events.

Belief: building and maintaining social relations are vital to humans, hence, to ensure their reputation as a credible source of information, they tend to share the information in which they believe. Belief is found strongly related to transmission of rumors. When it comes to political issues people tend to rationalize what they *want* to believe instead of attempting to find the truth. Hence, persuading themselves to believe what is aligned with their prior knowledge. This observation extends to the limit that appending corrections to misleading claims may worsen the situation; people who have believed the misleading claim may try to find reasons to dispute the corrections to an extent that they will believe in the misleading claim even more than before (Pennycook and Rand, 2019).

5.1.2 Social Engagement

Allcott and GentzkowSocial (Allcott and Gentzkow, 2017) studied different factors that are correlated with belief in fake news. They found three correlations to be statistically significant, one of which is how much time people spend consuming media. Social media users engage more with social media were less likely to believe fake news after the US presidential election in 2016. This observation indicates that users who occasionally refer to social media for news consumption are at higher risk of being impacted by fake news.

5.1.3 Position in the Network

In a study on workplace emails (Mitra and Gilbert, 2012), researchers observed that employees with the lowest rank had a major contribution to the circulation of information that is not confirmed to be true. Hence, we propose that the ranks of social media users might affect how they react to fake news. There is not a unique way of defining a rank for users on social media and lack of complete network information, in most cases, limits our choices. Measures such as popularity or influence are examples of rank indicators in social networks.

5.1.4 Relationship Enhancement

Enhancing social relations are one of the goals of spreading unverified information (DiFonzo and Bordia, 2007) and conspiracies (Kou *et al.*, 2017). In short term social relations formation, with the purpose of generating a positive effect in others, people may spread information to attract or hold attention without verifying its authenticity. Also, in friendship relations, negative fake news might be transmitted as a warning mechanism to provide the information that is relevant, useful, and pre-

vents harmful consequences (Weenig *et al.*, 2001). Another aspect of enhancing social relations is using unverified information with self enhancement motivations such as demoralizing enemy troops in war or downgrading the reputation of an opposing candidate in an election (Kapferer, 2013). If we consider follow relations on Twitter a resemblance of friendship relations in the real world, the same mechanism might be used by social media users to gain attention or warn the followers by spreading fake news about the the presidential candidate they oppose to prevent harm that might be caused by the election of them or gain more popularity for their favorite candidate. Moreover, one of the motivations for sharing conspiracies is entertaining and initiating conversations. This behavior is observed in the form of expressing uncertainty about the topic under discussion to build or shift the attention to other (opposing) aspects (Kou *et al.*, 2017).

5.2 Data

We use two datasets from FakeNewsNet repository (Shu *et al.*, 2018) in explaining our method (Section 5.4) and experiments (Section 5.5): PolitiFact and Gossip Cop. PolitiFact[2] is a fact-checking website operated by Tampa Bay Times[3], where reporters and editors from the media fact check the political news articles. The URLs of news articles are available on the PolitiFact website and is used to collect the tweets related to them. Gossip Cop[4] is a website for fact-checking entertainment stories aggregated from various media outlets. On Gossip Cop website, articles get a score between 0 and 10 as the degree from fake to real. However, URL or the headline of the articles are not explicitly mentioned. Hence, some heuristics are used to find the headline of the stories through Google search. When the headline of fake and real stories identified

[2]https://www.politifact.com/

[3]https://www.tampabay.com/

[4]https://www.gossipcop.com/

by PolitiFact and Gossip Cop are extracted, a crawler collects all the tweets referring to those stories. Further, recent posts, profile, and list of followers and friends of Twitter users who posted about fake and real stories are collected. In this paper, we limit our analysis to users who have at least 3 fake or real news to focus on users who actively tweet news stories. The statistics of these two datasets is provided in Table 5.1.

Table 5.1: Statistics of he Datasets.

Dataset	**News Category**	**Users**	**News Tweets**	**Timeline Tweets**
PolitiFact	Real News	12,571	88,930	2,457,146
	Fake News	6,330	45,184	1,241,317
Gossip Cop	Real News	2,831	134,055	555,457
	Fake News	29,384	267,803	2,283,225

5.2.1 Feature Extraction

All the features introduced in Section 5.1 can be extracted from the content (tweets), activity (tweeting behavior and likes), or network (followees and followers) of Twitter users in PolitiFact and Gossip Cop datasets. For measuring the features that are extracted from content, we exploit Linguistic Inquiry and Word Count (LIWC) (Tausczik and Pennebaker, 2010). LIWC includes a dictionary that lists a set of words for some psychologically-relevant categories such as positive and negative emotions, social relationships, and honesty and deception. For example, Positive Emotions category includes words such as happy, pretty, and good. Each word can belong to more than one category; for example, the word cried is part of five-word categories: Sadness, Negative Emotion, Overall Affect, Verb, and Past Focus. To measure a feature using an LIWC category, we find the percentage of words in a

preprocessed tweet that belongs to that category. The preprocessing step includes removing English stopwords, punctuations, URLs, hashtags, and mentions. Next, we explain how each of the features is calculated:

- Motivational Factors: there are three LIWC categories that are related to *uncertainty*: Discrepancy (e.g. should, would, and could), Tentativeness (e.g. maybe, perhaps, and guess), and Certainty (e.g. always and never). These categories are abbreviated as `discrep`, `tentat`, and `certain` respectively. *Anxiety* can be measured using the LIWC Anxiety category (`anx`) which includes words such as nervous, afraid, and tense. *Importance or outcome-relevance* is observed to be a difficult feature to measure in psychology so researchers suggest using proxies to quantify *importance*; we use *anxiety* as a proxy for measuring this feature, meaning that people are more anxious about a topic which is more important to them. We use LIWC Future Focus (`futurefocus`) to measure *lack of control*, this category includes words such as may, will, and soon. We do not measure *belief* explicitly because we assume that any user who tweets fake news articles believes in it.

- Social Engagement: the more a user is involved with social media the less likely it is for her/him to be misguided by fake news. We measure social engagement on Twitter using the average number of tweets per day.

- Position in the Network: this feature can be quantified using a variety of metrics when the network structure is known. However, in the case of social networks between Twitter users, we do not have complete structure and even collecting local information is time consuming due to the rate limitation of Twitter APIs. Hence, we extract *influence* using the number of followees and *popularity* using the number of followers of each user.

- Relationship Enhancement: improving the relation to other social media users and gaining more attention from the community is one of the motivations for spreading fake news. If the number of retweets and likes of a fake news post is higher than the average number of retweets, likes and of the spreader, it can indicate that this user has enhanced their social relation and initiated conversation. Hence, we use the difference between the number of retweets and likes of fake news posts and average values for each user as indicators of relationship enhancement motivation.

The summary of features and the metrics used to measure them is presented in Table 5.2. Based on this list we extract eleven features in four categories for each user in our datasets.

Table 5.2: Summary of the Metrics Used to Measure Features of Users Who Spread Fake News.

Feature Category	**Feature Name**	**Metric**	**Example Words**
Motivational Factors	Tentativeness	LIWC `tentat`	maybe, perhaps
	Discrepancy	LIWC `discrep`	should, would
	Certainty	LIWC `certain`	always, never
	Anxiety	LIWC `anx`	worried, fearful
	Lack of Control	LIWC `focusfuture`	may, will, soon
Social Engagement	Social Engagement	Avg Tweets per Day	-
Position in the Network	Influence	#Folowees	-
	Popularity	#Followers	-
Relationship Enhancement	Boosting #Retweets	Increase in Retweets	-
	Boosting #Likes	Increase in Likes	-

5.3 Experiments

We show the power of our proposed method in terms of (1) how it can be used in discriminating fake news spreaders from real news spreaders and (2) how this can be used in detecting fake news content.

5.4 Fake News vs. Real News Spreader

We study the differences between users who spread fake news and the ones who spread real news in terms of four feature categories introduced in Section 5.1. We perform two experiments toward this goal. First, we set the null hypothesis is that these two groups have the same mean in four categories of features. If the results of T-test shows that there is a significant difference (p-value less than 0.05) between two groups, we can reject the null hypothesis. Second, we use the proposed features to see if a supervised model (a classifier) can use these features to discriminate the two groups of users.

As shown in Table 5.3, in the Motivational Factors category, we observe a significant difference (p-value < 0.005) between users who share fake news and the ones who share real news in terms of all features except Anxiety. We believe one reason for this observation is that Gossip Cop articles are mostly concerned with celebrity news and regardless of the true or false nature of the news being propagated, the news has a low potential of causing distress or being propagated to address anxiety in the spreader. On the other hand, Politifact contains news article about a variety of topics, including natural disasters and politics. Hence, there is more potential for these articles to be circulated due to the anxiety of the spreader.

In social engagement, Politifact users who share fake news have significantly lower number of tweets per day but we do not observe a significant difference among users

in the Gossip Cop users. We used the Influence (number of followees) and popularity (number of followers) as indicators of the status in the network. Except for the Influence in the Politifact users, we observe a significant difference among users who spread real news and the ones spreading fake news. We also expected to see different number of likes and retweets for the fake news tweets which is only observed in reshares feature; indicating that Boosting #Likes can be a motivation for spreading fake news on social media.

Table 5.3: Comparison Between Fake News and Real News Spreaders in Terms of Psychological Features. The Features That Are Significantly Different Between Users Who Spread Fake News and the Ones Who Spread Reals News Are Marked With ** (p-value <0.005) or * (p-value <0.05)

Feature Category	**Feature Name**	**t-statistic Politifact**	**t-statistic Gossip Cop**
Motivational Factors	Tentativeness	-21.62**	14.44**
	Discrepancy	-13.55**	10.94**
	Certainty	-7.65**	13.15**
	Anxiety	-2.05*	0.94
	Lack of Control	-3.96**	13.34**
Social Engagement	Social Engagement	-13.16**	-0.41
Position in the Network	Influence	-0.26	2.74**
	Popularity	-2.69*	2.10*
Relationship Enhancement	Boosting #Retweets	-3.78**	9.70**
	Boosting #Likes	-1.54	-0.70

5.5 Predictive Power of Psychological Features in Fake News Detection

In this section, we shift our focus from the users who spread fake news to the news articles; we test the predictive power of these features in detecting fake news

articles on social media. We use a supervised method to classify news articles into fake and real. Each news article is represented by an 11-dimensional vector based on psychological and social features. The vector for each article is the mean of vectors for all the users who have tweeted about that article and have at least three fake or real news tweets. We test classifiers, Random Forest, Decision Tree, and SVM, to show the classification results. We measure the performance in terms of Precision, Recall, F1 Score, Accuracy, and ROC AUC.

As shown in Table 5.4, Random Forest has the highest accuracy and ROC AUC. Accuracy indicates that using these features, 81% and 84% of fake and real news articles in Politifact and Gossip Cop, respectively, are correctly labeled. Receiver Operating Characteristic (ROC) curve is created by plotting the true positive rate against the false positive rate at various threshold settings. The Area Under Curve (AUC) and ROC curve is used as a measure of sensitivity (the most sensitive classifier has no false negative with AUC equal to one) and is equal to 0.5 for random guess classifier. Using the features proposed in this work, we reach 0.91 and 0.93 ROC AUC for Politifact and Gossip Cop datasets respectively. The Precision, Recall, and F1 Score reported in Table 5.4, indicate the performance of the classifier incorrectly labeling the fake news articles. In the Politifact dataset, 84% of the articles labeled as fake news are correctly labeled and 81% of all the fake news articles are found by the classifier. In the Gossip Cop dataset, these values are 91% and 81% respectively.

In Section 5.4, we presented how the users who spread fake news tweet are significantly different from the ones who spread real news tweets in terms of the four categories of features. Here, we discuss the role of these features in detecting fake news articles using the features of their spreaders on Twitter. Using the feature importance in the Decision Tree Classifier that labels articles as fake or real (Table 5.5),

Table 5.4: Performance of the Psychological Features in Detecting Unobserved Fake News.

Metric	Politifact				Gossip Cop			
	REFOCUS	**Random Forest**	**Decision Tree**	**SVM**	**REFOCUS**	**Random Forest**	**Decision Tree**	**SVM**
Precision	0.73	0.84	0.80	0.77	0.76	0.91	0.89	0.83
Recall	0.98	0.81	0.83	0.70	0.97	0.81	0.89	0.70
F1 Score	0.84	0.83	0.81	0.73	0.85	0.86	0.89	0.76
ROC AUC	0.93	0.91	0.79	0.79	0.94	0.93	0.87	0.82

we observe that the most important features for both datasets are in the relationship enhancement category. This observation is an indicator that tweeting about fake news on social media can be due to the motivation of gaining more attention from social circles via collecting likes and retweets.

If we compare the non-significant features in Tables 5.3 and 5.5, we observe that all those features have a low importance in detecting fake news articles except the

Table 5.5: Importance of Features in Detecting Fake News Articles. the Results Are Feature Importance Scores Generated by a Decision Tree Classifier for Labeling Articles as Fake or Real Using the Proposed Features.

Feature Category	**Feature Name**	**Feature Weight Politifact**	**Feature Weight Gossip Cop**
Motivational Factors	Tentativeness	0.09	0.05
	Discrepency	0.12	0.07
	Centainty	0.06	0.07
	Anxiety	0.05	0.05
	Lack of Control	0.02	0.11
Social Engagement	Social Engagement	0.03	0.12
Position in the Network	Influence	0.03	0.02
	Popularity	0.07	0.03
Relationship Enhancement	Boosting #Retweets	0.06	0.34
	Boosting #Likes	0.42	0.05

Increase in Likes for the Politifact dataset and Average Tweets per Day for the Gossip Cop dataset that have the highest and the second highest importance. This is an interesting observation that shows, although these two features are not the most significant features when focusing on the user who spread news, they still play an important role when we take into consideration all the users who spread one piece of news to label it as fake or real.

5.6 Summary

In this work, we investigated whether the psychological features that are observed in users who spread fake news in the behavioral studies on human subjects are valid for social media users who spread fake news on social media. Toward this goal, we introduce four categories of features based on psychological theories that can be quantified for social media users. Based on our observations on two real-world datasets, we observed that (i) social media users who spread fake news are significantly different in terms of the majority of these features and (ii) these features have predictive power in the detecting new and unobserved fake news articles.

This study is a first step towards understanding users who are exposed to fake news. We wish to continue this work by collecting a larger set of psychological features such as personality traits and study them on news consumers on social media. Moreover, we plan to study the application of the psychological features in improving the performance of state-of-the-art fake news detection algorithms.

Chapter 6

CONCLUSION AND FUTURE WORK

In this chapter, we conclude the dissertation with enumerating our methodological contributions and also our future directions.

6.1 Methodological Contributions

The contributions of our work throughout this dissertation is summarized as follows:

1. **Understanding Disasters through the Lens of Social Media**: I introduced my studies on social media after major disasters in Chapter 3. I first extracted the socio-temporal stages of disasters during which the users provide a large number of posts on social media. This abundance of information is necessary for Machine Learning models to perform well. Then, I proposed a supervised model to filter actionable tweets from all millions of tweets that are posted after major disasters. A contribution of our work is we used the context (meta data of tweets such as URLs, hastags, and user profiles) along with the content. Context is more accessible and less expensive to process but it also provides a strong signal regarding the type of tweet (actionable or not).

2. **Understanding Bots who Spread Malicious Content**: in Chapter 4, I provided the details of two bot detection methods that we proposed. Both of these methods are supervised models that aim at optimizing the recall of the classification task while maintaining a high overall performance. The main contribution of our models is focusing on recall as opposed to the majority

of the previous studies. A bot detection model that has a high recall, helps clean social media datasets from non-human actors and also prevent users from interacting with bots.

- BoostOR: our first attempt (Section 4.2.2 in introducing a recall-focused approach is called BoostOR. This method is a variation of AdaBoost classifier in which we change the weights of the misclassified instances in every iteration. The main different between BoostOR and AdaBoost is that we only increase the weight of misclassified bots, hence, the classifier is penalized more if it misclassifies bots. As a results we achieve higher recall on real-world datasets and outperform other algorithms in overall performance ($F_1 score$).
- REFOCUS: in our second attempt (Section 4.2.3), we looked at the problem of bot detection from a new perspective. We proposed a method that given a dataset and a based classifier finds the sweet spot between precision and recall at which we have high recall without sacrificing the overall performance. We showed that using F_2 score during the training phase helps us find the best classification threshold to achieve our goal. Moreover, our overall performance in terms of F_1 score is similar to the state-of-the-art bot detection methods on real-world datasets.

3. **Understanding Users who are Affected by Malicious Content**: bots, especially if generated in large numbers, can significantly increase the visibility and reach of malicious content on social media. However, without benign users (i.e. human users) who believe and reshare this malicious content, the spread cannot be viral. In Chapter 5, we studied the uses who helped spread malicious content on social media. We introduced five categories of features based on

psychological theories to understand the characteristics of these users. These features helped us understand what motivates users to spread malicious content. We also, studied the potential of these features in detecting fake news.

6.2 Future Directions

1. **Fake News in Disasters**: An application of Machine Learning models is detecting actionable posts in the aftermath of disasters. An underlying assumption of such models is that the social media posts are generated by legitimate users and are not intentionally false. However, there are at least two scenarios in which this assumption is not valid: (1) when the posts are generated by malicious actors such as bots or (2) when posts contain false information such as fake content. A situation which falls into these scenarios is when a users posts a "fake" help-seeking tweet. These tweets can be very similar in content and structure to the help-seeking of legitimate users in need and it makes filtering them even more difficult. We believe that exploiting more features from the users, beside the content, can help improve the Machine Learning models designed for this task. Examples of such features are profiles of users, their coordination with other users, and also positing behaviors.

2. **Discovering Types of Bots on Social Media**: alighted with the majority of studies on social bots, we proposed methods to distinguish bots from humans. Our methods, although look at this problem from a new perspective, they do not have the capability to dissect bots into their types. Different types of bots have been observed on social media: Duplicate Spammers, Duplicate @ Spammers, Malicious Promoters, and Friend Infiltrators. But, developing methods to detect these types remains overlooked. We believe that semi-supervised models can

be a fit to this problem; an supervised model to find botnets (clusters of bots) and a supervised model to find which type of bot each botnet is.

3. **Evolution of Bots over Time**: An study in 2011 (Lee *et al.*, 2011) shows that bots have at least four categories. Since 2011, bots have evolved significantly and became extremely more difficult to detect. These malicious actors have been active during major events such as the the 2016 US Presidential Election and are expected to be as destructive in the 2020 US Presidential Election. Hence, it is important to understand how different types of bots have evolved since that study. An understudied problem in the area of bot detection is understanding how bots evolve over time. It is difficult to observe how the characteristics of a fixed set of bots change as many of them are banned by the regulators. However, we can study the distribution of the known types of bots across events and over time. Moreover, it is interesting to see what new types of bots are generates especially during major events such as as elections and natural crises.

4. **Exploiting Psychological Features of Users in Detecting Fake News**: our studies in Chapter 3 showed that users who spread malicious content, especially fake news in our study, have different psychological features. This finding shed light on the motivations and characteristics of these users. It is valuable to study how we can incorporate user features in a fake news detection method. Majority of fake news detection approaches use the content or user profile features. Hence, users' psychological features are a valuable addition towards improving the fake news detection models.

REFERENCES

Abbasi, M.-A., S. Kumar, J. A. Andrade Filho and H. Liu, "Lessons learned in using social media for disaster relief-ASU crisis response game", in "International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction", pp. 282–289 (Springer, 2012).

Abokhodair, N., D. Yoo and D. W. McDonald, "Dissecting a social botnet: Growth, content and influence in Twitter", in "Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing", pp. 839–851 (ACM, 2015).

Ahmed, F. and M. Abulaish, "A generic statistical approach for spam detection in Online Social Networks", Computer Communications **36**, 10-11, 1120–1129 (2013).

Allan, J., J. G. Carbonell, G. Doddington, J. Yamron and Y. Yang, "Topic detection and tracking pilot study final report", in "Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop", pp. 194–218 (1998).

Allcott, H. and M. Gentzkow, "Social media and fake news in the 2016 election", Journal of Economic Perspectives **31**, 2, 211–236 (2017).

Allem, J.-P., E. Ferrara, S. P. Uppu, T. B. Cruz and J. B. Unger, "E-Cigarette Surveillance With Social Media Data: Social Bots, Emerging Topics, and Trends", JMIR Public Health and Surveillance **3**, 4, e98, URL `http://publichealth.jmir.org/2017/4/e98/` (2017).

Allport, G. W. and L. Postman, *The psychology of rumor.* (Oxford, England: Henry Holt, 1947).

Alothali, E., N. Zaki, E. A. Mohamed and H. Alashwal, "Detecting Social Bots on Twitter: A Literature Review", 2018 International Conference on Innovations in Information Technology (IIT) pp. 175–180, URL `https://ieeexplore.ieee.org/document/8605995/` (2018).

Anthony, S., "Anxiety and rumor", The Journal of social psychology **89**, 1, 91–98 (1973).

Asur, S. and B. A. Huberman, "Predicting the future with social media", in "Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on", vol. 1, pp. 492–499 (IEEE, 2010).

Atefeh, F. and W. Khreich, "A survey of techniques for event detection in twitter", Computational Intelligence **31**, 1, 132–164 (2015).

Backstrom, L., E. Sun and C. Marlow, "Find me if you can: improving geographical prediction with social and spatial proximity", in "Proceedings of the 19th international conference on World wide web", pp. 61–70 (ACM, 2010).

Becker, H., F. Chen, D. Iter, M. Naaman and L. Gravano, "Automatic identification and presentation of twitter content for planned events.", in "ICWSM", pp. 655–656 (2011a).

Becker, H., M. Naaman and L. Gravano, "Beyond trending topics: Real-world event identification on twitter.", ICWSM **11**, 438–441 (2011b).

Beigi, G., X. Hu, R. Maciejewski and H. Liu, "An overview of sentiment analysis in social media and its applications in disaster relief", in "Sentiment Analysis and Ontology Engineering", pp. 313–340 (Springer, 2016).

Bessi, A. and E. Ferrara, "Social bots distort the 2016 u.s. presidential election online discussion", First Monday **21**, 11, URL `https://uncommonculture.org/ojs/index.php/fm/article/view/7090` (2016).

Blei, D. M., A. Y. Ng and M. I. Jordan, "Latent dirichlet allocation", Journal of machine Learning research **3**, 01, 993–1022 (2003).

Bonn, G., "Update on UNV support to Typhoon Haiyan response efforts", URL `https://www.unv.org/news/update-unv-support-typhoon-haiyan-response-efforts` (2013).

Broniatowski, D. A., A. M. Jamison, S. H. Qi, L. AlKulaib, T. Chen, A. Benton, S. C. Quinn and M. Dredze, "Weaponized Health Communication: Twitter Bots and Russian Trolls Amplify the Vaccine Debate", American Journal of Public Health **108**, 10, 1378–1384, URL `https://ajph.aphapublications.org/doi/pdf/10.2105/AJPH.2018.304567` (2018).

Butler, D., "Crowdsourcing goes mainstream in typhoon response", Nature **10** (2013).

Cao, Q., M. Sirivianos, X. Yang and T. Pregueiro, "Aiding the detection of fake accounts in large scale social online services", in "Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation", p. 15 (USENIX Association, 2012).

Castillo, C., *Big Crisis Data* (Cambridge University Press, 2016).

Castillo, C., M. Mendoza and B. Poblete, "Information credibility on twitter", in "Proceedings of the WWW conference", pp. 675–684 (ACM, 2011).

Chavoshi, N., H. Hamooni and A. Mueen, "Identifying Correlated Bots in Twitter", Encyclopedia of Library and Information Sciences, Third Edition pp. 4814–4819, URL `http://www.crcnetbase.com/doi/10.1081/E-ELIS3-120043526` (2009).

Chavoshi, N., H. Hamooni and A. Mueen, "Debot: Twitter bot detection via warped correlation.", in "ICDM", pp. 817–822 (2016).

Chen, Z. and D. Subramanian, "An Unsupervised Approach to Detect Spam Campaigns that Use Botnets on Twitter", CoRR **abs/1804.0**, 1–7, URL `http://arxiv.org/abs/1804.05232` (2018).

Cheng, Z., J. Caverlee and K. Lee, "You are where you tweet: a content-based approach to geo-locating twitter users", in "Proceedings of the 19th ACM international conference on Information and knowledge management", pp. 759–768 (ACM, 2010).

Chino, D. Y. T., A. F. Costa, A. J. M. Traina and C. Faloutsos, "VOLTIME : Unsupervised Anomaly Detection on Users' Online Activity Volume", Proceedings of the 2017 SIAM International Conference on Data Mining **C**, June, 108–116, URL `https://epubs.siam.org/doi/10.1137/1.9781611974973.13` (2017).

Chu, Z., S. Gianvecchio, H. Wang and S. Jajodia, "Who is tweeting on Twitter: human, bot, or cyborg?", in "Proceedings of the 26th annual computer security applications conference", pp. 21–30 (ACM, 2010).

Chu, Z., S. Gianvecchio, H. Wang and S. Jajodia, "Detecting automation of Twitter accounts: Are you a human, bot, or cyborg?", IEEE Transactions on Dependable and Secure Computing **9**, 6, 811–824 (2012).

Clark, D. B., "The Bot Bubble", URL `https://newrepublic.com/article/121551/bot-bubble-click-farms-have-inflated-social-media-currency` (2015).

Cohn, M. A., M. R. Mehl and J. W. Pennebaker, "Linguistic markers of psychological change surrounding september 11, 2001", Psychological science **15**, 10, 687–693 (2004).

Cook, D. M., B. Waugh, M. Abdipanah, O. Hashemi and S. Abdul Rahman, "Twitter Deception and Influence: Issues of Identity, Slacktivism, and Puppetry", Journal of Information Warfare **13**, 1, 58–71 (2014).

Cresci, S., R. Di Pietro, M. Petrocchi, A. Spognardi and M. Tesconi, "DNA-Inspired Online Behavioral Modeling and Its Application to Spambot Detection", IEEE Intelligent Systems **31**, 5, 58–64 (2016).

Cresci, S., R. Di Pietro, M. Petrocchi, A. Spognardi and M. Tesconi, "The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race", in "Proceedings of the 26th International Conference on World Wide Web Companion", pp. 963–972 (International World Wide Web Conferences Steering Committee, 2017).

Davis, C. A., O. Varol, E. Ferrara, A. Flammini and F. Menczer, "Botornot: A system to evaluate social bots", in "Proceedings of the 25th International Conference Companion on World Wide Web", pp. 273–274 (International World Wide Web Conferences Steering Committee, 2016).

DiFonzo, N. and P. Bordia, *Rumor psychology: Social and organizational approaches.* (American Psychological Association, 2007).

Elder, J., "Inside a Twitter Robot Factory; Fake Activity, Often Bought for Publicity Purposes, Influences Trending Topics", URL `https://www.wsj.com/articles/bogus-accounts-dog-twitter-1385335134` (2013).

Ellis, E., "How the USGS uses Twitter data to track earthquakes", \url{https://goo.gl/E6r0b2} (2015).

Faulkner, M., R. Clayton, T. Heaton, K. M. Chandy, M. Kohler, J. Bunn, R. Guy, A. Liu, M. Olson, M. Cheng *et al.*, "Community sense and response systems: Your phone as quake detector", Communications of the ACM **57**, 7, 66–75 (2014).

Faulkner, M., M. Olson, R. Chandy, J. Krause, K. M. Chandy and A. Krause, "The next big one: Detecting earthquakes and other rare events from community-based sensors", in "Information Processing in Sensor Networks (IPSN), 2011 10th International Conference on", pp. 13–24 (IEEE, 2011).

Fernandez, M., L. Alvarez and R. Nixon, "Still Waiting for FEMA in Texas and Florida After Hurricanes", URL `https://www.nytimes.com/2017/10/22/us/fema-texas-florida-delays-.html` (2017).

Ferrara, E., O. Varol, C. Davis, F. Menczer and A. Flammini, "The rise of social bots", Communications of the ACM **59**, 7, 96–104 (2016).

Freitas, C., F. Benevenuto, S. Ghosh and A. Veloso, "Reverse engineering socialbot infiltration strategies in twitter", in "Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015", pp. 25–32 (ACM, 2015).

Freund, Y. and R. E. Schapire, "A decision theoretic generalization of on-line learning and an application to boosting", Journal of Computer and System Sciences **55**, 1, 119–139 (1997).

Goel, V. and S. Ember, "As Paris Terror Attacks Unfolded, Social Media Tools Offered Help in Crisis", URL `http://www.nytimes.com/2015/11/15/technology/as-paris-terror-attacks-unfolded-social-media-tools-offered-help-in-crisis.html` (2015).

Graham, M. and M. Zook, "Visualizing global cyberscapes: Mapping user-generated placemarks", Journal of Urban Technology **18**, 1, 115–132 (2011).

Grenoble, R., "Hurricane Harvey Is Just The Latest In Facebook's Fake News Problem", URL `https://www.huffingtonpost.com/entry/facebook-hurricane-harvey-fake-news{\_}us{\_}59b17900e4b0354e441021fb` (2017).

Grier, C., K. Thomas, V. Paxson and M. Zhang, "@spam: the underground on 140 characters or less", in "Conference on Computer and Communications Security", pp. 27–37 (ACM, 2010).

Gupta, A., H. Lamba and P. Kumaraguru, "$1.00 per RT #BostonMarathon #PrayForBoston: Analyzing Fake Content on Twitter", in "eCrime Researchers Summit (eCRS)", pp. 1–12 (IEEE, 2013).

Han, B., P. Cook and T. Baldwin, "A stacking-based approach to twitter user geolocation prediction.", in "ACL (Conference System Demonstrations)", pp. 7–12 (2013).

Harris, R., D. G. J.X. and D. Debelius, "The Twitter Purge: How Many Followers Trump, Nicki Minaj and Others Lost", URL https://www.nytimes.com/interactive/2018/07/13/technology/twitter-purge-fake-followers.html (2018).

Hasher, L., D. Goldstein and T. Toppino, "Frequency and the conference of referential validity", Journal of verbal learning and verbal behavior **16**, 1, 107–112 (1977).

Houston, J. B., J. Hawthorne, M. F. Perreault, E. H. Park, M. Goldstein Hode, M. R. Halliwell, S. E. Turner McGowen, R. Davis, S. Vaid, J. A. McElderry *et al.*, "Social media and disasters: a functional framework for social media use in disaster planning, response, and research", Disasters **39**, 1, 1–22 (2015).

Imran, M., C. Castillo, J. Lucas, P. Meier and S. Vieweg, "AIDR: Artificial intelligence for disaster response", in "Proceedings of the 23rd International Conference on World Wide Web", pp. 159–162 (ACM, 2014).

Imran, M., S. M. Elbassuoni, C. Castillo, F. Diaz and P. Meier, "Extracting information nuggets from disaster-related messages in social media", Proc. of ISCRAM, Baden-Baden, Germany (2013).

Jane, P. H., "Click and elect: how fake news helped Donald Trump win a real election", URL https://www.theguardian.com/commentisfree/2016/nov/14/fake-news-donald-trump-election-alt-right-social-media-tech-companies (2016).

Jin, Z., J. Cao, Y. Zhang and J. Luo, "News Verification by Exploiting Conflicting Social Viewpoints in Microblogs.", in "AAAI", pp. 2972–2978 (2016).

John, J. P., A. Moshchuk, S. D. Gribble and A. Krishnamurthy, "Studying Spamming Botnets Using Botlab", in "Networked Systems Design and Implementation", vol. 9, pp. 291–306 (2009).

Joshi, S., "Twitter, Facebook and Google Activate Features to Help People Affected by Chennai Floods", URL https://mashable.com/2015/12/03/facebook-security-check-twitter-google-chennai/#m7OlZHJdWZq1 (2015).

Jurgens, D., "That's what friends are for: Inferring location in online social media platforms based on social relationships.", ICWSM **13**, 273–282 (2013).

Kanich, C., C. Kreibich, K. Levchenko, B. Enright, G. M. Voelker, V. Paxson and S. Savage, "Spamalytics: An empirical analysis of spam marketing conversion", in "Conference on Computer and Communications Security", pp. 3–14 (ACM, 2008).

Kapferer, J.-N., *Rumors: Uses, interpretations, and images* (Transaction Publishers, 2013).

Kelion, L. and S. Silva, "Pro-Clinton bots 'fought back but outnumbered in second debate'", URL http://www.bbc.com/news/technology-37703565 (2016).

Khaund, T., S. Al-Khateeb, S. Tokdemir and N. Agarwal, "Analyzing Social Bots and Their Coordination During Natural Disasters", in "International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation", pp. 207–212 (Springer, 2018).

Kitzie, V. L., A. Karami and E. Mohammadi, "Life Never Matters in the Democrats Mind Examining Strategies of Retweeted Social Bots During a Mass Shooting Event", arXiv:1808.09325 **nan**, nan, nan, URL `http://arxiv.org/pdf/1808.09325v1` (2018).

Koh, Y., "Only 11% of New Twitter Users in 2012 Are Still Tweeting", URL `https://blogs.wsj.com/digits/2014/03/21/new-report-spotlights-twitters-retention-problem/` (2014).

Kossinets, G. and D. J. Watts, "Empirical analysis of an evolving social network", science **311**, 5757, 88–90 (2006).

Kou, Y., X. Gui, Y. Chen and K. Pine, "Conspiracy talk on social media: collective sensemaking during a public health crisis", Proceedings of the ACM on Human-Computer Interaction **1**, CSCW, 61 (2017).

Kudugunta, S. and E. Ferrara, "Deep neural networks for bot detection", Information Sciences **467**, 312–322 (2018).

Kumar, S., G. Barbier, M. A. Abbasi and H. Liu, "TweetTracker: An Analysis Tool for Humanitarian and Disaster Relief.", in "ICWSM", (2011).

Kumar, S., X. Hu and H. Liu, "A behavior analytics approach to identifying tweets from crisis regions", in "Proceedings of the 25th ACM conference on Hypertext and social media", pp. 255–260 (ACM, 2014a).

Kumar, S., F. Morstatter and H. Liu, *Twitter Data Analytics* (Springer, 2014b).

Lee, K., J. Caverlee and S. Webb, "Uncovering social spammers: social honeypots+ machine learning", in "Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval", pp. 435–442 (ACM, 2010).

Lee, K., B. D. Eoff and J. Caverlee, "Seven Months with the Devils: A Long-Term Study of Content Polluters on Twitter", in "ICWSM", pp. 185–192 (AAAI, 2011), URL `https://www.aaai.org/ocs/index.php/ICWSM/ICWSM11/paper/view/2780`.

Lee, S. and J. Kim, "Early filtering of ephemeral malicious accounts on Twitter", Computer Communications **54**, 48–57 (2014).

Li, R., S. Wang, H. Deng, R. Wang and K. C.-C. Chang, "Towards social user profiling: unified and discriminative influence model for inferring home locations", in "Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining", pp. 1023–1031 (ACM, 2012).

Lynch, M. P., "Fake News and the Internet Shell Game", URL https://www.nytimes.com/2016/11/28/opinion/fake-news-and-the-internet-shell-game.html (2016).

Mejova, Y., I. Weber and M. W. Macy, *Twitter: A Digital Socioscope* (Cambridge University Press, 2015).

Minnich, A., N. Chavoshi, D. Koutra and A. Mueen, "BotWalk : Efficient Adaptive Exploration of Twitter Bot Networks", 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining BotWalk: pp. 467–474 (2017).

Mishne, G., N. S. Glance *et al.*, "Predicting movie sales from blogger sentiment.", in "AAAI spring symposium: computational approaches to analyzing weblogs", pp. 155–158 (2006).

Mitra, T. and E. Gilbert, "Have You Heard?: How Gossip Flows Through Workplace Email.", in "ICWSM", (2012).

Morstatter, F., N. Lubold, H. Pon-Barry, J. Pfeffer and H. Liu, "Finding eyewitness tweets during crises", in "ACL 2014 Workshop on Language Technologies and Computational Social Science", (2014).

Morstatter, F., L. Wu, T. H. Nazer, K. M. Carley and H. Liu, "A new approach to bot detection: striking the balance between precision and recall", in "Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining", pp. 533–540 (IEEE Press, 2016).

Nied, A. C., L. Stewart, E. Spiro and K. Starbird, "Alternative narratives of crisis events: Communities and social botnets engaged on social media", in "Companion of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing", pp. 263–266 (ACM, 2017).

Okolloh, O., "Ushahidi, or â testimonyâ : Web 2.0 tools for crowdsourcing crisis information", Participatory learning and action **59**, 1, 65–70 (2009).

Palen, L. and K. M. Anderson, "Crisis informatics—new data for extraordinary times", Science **353**, 6296, 224–225, URL http://science.sciencemag.org/content/353/6296/224 (2016).

Palen, L. and S. B. Liu, "Citizen communications in crisis: anticipating a future of ICT-supported public participation", in "Proceedings of the SIGCHI conference on Human factors in computing systems", pp. 727–736 (ACM, 2007).

Palen, L. and S. Vieweg, "The emergence of online widescale interaction in unexpected events: assistance, alliance & retreat", in "Proceedings of the CSCW Conference", pp. 117–126 (ACM, 2008).

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and E. Duchesnay, "Scikit-learn: Machine Learning in {P}ython", Journal of Machine Learning Research **12**, 2825–2830 (2011).

Pennycook, G. and D. Rand, "Why Do People Fall for Fake News?", (2019).

Popescu, A.-M. and M. Pennacchiotti, "Detecting controversial events from twitter", in "Proceedings of the 19th ACM international conference on Information and knowledge management", pp. 1873–1876 (ACM, 2010).

Popescu, A.-M., M. Pennacchiotti and D. Paranjpe, "Extracting events and event descriptions from twitter", in "Proceedings of the 20th international conference companion on World wide web", pp. 105–106 (ACM, 2011).

Powell, J. W., "An introduction to the natural history of disaster", Univ. of Maryland: Disaster Research Project (1954).

Power, R., B. Robinson, J. Colton and M. Cameron, "Emergency situation awareness: Twitter case studies", in "International Conference on Information Systems for Crisis Response and Management in Mediterranean Countries", pp. 218–231 (Springer, 2014).

Proctor, B., "Update #1 on Nepal earthquake deployment", URL `http://www.standbytaskforce.org/2015/04/25/update-1-on-nepal-earthquake-deployment/` (2015).

Proctor, B., "Standby Task Force is deploying for Hurricane Maria response efforts", URL `http://www.standbytaskforce.org/2017/09/10/we-are-deploying-in-support-of-the-hurricane-irma-response/` (2017).

Proctor, B. and D. Dalchand, "Standby Task Force is deploying for Hurricane Maria response efforts", URL `http://www.standbytaskforce.org/2017/10/06/standby-task-force-is-supporting-fema-in-response-to-hurricane-maria/` (2017).

Purohit, H., C. Castillo, F. Diaz, A. Sheth and P. Meier, "Emergency-relief coordination on social media: Automatically matching resource requests and offers", First Monday **19**, 1 (2013).

Ratkiewicz, J., M. Conover, M. Meiss, B. Gonçalves, S. Patil, A. Flammini and F. Menczer, "Truthy: mapping the spread of astroturf in microblog streams", in "World Wide Web Companion", pp. 249–252 (ACM, 2011a).

Ratkiewicz, J., M. Conover, M. R. Meiss, B. Gonçalves, A. Flammini and F. Menczer, "Detecting and Tracking Political Abuse in Social Media", in "ICWSM", vol. 11, pp. 297–304 (AAAI, 2011b).

Reese, A., "How we'll predict the next natural disaster: Advances in natural hazard forecasting could help keep more people out of harm's way", Discover Magazine (Sep. 2016).

Rosnow, R. L. and G. A. Fine, *Rumor and gossip: The social psychology of hearsay.* (Elsevier, 1976).

Sakaki, T., M. Okazaki and Y. Matsuo, "Earthquake shakes twitter users: real-time event detection by social sensors", in "Proceedings of the 19th international conference on World wide web", pp. 851–860 (ACM, 2010).

SAMBULI, N., "How useful is a tweet? a review of the first tweets from the westgate mall attack", https://goo.gl/qRGYZD, accessed 10 Feb 2017 (2013).

Sampson, J., F. Morstatter, R. Zafarani and H. Liu, "Real-time crisis mapping using language distribution", in "2015 IEEE International Conference on Data Mining Workshop (ICDMW)", pp. 1648–1651 (IEEE, 2015).

Schulz, A., A. Hadjakos, H. Paulheim, J. Nachtwey and M. Mühlhäuser, "A multi-indicator approach for geolocalization of tweets", in "ICWSM", pp. 573–582 (2013).

Shao, C., G. L. Ciampaglia, O. Varol, K. Yang, A. Flammini and F. Menczer, "The spread of low-credibility content by social bots", Nature Communications , 2018, URL http://arxiv.org/abs/1707.07592 (2017).

Shao, C., G. L. Ciampaglia, O. Varol, K. C. Yang, A. Flammini and F. Menczer, "The spread of fake news by social bots", Nature communications **9**, 1, 4787 (2018a).

Shao, C., P. M. Hui, L. Wang, X. Jiang, A. Flammini, F. Menczer and G. L. Ciampaglia, "Anatomy of an online misinformation network", PLoS ONE **13**, 4, 1–23 (2018b).

Shearer, E., "Candidates' Social Media Outpaces Their Websites and Emails as an Online Campaign News Source", URL http://www.pewresearch.org/fact-tank/2016/07/20/candidates-social-media-outpaces-their-websites-and-emails-as-an-online-campaign-news-source/ (2016).

Shu, K., D. Mahudeswaran, S. Wang, D. Lee and H. Liu, "FakeNewsNet: A Data Repository with News Content, Social Context and Dynamic Information for Studying Fake News on Social Media", arXiv preprint arXiv:1809.01286 (2018).

Shu, K., A. Sliva, S. Wang, J. Tang and H. Liu, "Fake News Detection on Social Media: A Data Mining Perspective", KDD exploration newsletter (2017).

Silverman, C., "This Analysis Shows How Viral Fake Election News Stories Outperformed Real News On Facebook", URL https://www.buzzfeednews.com/article/craigsilverman/viral-fake-election-news-outperformed-real-news-on-facebook (2016).

Sinha, S., "3 Ways Nepalis Are Using Crowdsourcing to Aid in Quake Relief", URL https://www.nytimes.com/2015/05/02/world/asia/3-ways-nepalis-are-using-crowdsourcing-to-aid-in-quake-relief.html (2015).

Sorensen, J. H., "Hazard warning systems: Review of 20 years of progress", Natural Hazards Review **1**, 2, 119–125 (2000).

Starbird, K., "Examining the Alternative Media Ecosystem through the Production of Alternative Narratives of Mass Shooting Events on Twitter", Icwsm (2017) , Icwsm, 230–239, URL https://faculty.washington.edu/kstarbi/Alt{_}Narratives{_}ICWSM17-CameraReady.pdf{\%}0Ahttp://faculty.washington.edu/kstarbi/Alt{_}Narratives{_}ICWSM17-CameraReady.pdf{\%}0Ahttps://aaai.org/ocs/index.php/ICWSM/ICWSM17/paper/view/15603 (2017).

Stella, M., E. Ferrara and M. De Domenico, "Bots sustain and inflate striking opposition in online social systems", pp. 1–10, URL http://arxiv.org/abs/1802.07292 (2018).

Subrahmanian, V. S., A. Azaria, S. Durst, V. Kagan, A. Galstyan, K. Lerman, L. Zhu, E. Ferrara, A. Flammini, F. Menczer and Others, "The DARPA Twitter bot challenge", arXiv preprint arXiv:1601.05140 (2016).

Sutton, J., E. S. Spiro, S. Fitzhugh, B. Johnson, B. Gibson and C. T. Butts, "Terse message amplification in the Boston bombing response", Proceedings of the ISCRAM Conference pp. 612–621 (2014).

Szabo, G. and B. A. Huberman, "Predicting the popularity of online content", Communications of the ACM **53**, 8, 80–88 (2010).

Tausczik, Y. R. and J. W. Pennebaker, "The psychological meaning of words: LIWC and computerized text analysis methods", Journal of language and social psychology **29**, 1, 24–54 (2010).

Thomas, K., C. Grier and V. Paxson, "Adapting social spam infrastructure for political censorship", in "Conference on Large-Scale Exploits and Emergent Threats", (USENIX, 2012).

Thomas, K., C. Grier, D. Song and V. Paxson, "Suspended accounts in retrospect: an analysis of twitter spam", in "Internet Measurement Conference", pp. 243–258 (ACM, 2011).

Thonnard, O. and M. Dacier, "A strategic analysis of spam botnets operations", in "Collaboration, Electronic Messaging, Anti-Abuse and Spam Conference", pp. 162–171 (ACM, 2011).

Timberg, C. and E. Dwoskin, "Twitter is Sweeping out Fake Accounts Like Never Before, Putting User Growth at Risk", URL https://www.washingtonpost.com/technology/2018/07/06/twitter-is-sweeping-out-fake-accounts-like-never-before-putting-user-growth-risk/ (2018).

Tumasjan, A., T. O. Sprenger, P. G. Sandner and I. M. Welpe, "Predicting elections with twitter: What 140 characters reveal about political sentiment.", ICWSM **10**, 178–185 (2010).

Van Dongen, S., "Graph clustering via a discrete uncoupling process", SIAM Journal on Matrix Analysis and Applications **30**, 1, 121–141 (2008).

Varol, O., E. Ferrara, C. A. Davis, F. Menczer and A. Flammini, "Online human-bot interactions: Detection, estimation, and characterization", in "ICWSM", pp. 280–289 (2017).

Varol, O., E. Ferrara, C. A. Davis, F. Menczer, A. Flammini, V. S. Subrahmanian, A. Azaria, S. Durst, V. Kagan, A. Galstyan and Others, "Online Human-Bot Interactions: Detection, Estimation, and Characterization", Comm. ACM **59**, 7 (2016).

Vieweg, S., A. L. Hughes, K. Starbird and L. Palen, "Microblogging during two natural hazards events: what twitter may contribute to situational awareness", in "Proceedings of the SIGCHI Conference", pp. 1079–1088 (ACM, 2010).

Vieweg, S. E., *Situational awareness in mass emergency: A behavioral and linguistic analysis of microblogged communications*, Ph.D. thesis, University of Colorado at Boulder (2012).

Vosoughi, S., D. Roy and S. Aral, "The spread of true and false news online", Science **359**, 6380, 1146–1151 (2018).

Waddington, K., *Gossip and organizations* (Routledge, 2012).

Wang, A. H., "Detecting spam bots in online social networking sites: a machine learning approach", in "Data and Applications Security and Privacy XXIV", pp. 335–342 (Springer, 2010).

Wang, G., M. Mohanlal, C. Wilson, X. Wang, M. Metzger, H. Zheng and B. Y. Zhao, "Social Turing Tests: Crowdsourcing Sybil Detection", arXiv preprint arXiv:1205.3856 (2013).

Wang, X., M. S. Gerber and D. E. Brown, "Automatic crime prediction using events extracted from twitter posts", in "International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction", pp. 231–238 (Springer, 2012).

Webb, S., J. Caverlee and C. Pu, "Social honeypots: Making friends with a spammer near you.", in "CEAS", pp. 1–10 (2008).

Weenig, M. W. H., A. C. W. J. Groenenboom and H. A. M. Wilke, "Bad news transmission as a function of the definitiveness of consequences and the relationship between communicator and recipient.", Journal of personality and social psychology **80**, 3, 449 (2001).

Wei, W., K. Joseph, H. Liu and K. M. Carley, "The fragility of Twitter social networks against suspended users", in "IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining", pp. 9–16 (IEEE, 2015).

Weir, K., "Why We Believe Alternative Facts", Monitor on Psychology **48**, 5, 34–39, URL `https://www.apa.org/monitor/2017/05/alternative-facts` (2017).

White, J. I., L. Palen and K. M. Anderson, "Digital mobilization in disaster response: the work & self-organization of on-line pet advocates in response to hurricane sandy", in "Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing", pp. 866–876 (ACM, 2014).

Wolpert, D. H., "Stacked generalization", Neural networks **5**, 2, 241–259 (1992).

Wu, L., X. Hu, F. Morstatter and H. Liu, "Detecting Camouflaged Content Polluters.", in "ICWSM", pp. 696–699 (2017).

Xie, Y., F. Yu, K. Achan, R. Panigrahy, G. Hulten and I. Osipkov, "Spamming botnets: signatures and characteristics", ACM SIGCOMM Computer Communication Review **38**, 4, 171–182 (2008).

Yates, D. and S. Paquette, "Emergency knowledge management and social media technologies: A case study of the 2010 Haitian earthquake", in "Proceedings of the 73rd ASIS&T Annual Meeting on Navigating Streams in an Information Ecosystem-Volume 47", p. 42 (American Society for Information Science, 2010).

Yu, S. and S. Kak, "A survey of prediction using social media", arXiv preprint arXiv:1203.1647 (2012).

Zafarani, R., M. A. Abbasi and H. Liu, *Social Media Mining: An Introduction* (Cambridge University Press, 2014).

Zafarani, R. and H. Liu, "10 Bits of Surprise: Detecting Malicious Users with Minimum Information", in "Conference on Information and Knowledge Management", pp. 423–431 (ACM, 2015).

Zeimpekis, V., S. Ichoua and I. Inis, "Humanitarian and relief logistics", Springer **60** (2013).

Zhang, C. M. and V. Paxson, "Detecting and Analyzing Automated Activity on Twitter. In N. Spring and G. Riley (Eds.)", Passive and Active Measurement. PAM 2011 **LNCS 6579**, 102–111 (2011).

Zhang, W. and S. Skiena, "Improving movie gross prediction through news analysis", in "Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology-Volume 01", pp. 301–304 (IEEE Computer Society, 2009).

Zook, M., M. Graham, T. Shelton and S. Gorman, "Volunteered geographic information and crowdsourcing disaster relief: a case study of the Haitian earthquake", World Medical & Health Policy **2**, 2, 7–33 (2010).

APPENDIX A

TWEETTRACKER

TweetTracker (Kumar *et al.*, 2011) is a system for tracking, analyzing, and understanding tweets related to a specific topic:

- Tracking: to track the status of an event, data can be collected using a set of criteria including keywords, location, and users. The source of the data can be chosen from Twitter, Facebook, YouTube, VK, and Instagram. See Figure A.1.

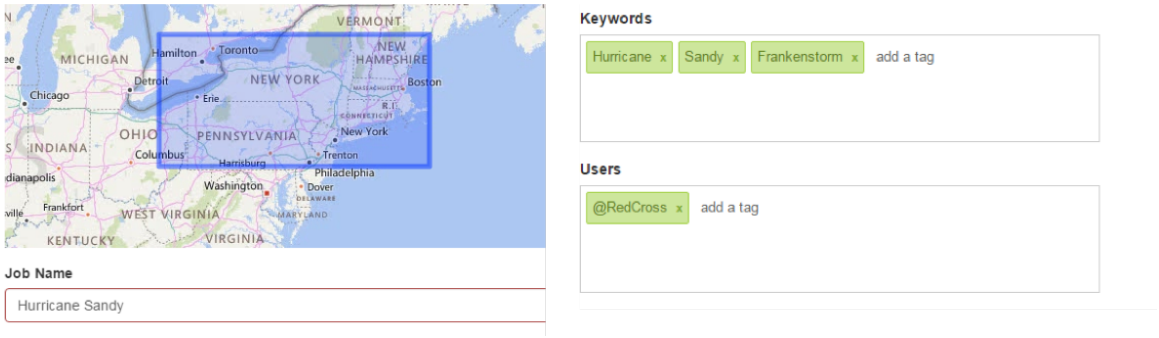


Figure A.1: Tracking Hurricane Sandy, 2012, on TweetTracker.

- Analyzing: changes in the total number of post or frequency of posts with specific words can be plotted for different time periods. Moreover, keywords, hashtags, links, images, and videos with their frequencies are available to the user. See Figure A.2.

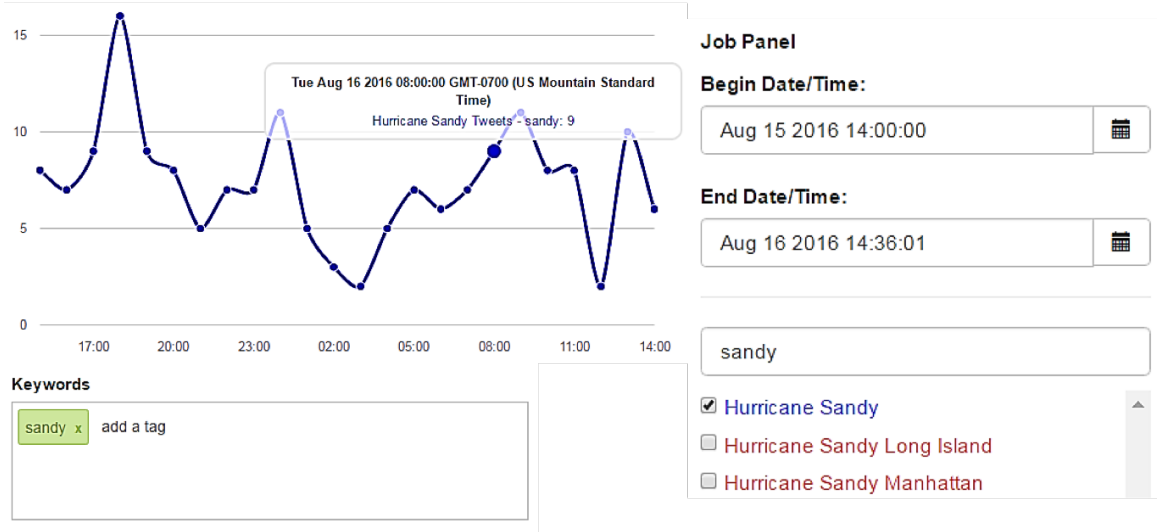


Figure A.2: Analyzing Hurricane Sandy Dataset on TweetTracker.

- Understanding: to better understand the geographic distribution of posts on the globe, the posts which are geotagged will be shown on a map. See Figure A.3.

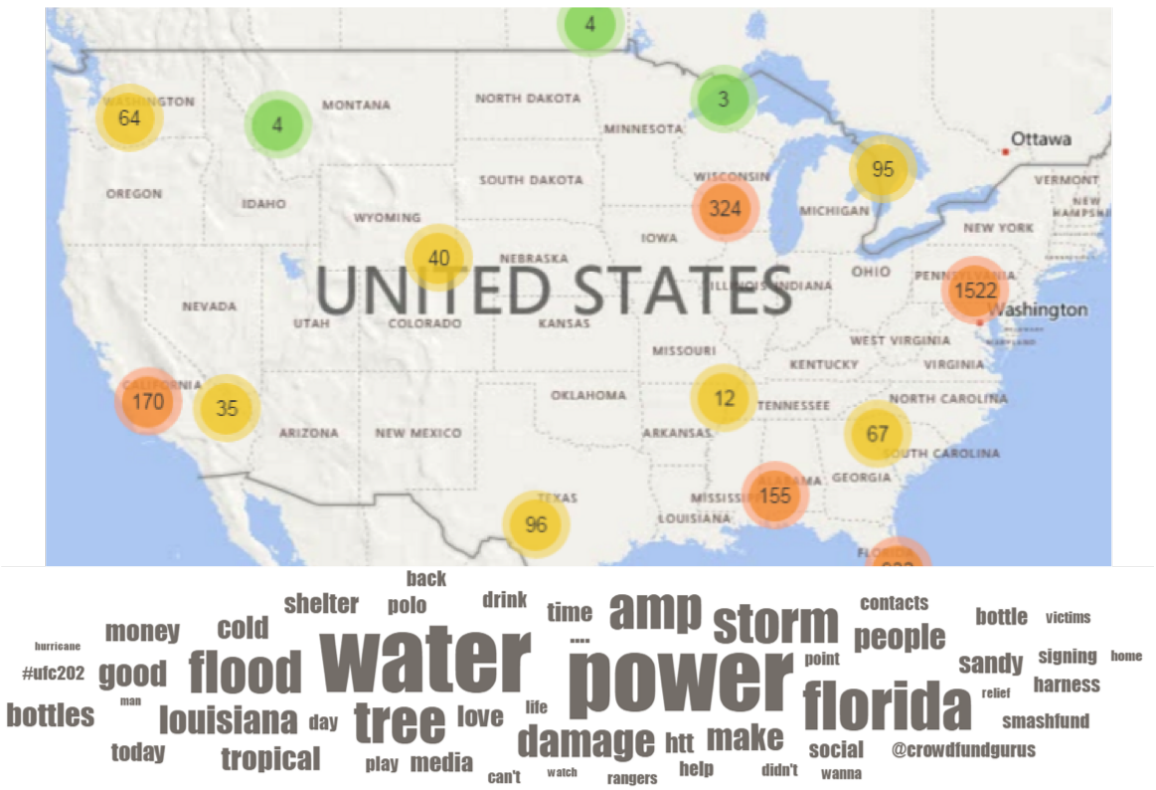


Figure A.3: Understanding the Discussion About Hurricane Sandy Using TweetTracker.

APPENDIX B

HONEYPOT METHOD FOR GROUND TRUTH COLLECTION

Social honeypots for tracking malicious actors on social media were introduced in 2008 by Webb, Caverlee, and Pu (Webb *et al.*, 2008). They constructed 51 honeypot profiles and associated them with distinct geographic locations in MySpace which was a popular social networking community at the time. They collected 1,570 spamming profiles during a four-month evaluation period. The characteristics of these honeypots are as follows:

- To capture the geographic artifacts of spamming behaviors, each honeypot account was associated with the most populated city in a US state.
- Except for their geographical locations, all honeypots share identical profile features: same relationship status (single), body type (athletic), and ethnicity (White / Caucasian).
- They passively wait for malicious users to send them a friend request, collect the profile information of the spammer, and reject the request to avoid being associated with malicious actors and consequently being banned from the network.

In a honeypot method, all the users who interact with a honeypot are considered as bots/spammers/content polluters. Off course any user who follows or messages a honeypot account is not necessarily a bot. However, honeypots post random content and do not engage in any interactions like normal users would. Hence, there is no point for legitimate users to interact with honeypots and this makes the assumption of honeypots reasonable. Lee (Lee *et al.*, 2011) that we can safely assume that the probability of a users mistakenly interacting with a honeypot is similar, if not less that, the probability of making a mistake in a manual annotation of social media users. So, the honeypot method is expected to be as trustworthy as a ground-truth collection method based on manual annotation.

Using honeypots for understanding bots on social media has multiple advantages (Lee *et al.*, 2011): (1) we can automatically collect data on these malicious actors without manual annotation by human experts; (2) the data collection process and monitoring can be performed without inference in the normal activities of legitimate users; (3) as bots change their strategies, honeypots can be modified accordingly to capture and monitor them. Having access to the profiles of a large number of malicious users and monitoring their activities over time yields to interesting findings such as:

- Malicious actors were most active the day before, the day of, and the day after Columbus Day, Halloween, and Thanksgiving. Authors hypothesize that legitimate users are expected to be highly active on these days and provide a larger audience for spammers to lure.
- Users in the Midwest states received more spamming friend request. One explanation is that these states started using MySpace later that users in the Western states and are less "MySpace-savvy" and hence more prone to be manipulated by the malicious actors.
- Malicious actors send large number of friend requests in short periods of time (bursty activity) and target users in diverse geographic locations.

Webb (Webb *et al.*, 2008) found five categories of malicious actors among the spammers that were collected by their honeypots:

1. Click Traps have deceptive profiles and direct MySpace users to nefarious websites.

2. Friend Infiltrators try to connect as many users as possible. Once the friendship connection is established, they use available communications channels to spam the target users.

3. Pornographic Storytellers have and "About me" section that consists of randomized pornographic stories, usually with links leading to pornographic web pages.

4. Japanese Pill Pushers promote sales of pills with intriguing examples and pictures of previous customers.

5. Winnies are a set of profiles whose bios all start with "Hey its winnie." and direct users to pornographic web pages.

In another attempt on using honeypots to study malicious actors on social media, Lee, Eoff, and Caverlee (Lee *et al.*, 2011) used honeypots to collect and understand content polluters on Twitter. They deployed 60 Twitter honeypots that were active for 7 months and collected 23,869 polluters. These honeypots do not interact with legitimate users. They use the mention mechanism, @username, between themselves and tweets regularly to remain active in the network. They publish four types of tweets : (1) a normal textual tweet; (2) an "@" reply to one of the other social honeypots; (3) a tweet containing a link; (4) a tweet containing one of Twitter's current Top 10 trending topics, which are popular n-grams. Some of the findings of this study is described bellow:

- The honeypot method can capture twitter bots much earlier that the official Twitter suspension mechanism. Lee et al. observed that 77% of bots were never suspended by Twitter and among the ones that were, some lived for 204 days before being banned.

- Malicious users had, on average, higher number of followers and followings. They followed 2,123 accounts, and the average number of followers they had was 2,163. These numbers are higher than most legitimate users which only have between 100 and 1,000 followers and following counts.

- Bots post an average of four tweets per day to mimic the activity patterns of legitimate users and avoid the Twitter's suspension mechanism.

- Content polluters have more fluctuations in their network. One possible explanation is that bots try to maintain balance between the number peoples they follow and the number of users who follow them back. Hence, they might unfollow the users who did not follow them to add other users to their network.

Similar to the study on MySpace (Webb *et al.*, 2008), Lee analyzed the bots they collected and categorized them into four groups:

1. Duplicate Spammers post identical tweets.

2. Duplicate @ Spammers target random legitimate users and send identical tweets to them, similar to Duplicate Spammers.

3. Malicious Promoters advertise online business, marketing, finance and so on.

4. Friend Infiltrators try to gain many followers by following a large number of users and hoping they reciprocate. When they reach a large audience, they engage in spamming activities.

Both of the studies mentioned so far (Webb *et al.*, 2008; Lee *et al.*, 2011) create general purpose honeypots and then analyze the lured malicious actors to find cluster structure in them. Morstatter et al. (Morstatter *et al.*, 2016), however, exploited the honeypot approach in a new way. The idea is that if we control the content and behavior of our honeypots, we can lure a specific group of bots in the wild. Specifically they aim to collect bots that spread the content aligned with the ideas of Arab Extremists on Twitter.

To collect this dataset, they constructed a honeypot network. This network consists of 9 accounts controlled automatically by a single controller. Each account tweets messages containing Arabic phrases identified by a subject matter expert pertaining to a specific group of people. Each account also randomly follows other honeypots in our network. Since bots have a lower chance of forming social ties (Thomas *et al.*, 2011), they performed this random following process to lower the chance that our accounts are deleted by Twitter's automatic account removal algorithm. Additionally, each honeypot can randomly retweet one of the other honeypots it follows in order to give that honeypot prominence on the network and lower its probability of being deleted due to Twitter's policies.

All of the bots in this honeypot dataset behave identically, as dictated by the controller. Each bot collects tweets using the selected Arabic phrases from Twitter's Streaming API. At random time intervals, the bot either copies the tweet, passing it off as its own, or retweets it from the user who originally posted it. The full logic for the honeypot controller is shown in Algorithm 3. Using a network of 9 honeypots, they considered all followers of the honeypots to be bots: due to the way these honeypots behave, no normal user would follow them (Lee *et al.*, 2011). With this approach they collected 3,602 active bot accounts who followed one of the honeypot accounts.

```
while True do
    Randomly choose one honeypot, h;
    r ⟵ random number ∈ [1, 10];
    if r = 1 then
        h retweets the most recent tweet from another randomly-selected
         honeypot;
    end
    else
        Sample tweets from Twitter Streaming API for 20 seconds, filtering
         based on Arabic phrases, call sample set S;
        r ⟵ random number ∈ [1, 10];
        s ⟵ randomly-selected tweet from S;
        if r < 3 then
            h retweets s;
        end
        else
            h copies s and tweets it word-for-word;
        end
    end
    Wait 10 seconds;
end
```

Algorithm 3: Logic for Honeypot Controller. the Controller Manages the Honeypot Accounts Used to Collect Bots in the Wild. All Randomly-Generated Numbers Mentioned in the Pseudocode Are Generated Uniformly at Random.