Database for Storing and Analyzing Tweets Posted During Disasters

by

Debarshi Saha

B.Tech, West Bengal University of Technology, 2014

MBA, West Bengal University of Technology, 2016

A REPORT

submitted in partial fulfillment of the
requirements for the degree

MASTER OF SCIENCE

Department of Computer Science
College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2019

Approved by:

Major Professor
Dr. Doina Caragea

Copyright

Abstract

In the last few decades, we have witnessed many natural disasters that have shaken the nations across the world. Millions of people have lost their lives, cities have been destroyed, people have gone homeless, injured and their lives have been affected. Sometimes hours or even days after a disaster, people are still stuck in the disaster sites, powerless, homeless and without food, as the rescue teams do not always get information about people in need in a timely manner. Whenever there is a natural disaster like a hurricane or an earthquake, people start tweeting about it. Most of the tweets are posted by users who are in the disaster sites, and may contain information about victims of the disaster: where they are and what the problem is, in what areas the rescue teams should work or focus on, or if someone needs special help. Such information can be very useful for the response teams, which can leverage this information in the recovery or rescue process. However, rescue team are faced with an information overload problem, due to the large number of tweets they need to sift through. To help with this issue, computational approaches can be used to analyze and prioritize information that may be useful to the rescue teams.

In this project, we have crawled tweets related to natural disasters, and extracted useful information in CSV files. Then, we have designed and developed a database to store the tweets. The design of the database is such that it will help us to query and gain information about a natural disaster. We have also performed some statistical analysis, such as deriving word clouds of the tweets posted during natural disasters. The analysis shows the areas where the users who post tweet about disaster are highly concerned. The word cloud analysis can help in comparing multiple natural disasters to understand patterns that are common or specific to disasters in terms of how Twitter users talk about them.

Table of Contents

List of Figures

List of Tables

Acknowledgments

I would like to thank my major professor, Dr. Doina Caragea for her expert advice and guidance throughout this project, without which this project would not have been possible. I would like to thank my committee members Dr. Cornelia Caragea and Dr. Dan Andresen for their valuable insights. I would also like to thank my parents as well as my friends for their constant encouragement throughout my grad life.

Dedication

This project is dedicated to all the people who have lost their lives in the Hurricane Irma, Hurricane Maria, Hurricane Harvey, California Wildfire, Mexico Earthquake, and to all whose lives have been affected by these natural calamities. Preventing a natural disaster may not be possible, but we can all focus on finding new ways to prevent loss of lives and minimize damage.

Chapter 1

Introduction

In the last few decades, we have witnessed many natural disasters that have shaken the nations across the world. (Zarin, 2017) Cities have been ruined, lost family members, homes destroyed, injuries abound, and lives changed forever.

There have been complaints that the rescue teams are not always able to reach out to the victims. Sometimes the rescue teams do not get information about people on time. This is mainly due to the lack of information on the disaster sites. Consequently, hours or even days after a disaster, people are still stuck in the disaster sites, powerless, homeless, agonized and without food.

Ironically, in this digital age, social media platforms may be the first to receive live, relevant information. Whenever, there is a natural disaster like a hurricane or an earthquake, people start tweeting about it. Most of the tweets are posted by the users who are in the disaster sites. We can gain valuable information from these tweets. (Dennis, 2016) Oftentimes, the users post about the victims: where they are, what the problem is, in what areas the rescue team should work or focus on, or if someone needs special help or any other information. This information can act as a good source for the rescue teams to focus on, and they can leverage this information in the recovery or rescue process. Unfortunately, rescue teams can also face an information overload problem. To effectively utilize social media information, they would need to sift through an unimaginable number of tweets. To help with this issue, computational approaches can be used to analyze and prioritize information that may be useful to the rescue teams.

In this project, we have crawled tweets related to natural disasters and extracted useful information in CSV files. Then, we have designed and developed a database which will house the tweets. The design of the database is such that it will help us to query and gain information about a natural disaster. We have also performed some statistical analysis. Namely, we have derived word clouds of the tweet from natural disasters. The analysis shows the areas where the users who post tweets about disasters are highly concerned. The word cloud analysis can help in comparing multiple natural disasters to understand patterns that are common or specific to disasters.

This project is just a basic implementation of a database for storing and analysis tweets useful to disaster management teams. Plenty of features can be added, which will not only help to gain valuable information, but also forms the basis of subsequent statistical and machine learning analyses. Analyzing disaster data will help to inform modern techniques in disaster management, build software and machines which aid in prediction, management, and recovery of disaster sites, as well as provide rapid aid to victims. Not only that, but it may also help to launch automated recovery in disaster areas using modern equipment and thus reach out to victims faster than ever. At the end of the day, it is all about minimizing damage and saving precious lives.

Chapter 2

Disaster Tweet Data

2.1 What Was Crawled?

Tweet crawling is the process of retrieving tweets and related information, posted by users, from the distributed Twitter servers across the world. Millions of tweets were crawled at Kansas State University during the time when the following natural disasters occurred: Hurricane Harvey, Hurricane Irma, Hurricane Maria, Mexico Earthquake, California Wildfire. The tweets were accumulated and stored in JSON files.

2.2 Keywords Used in Crawling

The words which are used to search for tweets of interest are called keywords. They form the basis for web crawling. If a tweet contains a specified keyword, it is shortlisted. Some examples of keywords which have been used in crawling the data which forms the basis of this project include: "Hurricane Harvey", "Hurricane Irma", "California Wildfires", "Mexico Earthquake", etc. Figure 2.1 shows an example of a tweet posted during Hurricane Harvey.

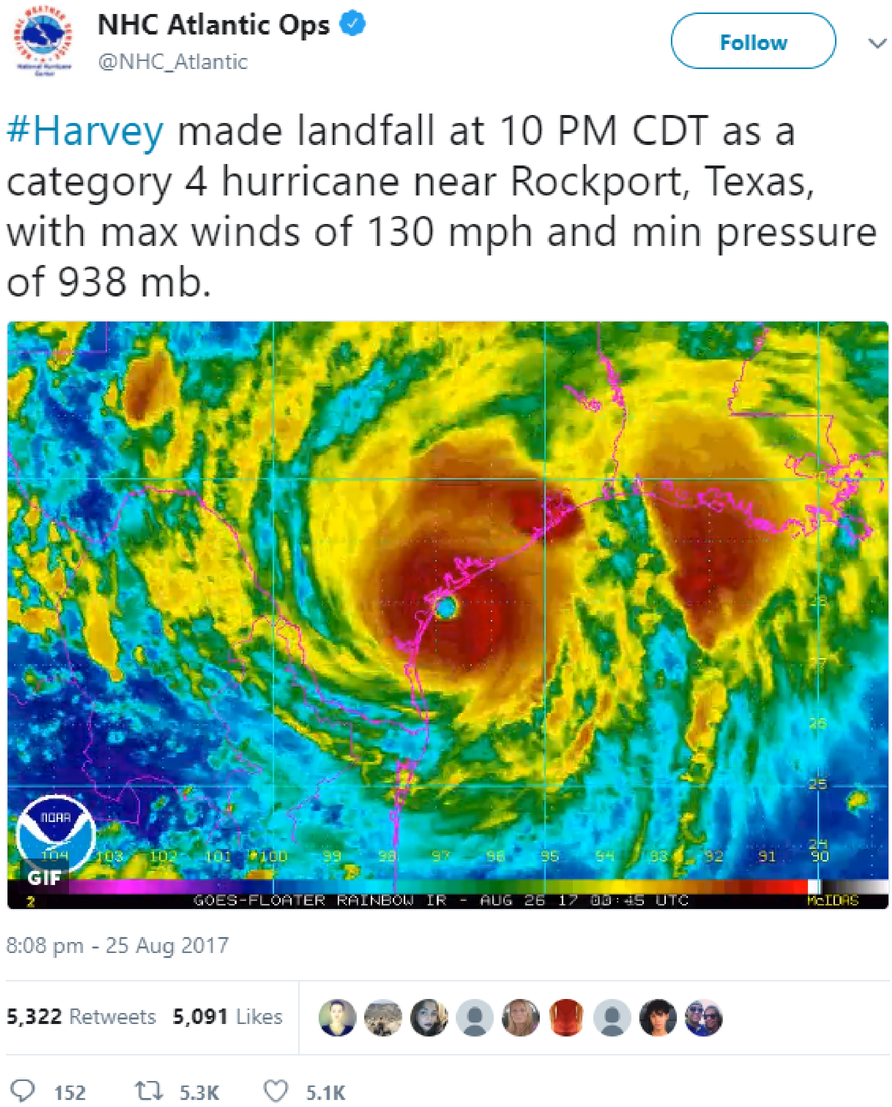

Figure 2.1: *Tweet posted by the user NHC Atlantic OPS during Hurricane Harvey*

2.3 Preprocessing of Data

2.3.1 Information Extraction from JSON files

The crawled data containing the tweets and relevant information was stored in JSON files. A JSON file contains many data fields. Only the fields which are important for subsequent intended analyses were extracted into specific columns. The data with the extracted columns was saved in CSV files, which will facilitate in loading the data into a database. The CSV files which were extracted from the JSON files mostly contains information about original tweets and retweets, and their corresponding users, time, location, media information, hashtags, etc. Figure 2.2 shows a fragment of a tweet's information stored in the JSON file, specifically information related to the tweet's user. Figure 2.3 shows a fragment of a CSV file, which contains useful information about tweets organized by columns.

```
{"created_at":"Thu Jan 11 02:13:28 +0000 2018",
  "id":951275839559172096,
  "id_str":"951275839559172096",
  "text":"RT @CNN: \"It looked like a World War I battlefield.\" A day after mudslides in Southern C
  "source":"\u003ca href=\"http:\/\/twitter.com\/download\/iphone\" rel=\"nofollow\"\u003eTwitter fo
  "truncated":false,"in_reply_to_status_id":null,
  "in_reply_to_status_id_str":null,
  "in_reply_to_user_id":null,
  "in_reply_to_user_id_str":null,"in_reply_to_screen_name":null,
  "user":{"id":25398687,
  "id_str":"25398687",
  "name":"Old Man Strick",
  "screen_name":"Pacino718",
  "location":null,
  "url":"http:\/\/www.vimeo.com\/blackpacino",
  "description":"Nobody. \u2620 http:\/\/www.magcloud.com\/user\/killergamz \u2620 http:\/\/www.thir
  "translator_type":"none",
  "protected":false,"verified":false,"followers_count":2644,
  "friends_count":1131,
  "listed_count":0,
  "favourites_count":53338,
  "statuses_count":170412,
  "created_at":"Thu Mar 19 22:27:32 +0000 2009",
  "utc_offset":-18000,"time_zone":"Quito","geo_enabled":false,"lang":"en",
  "contributors_enabled":false,"is_translator":false,
  "profile_background_color":"1A1B1F",
  "profile_background_image_url":"http:\/\/pbs.twimg.com\/profile_background_images\/807496053\/edc5
  "profile_background_image_url_https":"https:\/\/pbs.twimg.com\/profile_background_images\/80749605
  "profile_background_tile":true,
  "profile_link_color":"2FC2EF","profile_sidebar_border_color":"FFFFFF",
  "profile_sidebar_fill_color":"252429",
  "profile_text_color":"666666","profile_use_background_image":true,
  "profile_image_url":"http:\/\/pbs.twimg.com\/profile_images\/944935301025337344\/xWH2ykdb_normal.j
  "profile_image_url_https":"https:\/\/pbs.twimg.com\/profile_images\/944935301025337344\/xWH2ykdb_n
  "profile_banner_url":"https:\/\/pbs.twimg.com\/profile_banners\/25398687\/1513218665",
  "default_profile":false,"default_profile_image":false,"following":null,
  "follow_request_sent":null,"notifications":null},
  "geo":null,"coordinates":null,
  "place":null,"contributors":null,
  "retweeted_status":{"created_at":"Thu Jan 11 02:08:44 +0000 2018",
  "id":951274649375166466,"id_str":"951274649375166466",
  "text":"\"It looked like a World War I battlefield.\" A day after mudslides in Southern California
  "display_text_range":[0,140],"source":"\u003ca href=\"http:\/\/www.socialflow.com\" rel=\"nofollow
  "truncated":true,"in_reply_to_status_id":null,"in_reply_to_status_id_str":null,
  "in_reply_to_user_id":null,"in_reply_to_user_id_str":null,
  "in_reply_to_screen_name":null,
  "user":{"id":759251,"id_str":"759251","name":"CNN","screen_name":"CNN","location":null,"url":"http
  "description":"It\u2019s our job to #GoThere & tell the most difficult stories. Join us! For more
  "protected":false,"verified":true,"followers_count":39072702,"friends_count":1115,"listed_count":1
  "geo_enabled":true,"lang":"en","contributors_enabled":false,"is_translator":false,"profile_backgro
  "profile_background_image_url":"http:\/\/pbs.twimg.com\/profile_background_images\/515228058286952
  "profile_background_tile":false,"profile_link_color":"004287",
  "profile_sidebar_border_color":"000000","profile_sidebar_fill_color":"EEEEEE","profile_text_color"
  "profile_use_background_image":false,
  "profile_image_url":"http:\/\/pbs.twimg.com\/profile_images\/508960761826131968\/LnvhR8ED_normal.p
  "profile_image_url_https":"https:\/\/pbs.twimg.com\/profile_images\/508960761826131968\/LnvhR8ED_n
  "profile_banner_url":"https:\/\/pbs.twimg.com\/profile_banners\/759251\/1508752874",
  "default_profile":false,"default_profile_image":false,"following":null,
  "follow_request_sent":null,"notifications":null},"geo":null,"coordinates":null,"place":null,"contr
  "extended_tweet":{"full_text":"\"It looked like a World War I battlefield.\" A day after mudslides
  "display_text_range":[0,241],"entities":{"hashtags":[],"urls":[{"url":"https:\/\/t.co\/XCeL2cH12j"
  "user_mentions":[],"symbols":[],"media":[{"id":951274322877997056,"id_str":"951274322877997056",
  "indices":[242,265],"media_url":"http:\/\/pbs.twimg.com\/ext_tw_video_thumb\/951274322877997056\/p
  "media_url_https":"https:\/\/pbs.twimg.com\/ext_tw_video_thumb\/951274322877997056\/pu\/img\/bvVMc
  "url":"https:\/\/t.co\/k61afbYOyo","display_url":"pic.twitter.com\/k61afbYOyo","expanded_url":"htt
  "type":"video","sizes":{"thumb":{"w":150,"h":150,"resize":"crop"},"small":{"w":680,"h":680,"resize
  "url":"https:\/\/video.twimg.com\/ext_tw_video\/951274322877997056\/pu\/pl\/zusxO3iA054ePMu1.m3u8"
  "url":"https:\/\/video.twimg.com\/ext_tw_video\/951274322877997056\/pu\/vid\/240x240\/CafPRoAUtDok
  "url":"https:\/\/t.co\/k61afbYOyo","display_url":"pic.twitter.com\/k61afbYOyo","expanded_url":"htt
  "url":"https:\/\/video.twimg.com\/ext_tw_video\/951274322877997056\/pu\/vid\/480x480\/Nq_4zteFIe1G
}
```


Figure 2.2: *Example 1 of tweet information stored in a JSON File*

	source_file	created_a	retweeted	tweet_id	tweet_tex	user_id	user_name	user_scree	user_locat	user_time_	user_lang	coc	plac	plac	plac	plac	pla	hashtags	media_url_	extended_	type	lang	retweeted	retweeted	tweet_link	tweet_cl
1	stream__H	Sat Aug 26	Thu Aug 24	9.01E+17	God I'm as	3.25E+09	Camara	_itscamcam_		Pacific Tim	en							[]	[]	[]	[]	en	9.01E+17	javaunbutl	https://tw	god i'm a
2	stream__H	Sat Aug 26		9.01E+17	Be safe all	14936670	Cat Attack	myfilmcha	ATX baby	Central Tir	en							[{'indices':	[]	[]	[]	en			https://tw	be safe a
3	stream__H	Sat Aug 26	Fri Aug 25	9.01E+17	ðŸš¨Pleas	1.68E+09	chan da m	ChandlerV	mart, tx		en							[]	[]	[]	[]	en	9.01E+17	TexasCour	https://tw	please rt
5	stream__H	Sat Aug 26	Thu Aug 24	9.01E+17	I pray for e	1.51E+09	BlizzyðŸŽˆ	nickblizzy2	Las Vegas, NV		en							[]	[]	[]	[]	en	9.01E+17	_____gabt	https://tw	i pray for
6	stream__H	Sat Aug 26	Fri Aug 25	9.01E+17	General su	35510470	ðŸ“	JayLauren_		Central Tir	en							[{'indices':	[]	[]	[]	en	9.01E+17	callergabe	https://tw	general s
7	stream__H	Sat Aug 26		9.01E+17	I still find i	2.91E+09	Ethan Ebui	Eburwell92	Kingston, NY		en							[{'indices':	[]	[]	[]	en			https://tw	i still find
8	stream__H	Sat Aug 26	Sat Aug 26	9.01E+17	#Harvey m	3.71E+09	AmyK34	AmyK341			en							[{'indices':	['https://vi	['video/mp	['video', 'vi	en	9.01E+17	suvCBS	https://tw	disasterh
9	stream__H	Sat Aug 26	Sat Aug 26	9.01E+17	#Harvey m	82349533	Raymond	Iamshenra	Birminghai	Amsterdar	en							[{'indices':	[]	[]	[]	en	9.01E+17	NHC_Atlar	https://tw	disasterh
10	stream__H	Sat Aug 26	Sat Aug 26	9.01E+17	#Harvey m	2.68E+09	AM	_aliyahma	Tx	Central Tir	en							[{'indices':	['https://vi	['video/mp	['video', 'vi	en	9.01E+17	suvCBS	https://tw	disasterh
11	stream__H	Sat Aug 26	Fri Aug 25	9.01E+17	.@heb has	5.73E+08	Christian R	rhettregala	Corpus Chi	Central Tir	en							[{'indices':	[]	[]	[]	en	9.01E+17	RaeOnFOX	https://tw	. usernan
12	stream__H	Sat Aug 26	Fri Aug 25	9.01E+17	That time a	2.22E+09	Samwise	SamCyphe	Angola, NY	Eastern Tir	en							[{'indices':	[]	[]	[]	en	9.01E+17	FOX4	https://tw	that time
13	stream__H	Sat Aug 26	Fri Aug 25	9.01E+17	we really r	7.44E+08	Diana ðŸ–	descobar_		Arizona	en							[]	[]	[]	[]	en	9.01E+17	laryssa_ell	https://tw	we really
14	stream__H	Sat Aug 26	Sat Aug 26	9.01E+17	#Harvey m	9.24E+08	Aug. 26th	Weeta_Gu	Im on a pa	Eastern Tir	en							[{'indices':	[]	[]	[]	en	9.01E+17	NHC_Atlar	https://tw	disasterh
16	stream__H	Sat Aug 26	Sat Aug 26	9.01E+17	#Harvey m	15492983	KimMoldo	KimMoldo	Chicago, IL	Central Tir	en							[{'indices':	[]	[]	[]	en	9.01E+17	NHC_Atlar	https://tw	disasterh
17	stream__H	Sat Aug 26	Fri Aug 25	9.01E+17	we really r	1.97E+09	Ugoh	martinugo	Dallas, TXð	Central Tir	en							[]	[]	[]	[]	en	9.01E+17	laryssa_ell	https://tw	we really
18	stream__H	Sat Aug 26	Sat Aug 26	9.01E+17	Even thou	8.54E+17	Joe	watch_sta	Appalachia - Indian O		en							[{'indices':	[]	[]	[]	en	9.01E+17	NHC_Atlar	https://tw	even thou
19	stream__H	Sat Aug 26	Fri Aug 25	9.01E+17	we really r	8.1E+17	April ðŸŒ	appriilmm	Pasadena, TX		en							[]	[]	[]	[]	en	9.01E+17	laryssa_ell	https://tw	we really
21	stream__H	Sat Aug 26	Sat Aug 26	9.01E+17	NEW: winc	39768696	KATHY	HUEGELE			en							[]	[]	[]	[]	en	9.01E+17	ReedTimm	https://tw	new: win
23	stream__H	Sat Aug 26	Sat Aug 26	9.01E+17	That's the	7.46E+08	Robert Asc	BombRob1	Richland, V	Pacific Tim	en							[{'indices':	['https://vi	['video/mp	['animatec	en	9.01E+17	wxbrad	https://tw	that's the
25	stream__H	Sat Aug 26	Fri Aug 25	9.01E+17	Updated #	9.47E+08	Jack Bentc	EHSSafety	Chicago, IL	Central Tir	en							[{'indices':	[]	[]	[]	en	9.01E+17	DanLindse	https://tw	updated
26	stream__H	Sat Aug 26	Sat Aug 26	9.01E+17	Eyewall of	67559312	Griffin We	GLWeaver	Quartz Hill	Pacific Tim	en							[{'indices':	[]	[]	[]	en	9.01E+17	CNNweath	https://tw	eyewall c
27	stream__H	Sat Aug 26	Fri Aug 25	9.01E+17	we really r	2.45E+09	filipina bin	ceequiqua	301		en							[]	[]	[]	[]	en	9.01E+17	laryssa_ell	https://tw	we really
28	stream__H	Sat Aug 26	Sat Aug 26	9.01E+17	#Breaking:	3.43E+09	Kathy Wel	KathyWels	Matthews,	Pacific Tim	en							[{'indices':	[]	[]	[]	en	9.01E+17	ZachKFOX_	https://tw	otherhas
29	stream__H	Sat Aug 26	Sat Aug 26	9.01E+17	Texas Gulf	1.01E+08	StayUnitec	Vegas0408	Garner, NC	Pacific Tim	en							[{'indices':	[]	[]	[]	en	9.01E+17	CBSLA	https://tw	texas gul
30	stream__H	Sat Aug 26	Fri Aug 25	9.01E+17	BREAKING	7.36E+17	Rebecca B	beccablue	Treaty 1/MÃ©tis hon		en							[{'indices':	[]	[]	[]	en	9.01E+17	weatherch	https://tw	breaking:
31	stream__H	Sat Aug 26	Sat Aug 26	9.01E+17	#Harvey: N	2.34E+09	basic britt	basicbritt		Eastern Tir	en							[{'indices':	['https://vi	['video/mp	['video', 'vi	en	9.01E+17	worldonal	https://tw	disasterh

Figure 2.3: *Example 2 of tweet information stored in CSV File*

2.3.2 Information Filtering

The tweets crawled are multilingual. The majority of the tweets crawled are in English (given that the events crawled happened mostly in the US, and the keywords were in English), but some users posted tweets in other languages, with Spanish being the next frequent language after English. However, given the intended use of the database, in this project, we have only concentrated on tweets that are in English. So, we filtered the English tweets out of the original CSV files, and stored them in separate English CSV files.

2.3.3 Relevant Tweet Classification

Narrowing down and shortlisting the tweets which are directly related to an event can be challenging. We used a machine learning tool developed by Li et al. (2017) to discriminate between the tweets related to disasters and the tweets which are not related to the disasters of interest.

2.3.4 Data Storage

Finally, the English tweets that are relevant to the disasters of interest were stored in a database, as described in the next chapter.

Chapter 3

Database Description

3.1 Tables and Attributes

In this section, we describe the various tables in the database and their columns.

1. TUser: This table keeps information about all the users who are tweeting.

   - user_id : This attribute holds the unique id of each user in the crawled Twitter data.
   - name: This column contains the name of the user who has tweeted the tweet.
   - screen_name: (Primary Key) User screen name, the name which is displayed when the user tweets.
   - location: This column holds the user's location.
   - time_zone: This column holds the timezone of the location from where the user is tweeting.
   - coordinates: User's coordinates.
   - p_bounding_box: The place from where the user is tweeting.
   - p_country_code: The country code where the user belongs.
   - p_country: The country where the user belongs.

- p_full_name: The full name of the place where the user is at.
- p_name: The short name of the place where the user is at.

2. OrgTweets: This table holds information regarding all the original tweets.
   - ot_id (Primray Key): This attribute holds the unique ids of the original tweets.
   - tweet_text: This column contains the tweet text.
   - created at: This column contains the date and time the tweet was posted.
   - lang: This column stores the language of the tweet.
   - tweet link: This column gives the url of the tweet.
   - screen_name (Foreign Key): This column stores the unique user screen names of the user who posted the tweet. It acts as the foreign key to table TUser.
   - source_filename: The JSON file name from which the tweet has been extracted.
   - filtered: This column can take 2 values: 1 if the tweet is in English, 0 otherwise.
3. ReTweets: This table holds information regarding all the retweets.
   - rt_id (Primary Key): This column holds the unique ids of all the re-tweets.
   - created at: This column contains date and time at which the tweet was posted.
   - ot_id (Foreign Key): This column holds the unique id of the original tweet which the retweet refers to. It acts as a foreign key to table OrgTweets.
   - screen_name (Foreign Key): This column holds the unique screen name of the user who posted the retweet. It acts as a foreign key connecting to the User table.
4. Hashtags: This table holds information regarding all the Hashtags used in the tweets.
   - hashtag_id (Primary Key): Unique id of all hashtags used in a tweet.
   - word: This column holds the words of the hashtags.
   - start_ind: It holds the start index of the hashtags in a tweet.

- end_ind: It holds the end index of the hashtag in a tweet.
- ot_id (Foreign Key): Holds the unique id of the original tweet for which the hashtag has been used.

5. Media: This table holds information regarding all the media used in the tweets.

- m_id (Primary Key): Unique ids of all media used in a tweet.
- url: This column holds the web-address of all the media used in the tweet.
- type: This columns holds the type of media, e.g jpeg, video, etc.
- ot_id (Foreign Key): This column holds the unique tweet ids of the tweets in which the media has been used. It acts an as foreign key referring to OrgTweets.

3.2 Entity Relationship Diagram

The Entity-Relationship (E-R) diagram that visually describes the data modeled in stored in the database is shown in Figure 3.1.

3.3 Relational Schema

The relational schema corresponding to the E-R Diagram is shown below.

1. TUser (user_id, name, screen_name, location, timezone, u_lang, coordinates, p_bounding_box, p_country_code, p_country, p_full_name, p_name)

   - PK: screen_name
   - FK: NA

2. OrgTweets (ot_id, tweet_text, created_at, lang, tweet_link, user id,source filename,filtered)

   - PK: ot_id
   - FK: OrgTweets.screen_name references TUser.screen_name

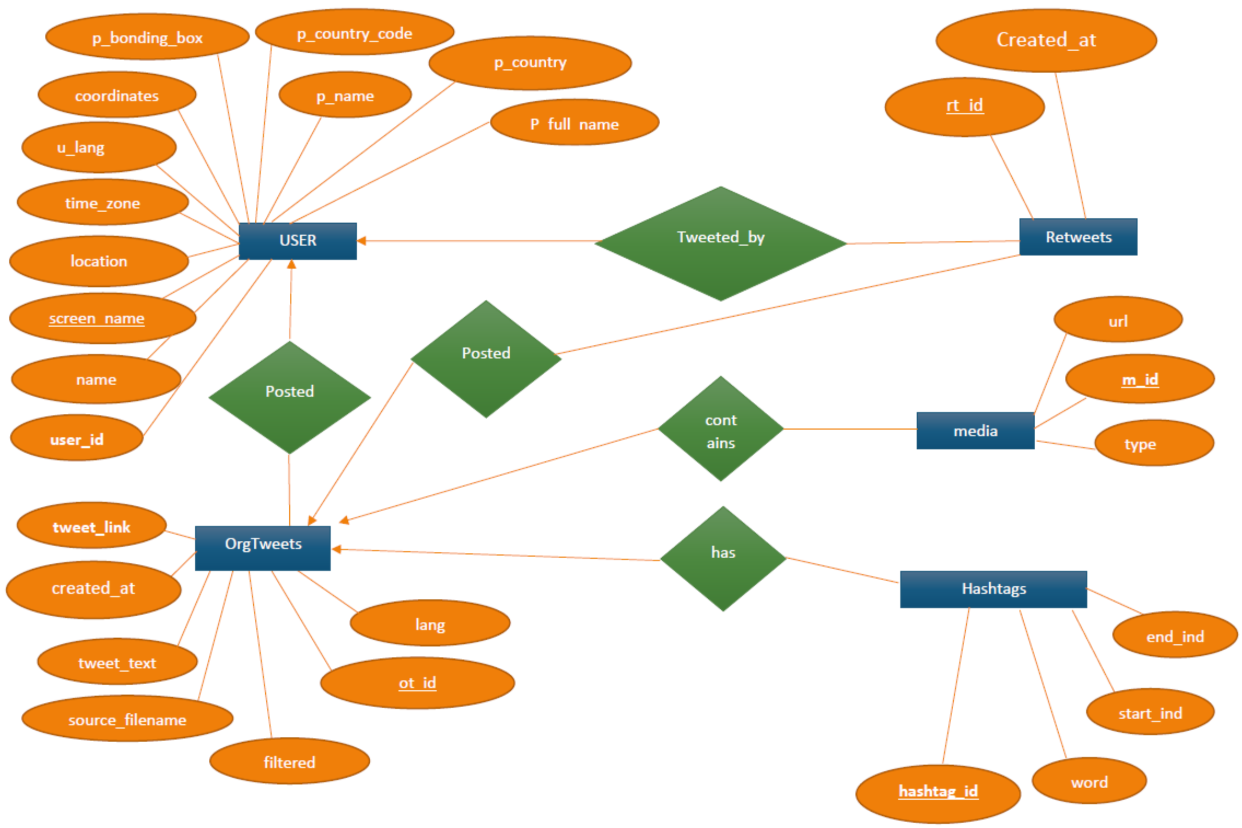

Figure 3.1: *The ER Diagram of the Database*

3. Retweets (rt_id, created_at, ot_id, user_id)

   - PK: rt id
   - FK: Retweets.screen_name references TUser.screen_name
   - FK: Retweets.ot_id references OrgTweets.ot_id

4. Hashtags (hashtag_id, word, start_ind, end_ind, ot_id)

   - PK: hashtag_id
   - FK: Hashtags.ot_id references OrgTweets.ot_id

5. Media (m_id, url, type, ot_id)

   - PK: m id
   - FK: Media.ot_id references OrgTweet.ot_id

Chapter 4

Database Statistics and Queries

4.1 The User Interface

4.1.1 Login

When the web application is run, the login form is displayed. Enter the credentials to login.

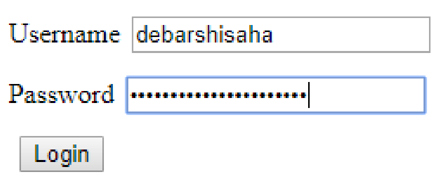

Figure 4.1: *The Login Page*

4.1.2 The Query Page

On successful login, the query page opens as shown in Figure 4.2

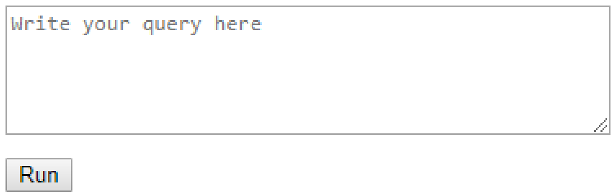

Figure 4.2: *The Query input page*

4.2 Database Record Information

Table 4.1: *Database Table Information*

Sl No	Table Name	No. of Records
1	TUser	3,014,874
2	OrgTweets	2,071,834
3	Retweets	6,783,647
4	Hashtags	1,448,728
5	Media	2,840,885

4.3 Query Set 1

The users who have posted the largest number of tweets (count), in decreasing order of the count:

Query: select TUser.screen_name, TUser.name , count(*) from OrgTweets, TUser where OrgTweets.screen_name = TUser.screen_name GROUP BY TUser.screen_name, TUser.screen_name ORDER BY count DESC;

Description: From this query we get to see the users who have tweeted the most. The results are grouped by the users who have tweeted the most in decreasing order.

4.4 Query Set 2

The number of tweets posted each day:

Query: select tweet_day, count(*) from (select substring(created_at from 5 for 6) tweet_day from orgtweets) as day_of_tweet group by tweet_day;

tweet_day	length	count
Aug 26	6	22593
Aug 27	6	5389
Oct 04	6	17730
Sep 07	6	819
Sep 08	6	261383
Sep 09	6	241211
Sep 11	6	363721
Sep 12	6	176947
Sep 13	6	91395
Sep 14	6	36016
Sep 15	6	10140
Sep 16	6	6342
Sep 17	6	49650
Sep 18	6	8668
Sep 19	6	43049
Sep 20	6	150768
Sep 21	6	96815
Sep 23	6	21457
Sep 24	6	9174
Sep 25	6	89376
Sep 26	6	119194
Sep 27	6	19415
Sep 28	6	1083

(23 rows)

Figure 4.3: *Query results showing the number of tweets posted each day*

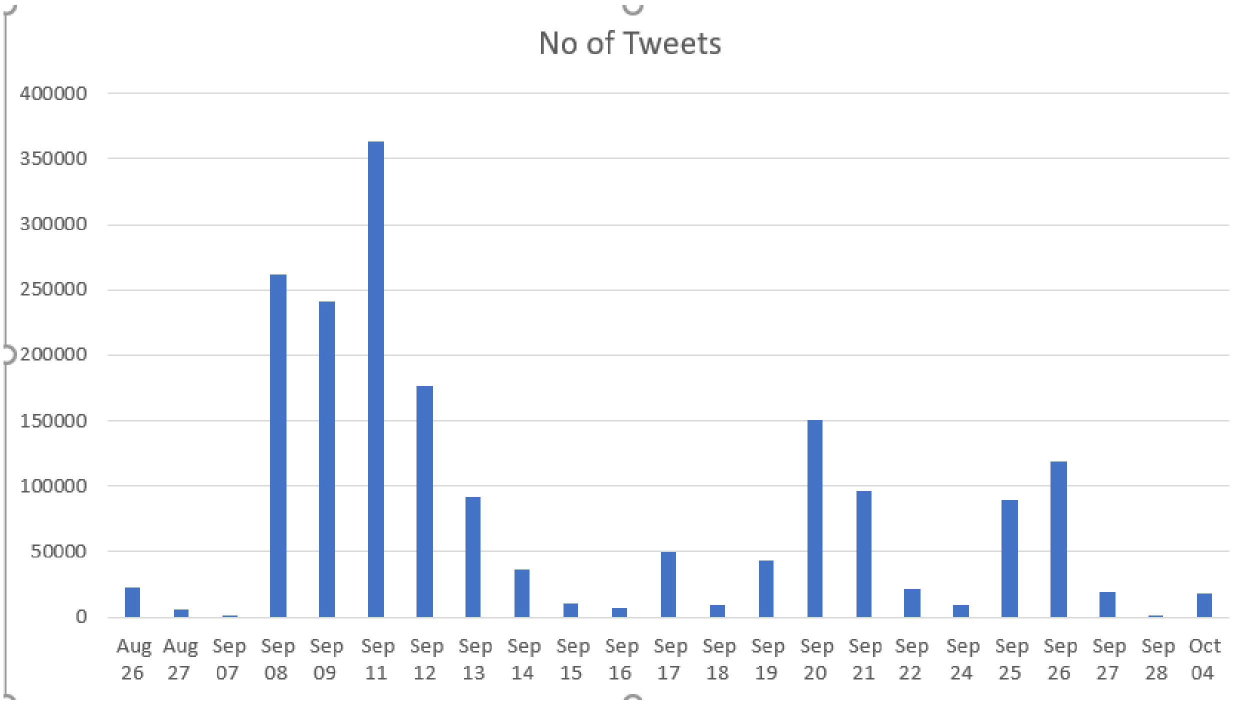

Figure 4.4: *Column chart of tweets posted each day*

Description: From this query Q2, we get an idea of the number of tweets posted each day in a certain time. The figure 4.3 shows the results. The column chart in figure 4.4 gives a graphical representation of the data. The horizontal axis represents the dates on which the tweets have been posted and the vertical axis represents the number of tweets posted.

4.5 Query Set 3

Tweets with the most number of retweets:

Query: select org.ot_id,sum(case when ret.ot_id is null then 0 else 1 end) retweet from orgtweets org left join retweets ret on org.ot_id=ret.ot_id group by org.ot_id order by retweet desc;

ot_id	no_of_retweets
905783770275610624	94434
906965252792832000	89293
906872264313888768	56996
905905797049311232	39912
906974851470168067	38478
906579283950403585	36059
912386484686204928	32425
912724426709504000	32104
910672452790841344	31580
907221230801088512	31225
906171275869118464	27319
901124355907866625	26793
906991730611953666	26281
905615147070259200	21978
912347377389928448	18855
906951381856215041	18629
906993475484237824	18450
907024735296516096	18163
907349794854711296	17318
912099935444766727	16314
900867990396252161	16225
910896407418228736	15957
911605641692700672	13919
911984783194050560	13718
910566790585110529	13478
906144438916931584	13446
911424890233851904	12091
911401176469565442	11559
912483249527644160	11151
906172777576579072	10855
910328626075389952	10599
907019340935045125	10037
905591967735742464	9938

Figure 4.5: *Query results showing the tweets having the maximum number of retweets*

Description: Figure 4.5 shows the output of the query. From this we get an idea of how many times an original tweet has been re-tweeted.

4.6 Query Set 4

Hashtags which have been used the most in the Tweets.

Query: select word, count(*) c from hashtags group by word order by c desc ;

word	c
Irma	155707
HurricaneIrma	91845
irma	45815
Harvey	35561
Maria	30211
Florida	25020
hurricaneirma	21018
HurricaneMaria	16684
PuertoRico	14680
Hurricane	14491
IRMA	14077
HurricaneHarvey	11148
news	10963
hurricane	10873
earthquake	10844
Mexico	10153
IrmaHurricane2017	8539
harvey	7043
Miami	7032
Houston	6951
News	6903
Jose	6328
IrmaHurricane	5745
MONEY	5489
hurricaneirma2017	4643
florida	4499
Cuba	4322
Earthquake	4163
Irma2017	4063
maria	4053
Trump	3852
job	3698
USA	3433
climatechange	3362
Hiring	3346
miami	3296
Caribbean	3057
Texas	3040
BREAKING	2891

Figure 4.6: *Query results showing the tweets with the highest number of retweets*

Description: Figure 4.6 shows the output of the query. From this we get an idea of how many times a particular hashtag has been used in all the tweets.

4.7 Query Set 5

Queries to determine the number of tweets in the database of each disaster type.

1. SELECT count(*) FROM OrgTweets WHERE source_filename like ´%earthquake%´ order by count desc;

2. SELECT count(*) FROM OrgTweets WHERE source_filename like ´%irma%´ order by count desc;

3. SELECT count(*) FROM OrgTweets WHERE source_filename like ´%Harvey%´ order by count desc;

4. SELECT count(*) FROM OrgTweets WHERE source_filename like ´%maria%´ order by count desc;

Table 4.2: *Tweet Count Based on Disasters*

Sl No	Disaster Name	Number of Tweets
1	Maria	3,014,874
2	Irma	276,575
3	Harvey	27,983
4	Mexico Earthquake	462,865

4.8 Query Set 6

Queries to determine the number of tweets posted each day related to a natural disaster.

1. **Hurricane Maria:** select tweet_day, count(*) from (select substring(created_at from 5 for 6) tweet_day from (SELECT * FROM OrgTweets WHERE source_filename SIMILAR TO ´%maria%´ as a) as day_of_tweet group by tweet_day order by tweet_day ASC;

tweet_day	count
Oct 04	17730
Sep 19	39513
Sep 20	107118
Sep 21	74272
Sep 23	2964
Sep 24	2444
Sep 25	85671
Sep 26	116255
Sep 27	16898

Figure 4.7: *Query results showing the number of tweets posted each day related to Maria*

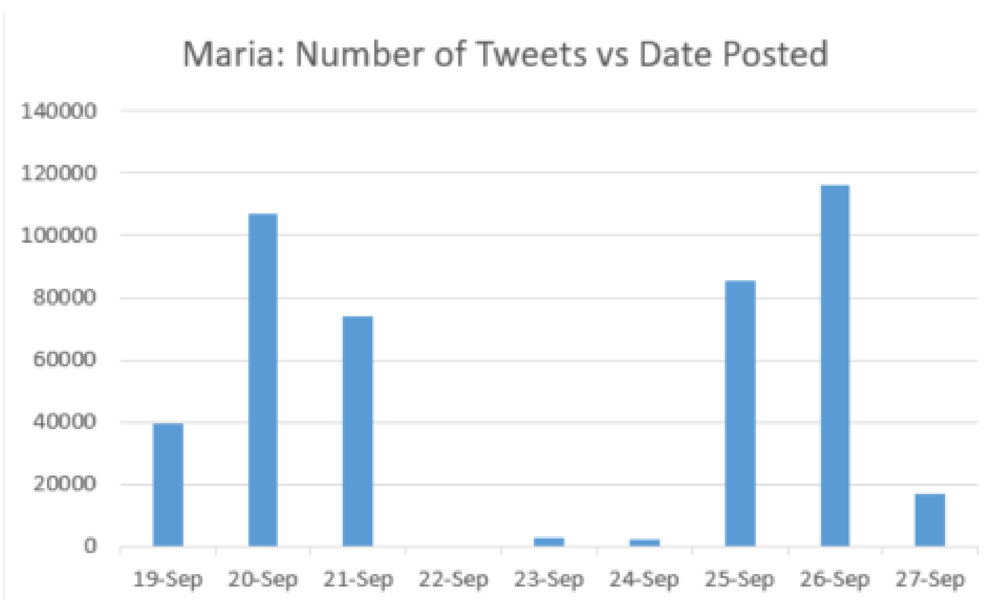

Figure 4.8: *Column chart showing the number of tweets posted each day related to Maria*

2. **Mexico Earthquake:** select tweet_day, count(*) from (select substring(created_at from 5 for 6) tweet_day from (SELECT * FROM OrgTweets WHERE source_filename SIMILAR TO ´%earthquake%—%mexico%´) as a) as day_of_tweet group by tweet_day order by tweet_day ASC;

tweet_day	**count**
Sep 09	4346
Sep 11	3393
Sep 20	42243
Sep 21	21686
Sep 23	18493
Sep 24	6730
Sep 25	3705
Sep 26	2939
Sep 27	2517
Sep 28	1083

Figure 4.9: *Query results showing the number of tweets posted each day related to Mexico Earthquake*

3. **Hurricane Irma:** select tweet_day, count(*) from (select substring(created_at from 5 for 6) tweet_day from (SELECT * FROM OrgTweets WHERE source_filename SIMILAR TO ´%irma%´) as a) as day_of_tweet group by tweet_day order by tweet_day ASC;

4. **Hurricane Harvey:** select tweet_day, count(*) from (select substring(created_at from 5 for 6) tweet_day from (SELECT * FROM OrgTweets WHERE source_filename SIMILAR TO ´%Harvey%´) as a) as day_of_tweet group by tweet_day order by tweet_day ASC;

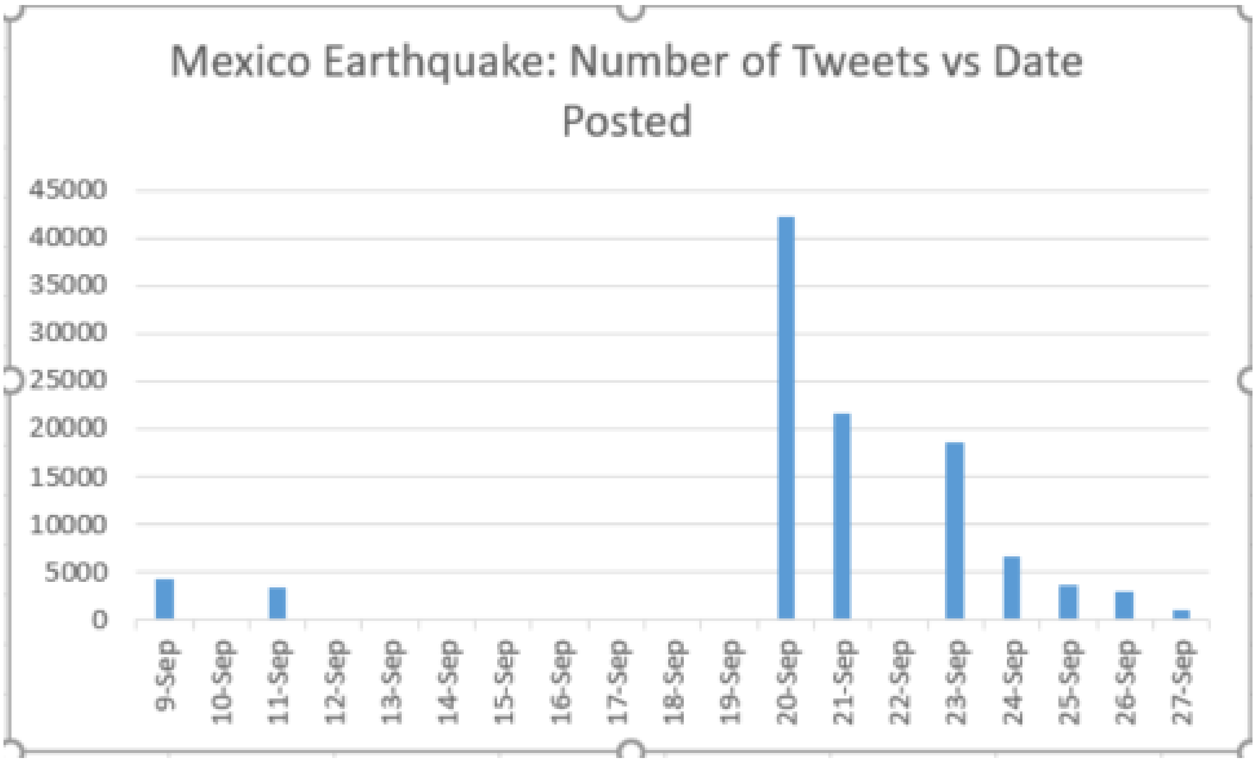

Figure 4.10: *Column chart showing the number of tweets posted each day related to Mexico Earthquake*

tweet_day	count
Sep 09	67513
Sep 11	86588
Sep 12	38128
Sep 13	27695
Sep 14	19447
Sep 15	10140
Sep 16	6342
Sep 17	6254
Sep 18	8668
Sep 19	3536
Sep 20	1407
Sep 21	857

Figure 4.11: *Query Results: Showing Number of tweets posted each day related to Irma*

4.9 Query Set 7

Queries to determine the number of video and pictures in the original tweets.

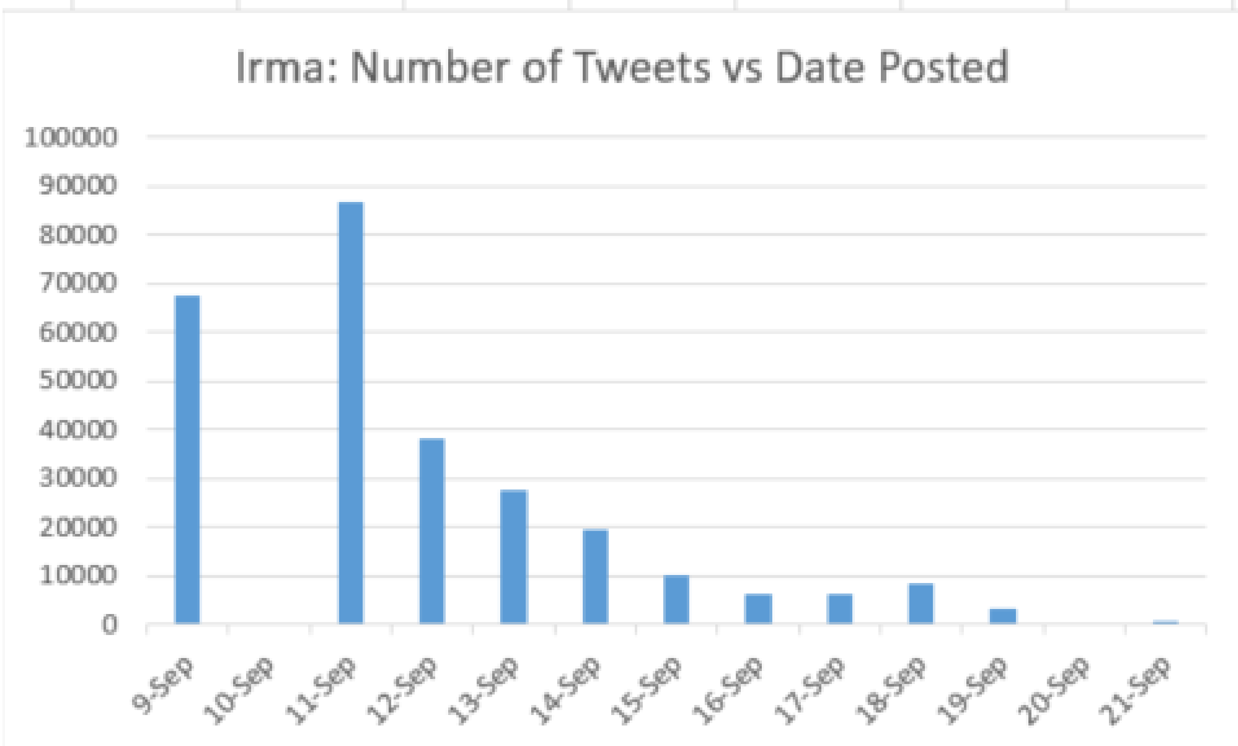

Figure 4.12: *Column chart showing the number of tweets posted each day related to Irma*

tweet_day	count
Aug 26	22594
Aug 27	5389

Figure 4.13: *Number of tweets posted each day related to Harvey*

Number Picture Links in original tweets:

- SELECT count(*) Number_of_Pictures FROM Media WHERE type SIMILAR TO '%photo%';
- SELECT count(*) Number_of_Videos FROM Media WHERE type SIMILAR TO '%video%';

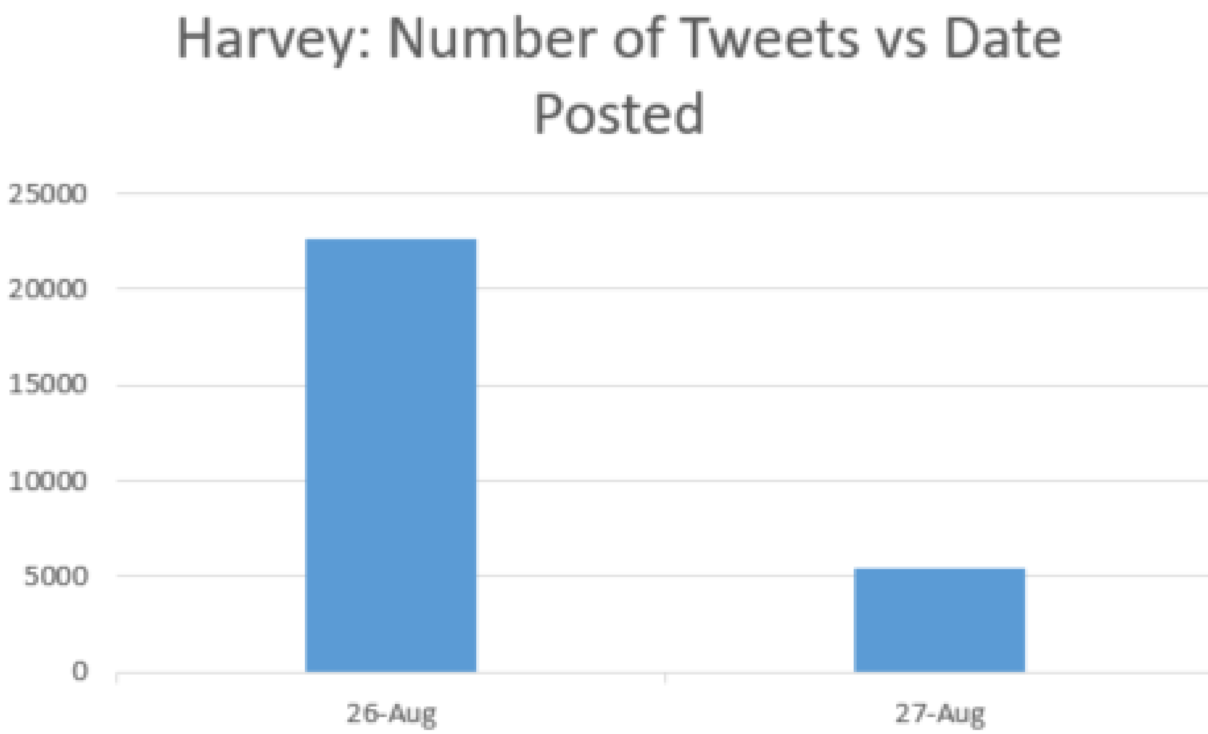

Figure 4.14: *Bar Graph showing number of tweets posted each day related to Harvey*

number_of_pictures
228678

Figure 4.15: *Number of picture links*

number_of_videos
63206

Figure 4.16: *Number of video links*

Chapter 5

Word Cloud Analysis

5.1 What are Word Clouds?

Word clouds or tag clouds are graphical representations of word frequencies that give greater prominence to words that appear more frequently in a source text. The larger the word in the visual cloud, the more common the word was in the document(s). This type of visualization can assist evaluators with exploratory textual analysis by identifying words that frequently appear in a set of interviews, documents, or other text. It can also be used for communicating the most salient points or themes in the reporting stage (BetterEvaluation, 2015). The following figures show word clouds for the disasters in our dataset.

5.1.1 Word Cloud: Hurricane Maria

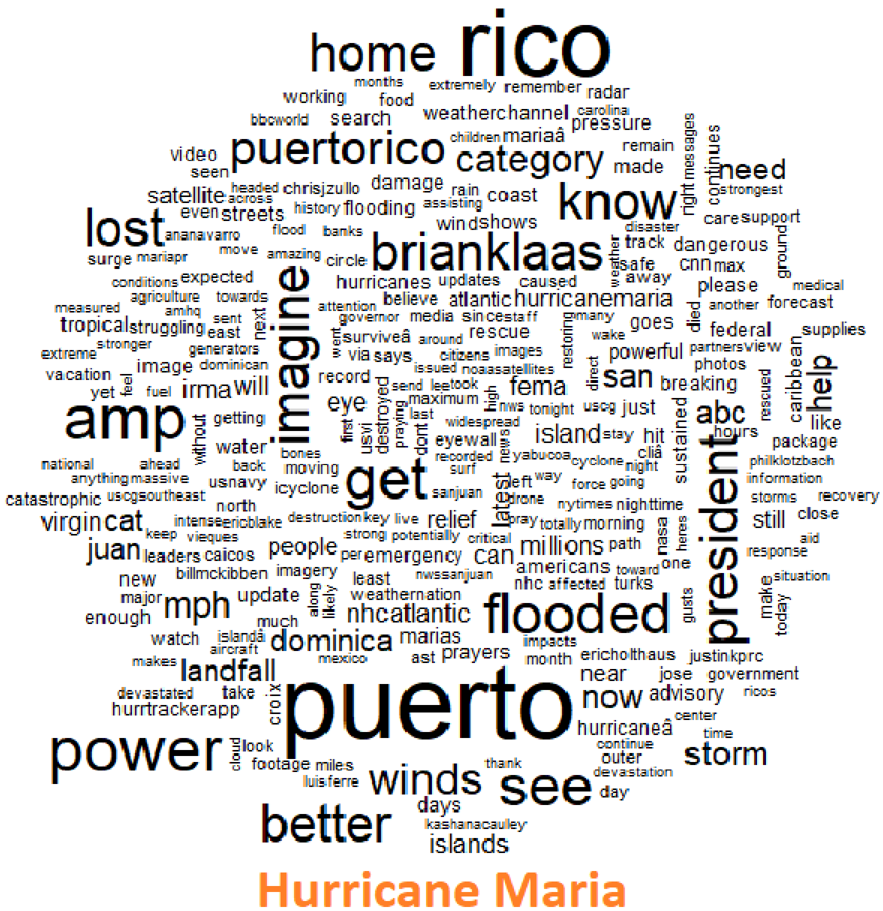

Figure 5.1: *Word Cloud: Hurricane Maria*

Description: As mentioned earlier in Section 5.1, more a specific word appears in a source of textual data, the bigger and bolder it appears in the word cloud. In the Figure 5.1, we can see the word 'puerto' is the biggest, that means it appeared the maximum number of times followed by the word, 'rico'. Then the other words such as, 'imagine', 'amp', 'power', 'better', 'president', 'get', 'category', 'flooded', 'better' and few more which are comparatively smaller in size. We can conclude that they have appeared a considerable number of times which emphasizes on the fact that the users are also concerned about these areas.

5.1.2 Word Cloud: Hurricane Irma

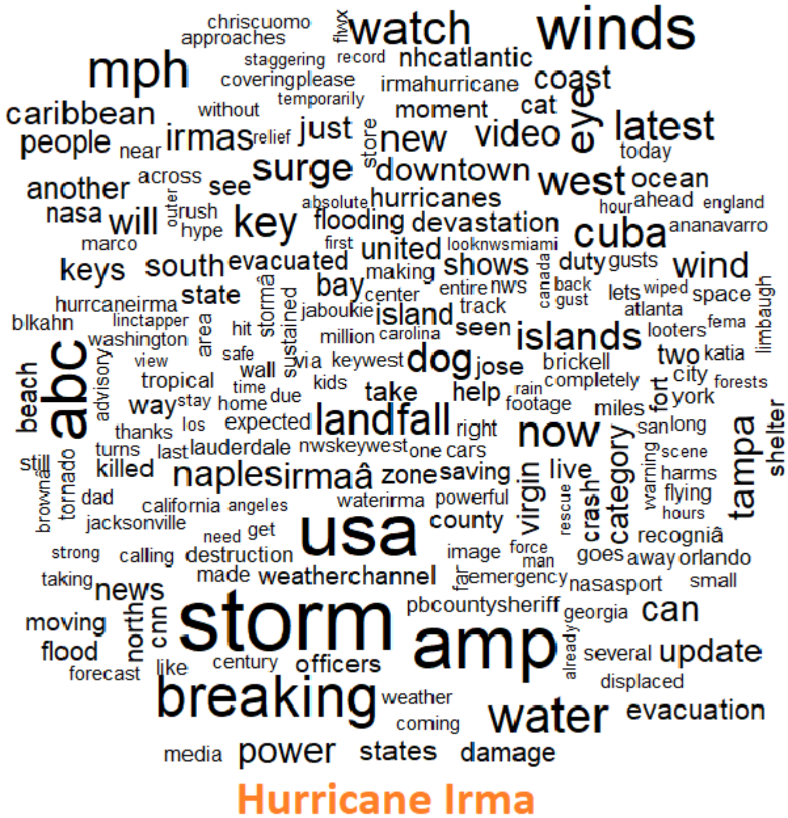

Figure 5.2: *Word Cloud: Hurricane Irma*

Description: In the Figure 5.2, we can see the word 'storm' is the biggest, that means it has appeared the maximum number of times. We can see that people are mostly using that word in their tweets. Then the other words such as, 'usa', 'winds', 'breaking', 'water', 'mph' and few more which are comparatively smaller in size. We can conclude that they have appeared a considerable number of times which emphasizes on the fact that the users are also concerned about these areas.

5.1.3 Word Cloud: Hurricane Harvey

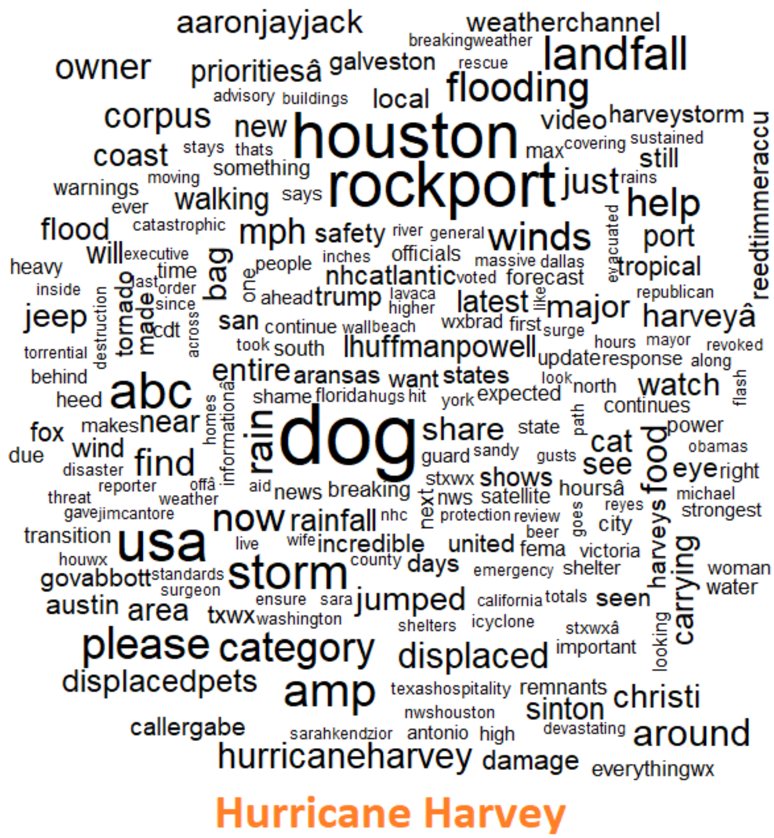

Figure 5.3: *Word Cloud: Hurricane Harvey*

Description: In the Figure 5.3, we can see the word 'dog' is the biggest, that means it has appeared the maximum number of times. Then the other words such as, 'usa', 'houston', 'landfall', 'rockport', 'storm', 'please','category' and few more are comparatively smaller in size. We can conclude that they have appeared a considerable number of times which emphasizes on the fact that the users are also concerned about these areas.

5.1.4 Word Cloud: Mexico Earthquake

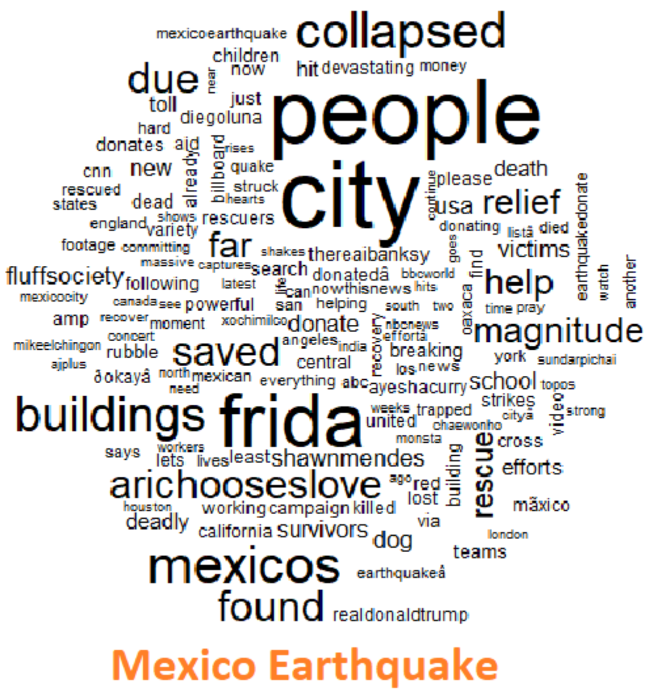

Figure 5.4: *Word Cloud: Mexico Earthquake*

Description: In the Figure 5.4, we can see the word 'city' and 'peolple' are among the biggest words, that means they have appeared the maximum number of times. Then the other words such as, 'frida', 'buildings', 'collapsed', 'found', 'rescue', 'due', 'magnitude' and few more which are comparatively smaller in size. We can conclude that they have appeared a considerable number of times which emphasizes on the fact that the users are also concerned about these areas.

5.1.5 Word Cloud: California Wildfire

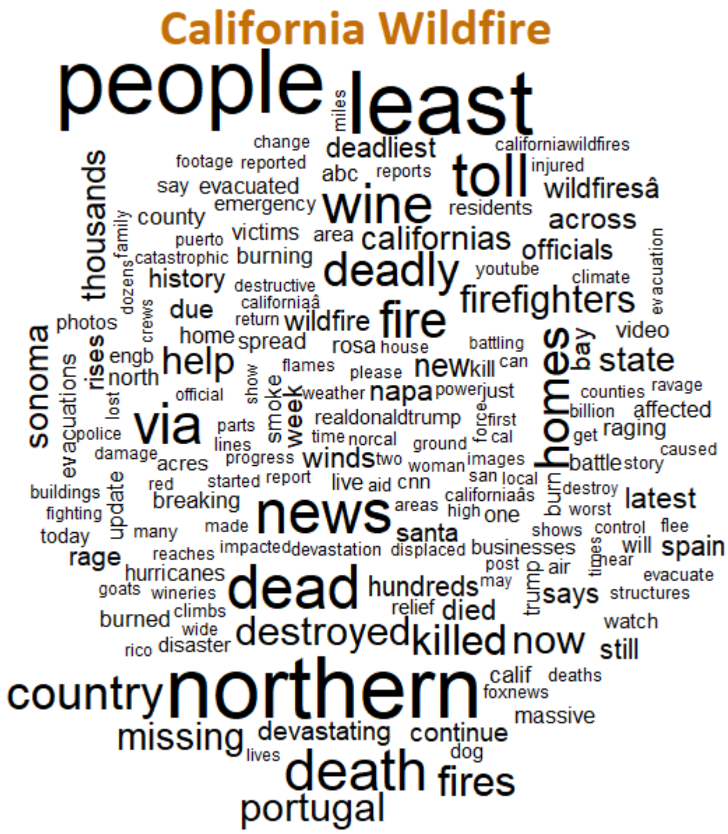

Figure 5.5: *Word Cloud: California Wildfire*

Description: In the Figure 5.5, we can see the words, 'least', 'peolple' and 'northern' are among the biggest words, that means they have appeared the maximum number of times. Then the other words such as, 'dead', 'deadly', 'homes', 'destroyed', 'killed', 'country', 'missing', 'devastating' and few more which are comparatively smaller in size. We can conclude that they have appeared a considerable number of times which emphasizes on the fact that the users are also concerned about these areas.

5.2 Comparison Clouds

In addition to word clouds, we also perform cloud comparisons between disasters, with the goal to identify similarities and differences in terms of word patterns in different disasters.

5.2.1 California Wildfires versus Hurricane Harvey

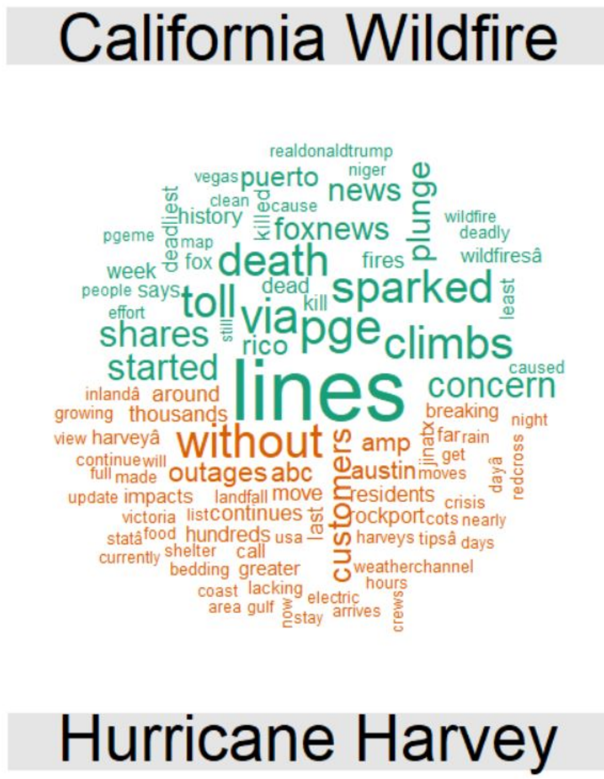

Figure 5.6: *Comparison Cloud: California Wildfire versus Hurricane Harvey*

Description: The Figure 5.6 is the comparison cloud of the tweets which are related to power. In the tweets related to California wildfire, people have used the words like 'lines', 'concern', 'death' , 'toll' etc. while the tweets of Hurricane Harvey have words such as 'without', 'customers', 'breaking', 'outages', 'residents'.

5.2.2 California Wildfire versus Hurricane Irma

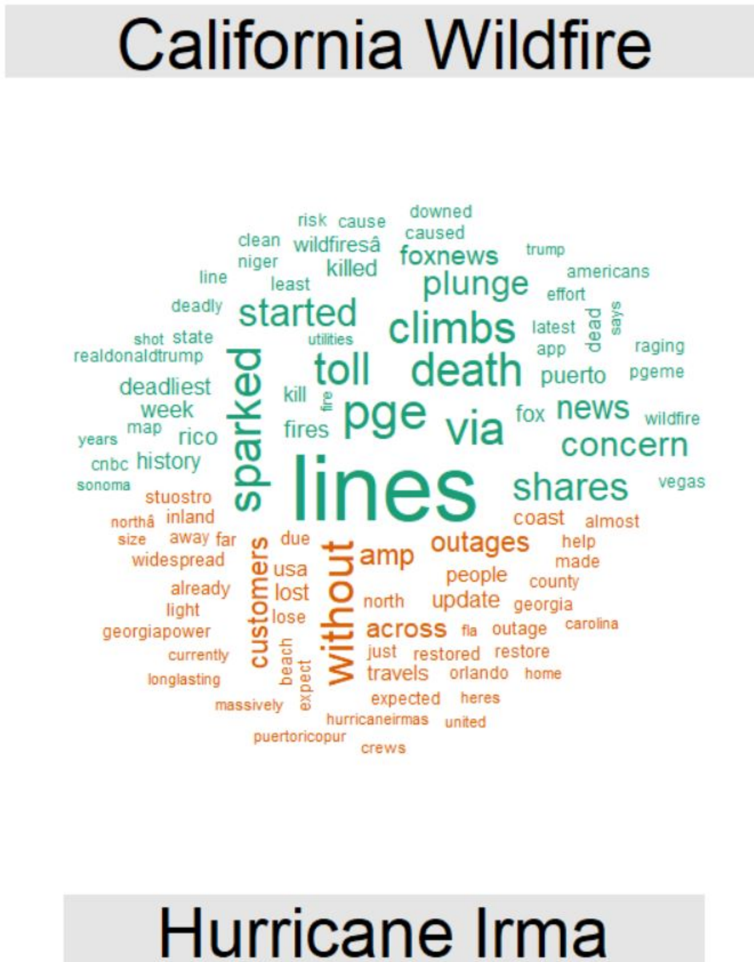

Figure 5.7: *Comparison Cloud: California Wildfires versus Hurricane Irma power tweets*

Description: The Figure 5.7 is the comparison cloud of the tweets which are related to power. In the tweets related to California wildfire, people have used the words like, 'lines', 'concern', 'death' , 'toll', 'sparkled' etc. while the tweets of Hurricane Irma have words such as 'without', 'customers', 'breaking', 'outages', 'lost'. In the former case we can conclude that power lines have contributed to deaths while in the latter case people are without power.

5.2.3 Hurricane Irma versus Hurricane Harvey

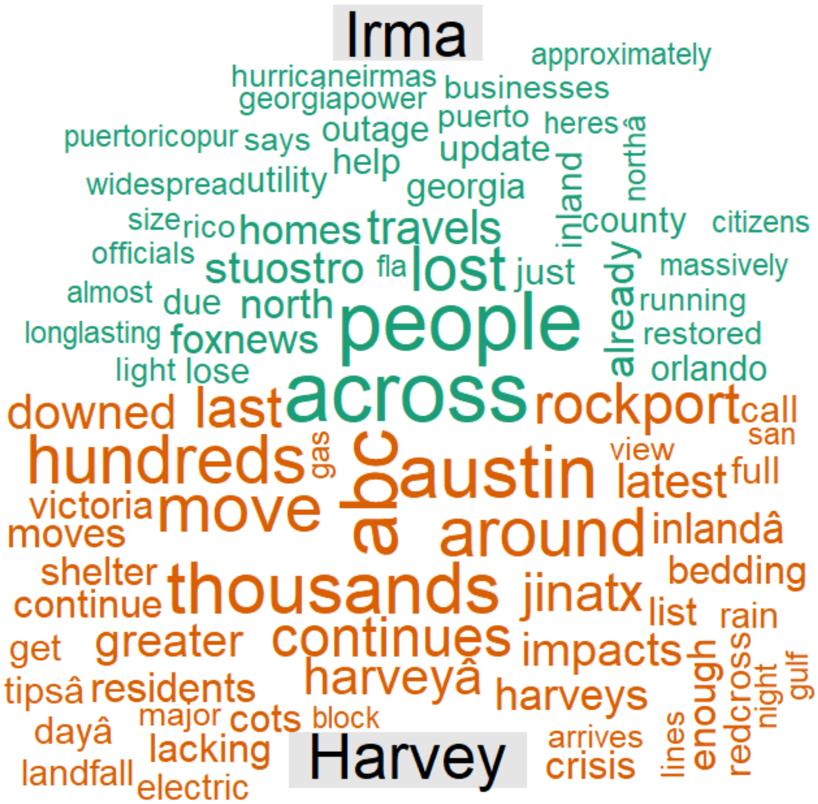

Figure 5.8: *Comparison Cloud: Hurricane Irma versus Hurricane Harvey tweets*

Description: Figure 5.8 is the comparison cloud of the Hurricane Irma and Hurricane Harvey tweets. In the tweets related to Irma, people have used the words like 'people', 'lost', 'already' etc. while the tweets of Hurricane Harvey have words such as 'thousands', 'austin', 'continues', 'impact' etc.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

This project is a very basic implementation of a database for disaster tweet storage and analysis. This project lays the groundwork upon which additional features and improvements can be added. One of the major component was the database design. This is a very important aspect of the application since the ease of querying the data relies on the way the database has been designed. Another goal was to make it future proof. Although a lot of thought went into the design, there is still scope for improvement. Another thing to note is that the tweets stored are the original tweets which have been posted by the users. They might contain foreign characters which are difficult to interpret. One needs to understand that the data stored is all crowd-sourced information, and some of this information may not be accurate or may be inappropriate. We should also note that the tweets stored in the database are only English tweets. We have excluded the ones which are in different languages. This is a limitation that would need to be addressed if the disaster site is at a location where the native language of the people is not English. We will be missing out on many tweets posted in local languages. These will mostly be eyewitness tweets, hence missing out on valuable insights for the recovery team.

6.2 Future Work

There are a lot of features which can be added to the application.

- Designing a new user interface with more interactive features.
- Adding more querying functionalities.
- Changing the back-end to a NoSQL database for more data management capabilities.
- Geo-tagging the tweets which might help to visualize where and which parts of the world the tweets came from. The application can also be integrated with Google maps.
- Adding UI features like option buttons or drop-down lists to select data instead of writing queries.
- As Tweeter provides an API that allows real-time data crawling, one can automate the process to load the data directly in the database, which would give real-time insight to the disaster management teams when they carry out the rescue operations during natural disasters.
- Using a translator to translate the tweets in foreign languages to English.

Bibliography

BetterEvaluation (2015). Word Clouds. http://www.betterevaluation.org/en/evaluation-options/wordcloud.

Dennis, B. (2016). Why your tweets could really matter during a natural disaster! https://www.washingtonpost.com/news/energy-environment/wp/2016/03/11/twitter-might-help-pinpoint-worst-damage-after-a-natural-disaster-study-finds/?utm_term=.8d81ce8b08b8.

Li, H., Caragea, D., Caragea, C., and Herndon, N. (2017). Disaster response aided by tweet classification with a domain adaptation approach. *Journal of Contingencies and Crisis Management (JCCM), Special Issue on HCI in Critical Systems. In press.*

Zarin, K. (2017). 10 Deadliest Natural Disasters of 21st Century. http://www.scienceve.com/10-deadliest-natural-disasters-of-21st-century/.