Abstract

The Twitter Metadata project was a class project as part of CS 6604. In it, I created TweetID, a tool for twitter collection collaboration and I define standards for collection level and tweet level metadata.
# Table of Contents

Table of Figures.................................................................................................................. 2
Executive Summary.................................................................................................................. 3
Users’ Manual....................................................................................................................... 4
Developer’s Manual............................................................................................................... 4
   Technologies Used .............................................................................................................. 5
   Setup Guide ....................................................................................................................... 5
   Prerequisites .................................................................................................................... 5
   Installation Requirements ............................................................................................... 5
   Before Running for the first time ..................................................................................... 5
   Running command-line commands ................................................................................. 5
   High-level explanation of files ....................................................................................... 6
   Current Limitations (next steps) ..................................................................................... 6
Lessons Learned..................................................................................................................... 7
   Proposed Timeline ........................................................................................................... 7
   Problems and Solutions .................................................................................................. 7
      Tweet-Level Metadata ................................................................................................. 7
      Collection-Level Metadata ......................................................................................... 8
      TweetID ......................................................................................................................... 11
Acknowledgements............................................................................................................... 15
References ............................................................................................................................. 16
Appendix A: Screenshots .................................................................................................... 17
Table of Figures

Figure 1: TweetID Hompeage .................................................................................................................. 4
Figure 2: Proposed Timeline ..................................................................................................................... 7
Figure 3: Example Namespaced JSON Extracted Using TweetID ................................................................. 8
Figure 4: Who, What, When, Where, How of Metadata .............................................................................. 9
Figure 5: Core Prov Objects .................................................................................................................... 10
Figure 6: PROV-O Starting Point Classes .................................................................................................. 12
Figure 7: Example Collection Level Metadata in Turtle Syntax ................................................................. 13
Figure 8: YourTwapperKeepr JSON File .................................................................................................... 14
Figure 9: Tab Separated Value Input File .................................................................................................. 14
Figure 10: Upload Screen .......................................................................................................................... 17
Figure 11: Collections View ..................................................................................................................... 18
Figure 12: Collection Details View ............................................................................................................ 19
Figure 13: Merged Collection Details View ............................................................................................... 20
Figure 14: Tweets Screen ............................................................................................................................ 21
Figure 15: Tweet Detail Screen .................................................................................................................. 22
Executive Summary

A number of projects and research efforts work with collections of tweets. Of particular interest is the collection of tweets related to world events. Many organizations have their own individual tweet collections regarding specific events; however, there is currently no effective support for collaboration. Metadata standards foster collaboration by allowing groups to adhere to a unified format and seamless interoperate. In part one of the Twitter Metadata project, I define a tweet-level metadata standard that leverages the Twitter API format, as well as a collection-level metadata standard which combines Dublin Core and PROV-O. By combining two diverse existing standards (Dublin Core and PROV-O) into a single RDF document, the proposed standard is able to capture both the descriptive metadata as well as provenance of the collections. In part two of the Twitter Metadata project, I create a tool called TweetID in order to further foster collaboration with tweet collections. TweetID is a web application that allows its users to upload tweet collections. TweetID extracts, and provides an interface to, the underlying tweet-level and collection-level metadata. Furthermore, TweetID also provides the ability to merge multiple collections together, allowing researchers to compare their collections to others’, as well as potentially augment their event collections for higher recall.
Users’ Manual

A video demo can be found through the file *Michael Shuffett – TweetID demo.mp4* which goes step by step through each screen and piece of functionality of TweetID.

**TweetID** is a project which supports the open sharing of Twitter metadata. With **TweetID**, you can upload tweet collections, view uploaded tweets and collections, and merge collections. **TweetID** extracts collection level metadata from the uploaded collection. For user’s interested in metadata, **TweetID** also provides unique uri’s for collections as well as tweets. **TweetID** also extracts the standard Twitter JSON from a variety of formats (useful for converting for collaboration).
Developer's Manual

Technologies Used

git. **Python 2.7. Flask** for the server. **sqlite3** for the db. **SQLAlchemy** with Flask-SQLAlchemy for the ORM. **jinja2** for the templates. **WTForm** and Flask-WTForm for forms, validation, and uploading. **Bootstrap** and Flask-Bootstrap for the CSS. **jQuery** for the client side scripting. **Flask-Script** for easy management.

Most of the code is pretty self-explanatory and serves as the best natural documentation. The challenge is understanding the large array of technologies used.

For an introduction to Flask, I recommend that you go through the documentation on the flask website.

Setup Guide

Prerequisites

- Python 2.7
- pip

Installation Requirements

The requirements have been listed in requirements.txt. In order to install all of the requirements, simply run `pip install -r requirements.txt`. (You might need sudo).

Before Running for the first time

You will need to edit the tweetid/config.py file. Change UPLOAD_FOLDER to the absolute path to the folder to store the uploads in and change INSTANCE_FOLDER_PATH to a temporary folder. On windows I recommend C:/Users/<user_name>/AppData/Local/Temp/ and on Linux I recommend /tmp. You will then need to initialize the database as shown below.

Running command-line commands

TweetID uses Flask-Script for runnable commands from the command line. You can find these commands in manage.py. To see the command line help just type `python manage.py`. To see the help for a specific command type `python manage.py command -h`.

To start out, cd into the same directory that manage.py is in.

Initializing the database

First we need to init the database (currently sqlite3 through SQLAlchemy and Flask-SQLAlchemy).

```
python manage.py init_db
```

Clearing the database

```
python manage.py drop_all
```

Starting the server

```
python manage.py run
```
By default the server runs only on the localhost. In order to specify a different ip address or port you can use the following command. Again, to see the help information for this or other commands you can pass the –h flag.

```bash
python manage.py runserver –t HOST –p PORT
```

Load a YourTwapperKeeper JSON file from stdin
Note: Some of the collection level metadata may be hard coded.

```bash
cat file.json | python manage.py import_json
```

Load a tsv file of the format provided by QCRI from stdin
Note: Some of the collection level metadata may be hard coded.

```bash
cat file.tsv | python manage.py load_tsv
```

Downloading json files from YourTwapperKeeper
Currently files are simply downloading using the json API and wget (not programmatically). You can use a call like follows. (Replace the variables between <>). Any tweet limit above ~80000 was found to cause a server error.

```bash
wget http://spare05.dlib.vt.edu/apiGetTweets.php?id=<archive id>&l=<tweet limit> –O <output filename>
```

High-level explanation of files
The main file is app.py. The setup of the server and all of the routing happens there. models.py is where the SQLAlchemy models are setup. load_json.py and tsv.py are utility files for loading the respective files. All of the templates are in the templates/ folder. They all inherit from base.html. The static/ folder is where all of the static files that are loaded are stored (js, css). tweetid.db in the root folder is the sqlite database. It isn't in the git but I will include a copy of it in the files so the developer has some test data to work with.

Current Limitations (next steps)

- Currently upload only works on the tsv files with the format from QCRI. Loading the YourTwapperKeeper JSON files should be extremely easy since the code is already largely there.
- Extra input metadata should go into an extra field in the database and populate the organization's namespace.
- Merged collections aren't persisted.
- The name of a collection is being used as the unique id but that might not be desirable.
- After a collection has been merged it would be nice to still know where the tweets came from.
- If the server is to be used for more than research purposes, we should use an enterprise level server not the internal Flask server.
Lessons Learned

Proposed Timeline

<table>
<thead>
<tr>
<th>Task</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reach out to stakeholders</td>
<td>Feb 5</td>
</tr>
<tr>
<td>Initial meeting with Mohamed</td>
<td>Feb 14</td>
</tr>
<tr>
<td>Gain access to QCRI Twitter collections</td>
<td>Feb 16</td>
</tr>
<tr>
<td>Gain access to VT Twitter collections</td>
<td>Feb 26</td>
</tr>
<tr>
<td>Proposal for metadata fields</td>
<td>Mar 6</td>
</tr>
<tr>
<td>Have base architecture for metadata extraction implemented</td>
<td>Mar 20</td>
</tr>
<tr>
<td>Extracted desired features from collections</td>
<td>April 10</td>
</tr>
<tr>
<td>Allow for merging of collections</td>
<td>April 25</td>
</tr>
<tr>
<td>Turn in deliverables</td>
<td>May 8</td>
</tr>
</tbody>
</table>

**Figure 2: Proposed Timeline**

Problems and Solutions

Tweet-Level Metadata

I spent some time evaluating potential solutions to a tweet-level metadata standard. Two design goals emerged that help inform the final decision. 1) Make conversion to the new standard as straightforward as possible for all varieties of stored formats. 2) Make the standard flexible enough to support any metadata fields an organization might have.

A key insight emerged that leads to the effective support of the first goal. No matter what format the tweets end up being stored as, they were all initially ingested from the standard twitter API format (Twitter, Inc., 2013). The format behind this API is well defined and inclusive.

However, the Twitter API format is rigid. In order to meet the second goal, the approach of namespacing the JSON by organization was taken. The JSON which would have originated from the Twitter API is nested under a `{Twitter:}` object, while the fields unique to any organization are listed under the name of the organization (see Figure 3 for an example that was automatically extracted using TweetID). This approach maintains the high detail and interoperability of the Twitter API standard, while allowing organizations to specify any additional fields they may desire.

Recommended Format

The recommended metadata format for a tweet is to use JSON where under the root object there is a mandatory “twitter” object and optional objects for each organization. Under the “twitter” object, either “id” or “id_str” should always appear. All other fields are optional. The organization objects are meant to be a catchall for any items that are not representable by the Twitter API format. The idea is to define a mapping from your native format to the Twitter format within TweetID. Anything that is not
mappable can be automatically contained in the organization-level metadata. Figure 3 is an example of
the desired JSON that was extracted automatically using TweetID. The tweetid object has a single entry,
collections, which automatically has a list of all of the collections the tweet belongs to.

```json
{
    "twitter": {
        "entities": {
            "urls": [
                {
                    "expanded_url": "http://dlvr.it/2mv5Pp"
                }
            ]
        },
        "created_at": "2013-01-27 08:37:34",
        "id_str": "29545046539064321",
        "coordinates": {
            "type": "Point",
            "coordinates": [
                10800.0,
                10000.0
            ]
        },
        "screen_name": "Fachurrozy__"
    },
    "tweetid": {
        "collections": [
            "Queensland Floods"
        ]
    }
}
```

**Figure 3: Example Namespaced JSON Extracted Using TweetID**

Collection-Level Metadata
In addition to the Tweet-Level Metadata standard, a collection-level standard was required. The
standard needed to include metadata covering a wide variety of areas.

The following are some of the fields that were identified that the standard should support:

- start and end, time and date stamps, indicating the time coverage of the collection
- geographic coverage of the collection
  - of those tweeting
  - of what is discussed in the tweets
- names and other information on the person, group, and institution that did the collecting
- keyword lists or other query characterizations used to get the tweets
• other details on how the tweet collection was prepared
  o if there was filtering of the original raw tweets collected
  o name/URI of raw tweet collection
  o format of that collection
  o details on how this collection was obtained from the raw tweet collection e.g., features and classifiers and parameters employed
  o if additional metadata has been added after the raw collecting, e.g., labels -- and details about that process

**Figure 4: WHO, WHAT, WHEN, WHERE, HOW OF METADATA**

It was clear that the typical metadata standards such as Dublin Core (DCMI Usage Board, 2012), were not sufficiently expressive to capture some of the details about the preparation of the collections. As you can see in Figure 4, the metadata was split into the who, what, when, where, and how, of the collection. “The what” of the collection could be addressed using typical metadata representations such as the Dublin Core Terms (DCMI Usage Board, 2012) but the who, when, and how of the collection would need to be expressed using a different type of metadata standard.

PROV is a set of W3 technical reports about provenance. In PROV-Overview, provenance is defined as “information about entities, activities, and people involved in producing a piece of data or thing, which can be used to form assessments about its quality, reliability or trustworthiness (Groth & Moreau, 2013).” Within the PROV family of documents, I identified PROV-O an OWL2 ontology allowing the mapping of the PROV data model to RDF (Belhajjame, et al., 2013).
In PROV Model Primer, the authors present a conceptual overview of PROV which includes three main types of objects: entity, agent, and activity as seen in Figure 5. PROV-O defines a set of Starting Point Classes, which are analogous to Dublin Core Terms—the key set of classes and predicates that can express the majority of applications, shown in Figure 6.

**Recommended Format**

By combining Dublin Core and PROV-O, we can achieve the desired collection-level metadata standard. The turtle representation for RDF is recommended but the format is syntax agnostic. For each collection, a prov:Collection should be defined that represents it. Under this collection there are five required fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dcterms:title</td>
<td>The title of the collection</td>
</tr>
<tr>
<td>dcterms:description</td>
<td>A brief description of the collection</td>
</tr>
<tr>
<td>prov:hadMember</td>
<td>A list of one or more tweets identifiable by a unique uri. (TweetID uri recommended)</td>
</tr>
<tr>
<td>prov:attributedTo</td>
<td>Minimally a single attributedTo entry mapping to a prov:Organization which describes the organization behind the collection. Multiple entries describing additional organizations or people (prov:Person) are optional.</td>
</tr>
<tr>
<td>prov:wasGeneratedBy</td>
<td>A prov:activity that describes how the collection was created. Further described below.</td>
</tr>
</tbody>
</table>

One or more prov:atLocation fields is also recommended for each location in the underlying tweets. Each location should be represented using ISO 3166-2 (International Organization for Standardization, 2012).
An activity should also be defined which precisely describes the process used to collect the tweets. It mandatorily includes prov:startedAtTime and prov:endedAtTime which are represented by the dateTime datatype from the W3 XMLSchema (Biron & Malhotra, 2004).

The standard requires that all of the above constraints be met; however, it does not limit the use of additional PROV-O Starting Point Classes or Dublin Core Terms. Currently, all tweet collections examined have used keyword queries as their source. A full example of a metadata representation in turtle syntax is presented in Figure 7.

**TweetID**

TweetID was the tool I created to support the unique identification of collections and tweets, the automatic standardization of tweet metadata, the extraction of new metadata fields from tweets and collections, the presentation of collections and tweets, and the merging of tweet collections. See figures 2-7 and the video, *Michael Shuffett – TweetID demo.mp4*, for an overview of TweetID. I made TweetID a web application to potentially support the tool being accessed by multiple organizations with zero setup required. The main challenge here was the amount of work required for implementation. Creating a polished and styled web application proved to be much more work than creating an equivalent one time script.

**Different File Formats**

One of the challenges presented in the implementation of TweetID was the fact that there were multiple heterogeneous input formats (see Figure 8 and Figure 9). The way I addressed this was to define a mapping between each format and the backend representation (which in turn maps to the standards I defined previously). Currently the extra metadata fields that were unexpected are being thrown away, but the design of the twitter-level standard allows for them to be stored under the organization’s namespace.
Collection Merging
Each field in the collection-level metadata needs to be merged when two collections are merged. However, each field might have unique aggregation logic. For example, the set of organizations needs to be unioned, the set of keywords needs to be unioned, the minimum start date should be taken, and the maximum end date needs to be taken. This is currently how it works; however, I am defining these aggregate operations in a more or less, ad-hoc manner. One of the visions I have for TweetID, should it undergo further development, is to allow for the definition of new metadata fields, including input mappings, persistence mappings, and aggregation operations in a declarative manner—through the use of meta-programming.
@prefix xsd:    <http://www.w3.org/2001/XMLSchema#> .
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix prov:    <http://www.w3.org/ns/prov#> .
@prefix tid:    <http://www.tweetid.org/tweet/> .
@prefix :        <http://example.com/> .

:vt-oklahoma-tornado
  a prov:Collection;
  dcterms:title "Oklahoma Tornado"
  dcterms:description "Tweets about Oklahoma Tornado";

  prov:hadMember
    tid:295450465339064321, tid:3954504653346069327, tid:5829504653346069829, ...;

  prov:wasAttributedTo :mark;
  prov:wasAttributedTo :vt;

  prov:atLocation "US OK" # ISO 3166-2

  prov:wasGeneratedBy :keyword-query;
  .

:mark a prov:Person, prov:Agent, prov:Entity .


:keyword-query
  a prov:Activity;
  prov:startedAtTime "2013-06-07T16:28:17Z"^^xsd:dateTime;
  prov:used :keyword-list;
  .

:keyword-list
  a prov:Collection;
  prov:hadMember
    "oklahoma tornado", "oklahoma storm", "#okc flood" ...;
  .

FIGURE 7: EXAMPLE COLLECTION LEVEL METADATA IN TURTLE SYNTAX
```json
{
    "archive_info": {
        "id": "84",
        "keyword": "oklahoma tornado",
        "description": ".",
        "tags": ",",
        "screen_name": "ctrnet",
        "user_id": "247438424",
        "count": "454458",
        "create_time": "1362804318"
    },
    "tweets": [
        {
            "archivesource": "twitter-search",
            "text": "Two killed by Tornado in Small Oklahoma Town - A tornado killed two people in... http://t.co/
            "so_user_id": "",
            "from_user": "GitterDoneNews",
            "id": "462844779617734656",
            "from_user_id": "1403574307",
            "iso_language_code": "en",
            "source": ""a href="http://www.ajaymatharu.com/"
            "profile_image_url": "http://abs.twimg.com/images/themes/theme1/bg.png",
            "geo_type": "",
            "geo_coordinates_0": "0",
            "geo_coordinates_1": "0",
            "created_at": "Sun May 04 06:42:49 +0000 2014",
            "time": "1399185769"
        },
        ...
    ]
}
```

**Figure 8: YourTrapperKeepr JSON File**

```
1 336620771126415361 Devastating Before-And-After Pictures Of One Of The Elementary Schools Wrecked In The Oklahoma Tornado: One of 3 2013-05-20 23:13:40 Mktg@theFuture 10000 10000 http://bit.ly/16vMXeA
3 336620771126415367 pray for Oklahoma 2013-05-20 23:13:40 angelafreer87 10000 10000
4 336620771243872256 Ray Manzarek dead of bile duct cancer. Gigantic tornado in OK. Asinine legal case in FL. I'm going to call for a do-over on today. 2013-05-20 23:13:40 galdusic 42.2774 -83.7333
5 336620771271395360 My thoughts & prayers with everyone in Moore oklahoma :( Sad day :) 2013-05-20 23:13:40
6 336620771449438864 RT @SteveBellybella: Prayers and lots of light for Oklahoma. 2013-05-20 23:13:40 SassysSmartGig
7 336620771629727744 All I want is a pain pill right now! This weather is stressing me out everywhere we go to avoid the tornado it goes that way! #notthanks 2013-05-20 23:13:40 SarahMaera65 10000 10000
9 336620771926464688 RT @PRRedCross: Concerned abt loved ones in #tornado damage area? Go to http://t.co/yN2yi7g9GK then click on SEARCH. 2013-05-20 23:13:40 kelsimons 10000 10000 http://Redcross.org/safeandwell
10 336620771945627117 " " 2013-05-20 23:13:40 " "
11 336620772135992034 For those asking about helping those hit in, #Oklahoma City to donate. Text "redcross" to 90999 or visit http://t.co/DQqOLk2uai 2013-05-20 23:13:40 robynd323 42.4485 -73.254 http://redcross.org
12 33662077272650400 RT @chicolenordeman: Oklahoma. HeartSick.
13 336620772959862352 Great way to start the week! On my way to Oklahoma 2013-05-20 23:13:40 A3_Nennigs
14 336620773016040568 @MichaelScolnik: 75 to 100 horses killed at Orr Farms in Moore, Oklahoma. (via @kfor) #tornado 2013-05-20 23:13:40
15 336620773178796032 RT @Harden13: Praying for everyone in Oklahoma City. This is crazy! 2013-05-20 23:13:40
16 336620773201269240 RT @GeriCanedo: Pray for Oklahoma. 2013-05-20 23:13:41 neetusander
```

**Figure 9: Tab Separated Value Input File**
Acknowledgements

Project Client – Mohamed Magdy (mmagdy@vt.edu)

Class Instructor/Project Advisor Dr. Edward Fox (fox@vt.edu)

NSF IIS - 1319578: Integrated Digital Event Archiving and Library (IDEAL)

Thanks to Carlos Castillo and Muhammad Imran for providing reference metadata.
References


Appendix A: Screenshots

See Figure 1 for a screenshot of the homepage.

**Figure 10: Upload Screen**
<table>
<thead>
<tr>
<th>ID</th>
<th>Created</th>
<th>Screen Name</th>
<th>Latitude</th>
<th>Longitude</th>
<th>URL Mentions</th>
</tr>
</thead>
<tbody>
<tr>
<td>461997779577888769</td>
<td>Thu May 01 22:37:08 +0000 2014</td>
<td>ChristianHollar</td>
<td>0</td>
<td>0</td>
<td>None</td>
</tr>
<tr>
<td>46199743475234817</td>
<td>Thu May 01 22:36:49 +0000 2014</td>
<td>KWiebs31</td>
<td>0</td>
<td>0</td>
<td>None</td>
</tr>
<tr>
<td>461996378734862336</td>
<td>Thu May 01 22:31:34 +0000 2014</td>
<td>OKReadiness</td>
<td>0</td>
<td>0</td>
<td>None</td>
</tr>
<tr>
<td>461996321788796929</td>
<td>Thu May 01 22:31:21 +0000 2014</td>
<td>IsalyWolf</td>
<td>0</td>
<td>0</td>
<td>None</td>
</tr>
<tr>
<td>46199630180643873638</td>
<td>Thu May 01 22:30:28 +0000 2014</td>
<td>NickReaderWBO</td>
<td>0</td>
<td>0</td>
<td>None</td>
</tr>
</tbody>
</table>
Queensland Floods
6030 total

<table>
<thead>
<tr>
<th>Organization</th>
<th>QCRI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collection Type</td>
<td>keyword</td>
</tr>
<tr>
<td>Keywords</td>
<td>#qldflood, #bigwet, queensland flood, australia flood, Missing, #qldfloods, #Bundaberg, queensland, #floods</td>
</tr>
<tr>
<td>Year</td>
<td>2013</td>
</tr>
<tr>
<td>Country</td>
<td>Australia</td>
</tr>
<tr>
<td>First Tweet Date</td>
<td>2013-01-27T08:37:34</td>
</tr>
<tr>
<td>Last Tweet Date</td>
<td>2013-02-07T23:55:57</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ID</th>
<th>Created</th>
<th>Screen Name</th>
<th>Latitude</th>
<th>Longitude</th>
<th>URL Mentions</th>
</tr>
</thead>
<tbody>
<tr>
<td>29546045539064321</td>
<td>2013-01-27 08:37:34</td>
<td>Fachrurozy__</td>
<td>10000</td>
<td>10000</td>
<td><a href="http://dlvr.it/2rv6PP">http://dlvr.it/2rv6PP</a></td>
</tr>
<tr>
<td>295451140710084608</td>
<td>2013-01-27 08:40:15</td>
<td>JaydenLestrange</td>
<td>-25.2744</td>
<td>133.775</td>
<td>None</td>
</tr>
<tr>
<td>295451317361602560</td>
<td>2013-01-27 08:40:57</td>
<td>ahammadafiz2010</td>
<td>10000</td>
<td>10000</td>
<td>None</td>
</tr>
<tr>
<td>295451408361209856</td>
<td>2013-01-27 08:41:19</td>
<td>geosurfnesc</td>
<td>10000</td>
<td>10000</td>
<td><a href="http://dlvr.it/2rv7Y0">http://dlvr.it/2rv7Y0</a></td>
</tr>
<tr>
<td>295451574178824192</td>
<td>2013-01-27 08:41:59</td>
<td>tiegan_brooks</td>
<td>10000</td>
<td>10000</td>
<td>None</td>
</tr>
<tr>
<td>295451761678782616</td>
<td>2013-01-27 08:42:43</td>
<td>RynannahGanney</td>
<td>-25.2744</td>
<td>133.775</td>
<td>None</td>
</tr>
<tr>
<td>29545182583953664</td>
<td>2013-01-27 08:42:55</td>
<td>Mellyjanee</td>
<td>-25.2744</td>
<td>133.775</td>
<td>None</td>
</tr>
<tr>
<td>295452249256239104</td>
<td>2013-01-27 08:44:40</td>
<td>MarcusForbes</td>
<td>-36.8474</td>
<td>174.766</td>
<td>None</td>
</tr>
<tr>
<td>295452270915645440</td>
<td>2013-01-27 08:44:45</td>
<td>AussieClaireC</td>
<td>-35.282</td>
<td>149.129</td>
<td>None</td>
</tr>
<tr>
<td>29545227695122432</td>
<td>2013-01-27 08:45:46</td>
<td>Switch1197</td>
<td>37.6873</td>
<td>-122.401</td>
<td><a href="http://fb.me/1gmrc468R">http://fb.me/1gmrc468R</a></td>
</tr>
<tr>
<td>295452848509030400</td>
<td>2013-01-27 08:47:02</td>
<td>amyldegnan</td>
<td>-37.8143</td>
<td>144.963</td>
<td>None</td>
</tr>
</tbody>
</table>

**Figure 12: Collection Details View**
Oklahoma Tornado and VT Oklahoma Tornado Merged
2022502 total

Organizations
OCR, Virginia Tech

Collection Type
keyword

Keywords
#okc, volunteer, #moore, #okhaves, #ok, poltunatomic, moore relief, #okc relief, moore disaster, shawnee, #okwx, oklahoma volunteer, #ok disaster, #ok relief, #okc, disaster, #ok, #ok storm, #ok flood, norman, oklahoma relief, oklahoma storm, tornado, oklahoma disaster, oklahoma tornado, moore storm, moore flood, moore tornado, #okneeds, #okc flood, #okc tornado, #okc storm, mary fallin, #ok tornado, moore volunteer, #ok volunteer

Year
2013

Country
US

First Tweet Date
2013-05-26T23:13:40

Last Tweet Date
2014-05-01T22:37:08

<table>
<thead>
<tr>
<th>ID</th>
<th>Created</th>
<th>Screen Name</th>
<th>Latitude</th>
<th>Longitude</th>
<th>URL Mentions</th>
</tr>
</thead>
<tbody>
<tr>
<td>336820770472112128</td>
<td>2013-06-20 23:13:40</td>
<td>meg_hurrell</td>
<td>38.8381</td>
<td>-104.821</td>
<td>None</td>
</tr>
<tr>
<td>336820771017359360</td>
<td>2013-06-20 23:13:40</td>
<td>vthompson2010</td>
<td>38.8816</td>
<td>-77.091</td>
<td>None</td>
</tr>
<tr>
<td>336820771042562387</td>
<td>2013-06-20 23:13:40</td>
<td>angelafraser87</td>
<td>10000</td>
<td>10000</td>
<td>None</td>
</tr>
<tr>
<td>336820771243872526</td>
<td>2013-06-20 23:13:40</td>
<td>gaiduric</td>
<td>42.2774</td>
<td>-89.7330</td>
<td>None</td>
</tr>
<tr>
<td>336820771449360864</td>
<td>2013-06-20 23:13:40</td>
<td>SassySmart5gRho</td>
<td>10000</td>
<td>10000</td>
<td>None</td>
</tr>
<tr>
<td>336820771629727744</td>
<td>2013-06-20 23:13:40</td>
<td>SarahMara95</td>
<td>10000</td>
<td>10000</td>
<td>None</td>
</tr>
<tr>
<td>336820771738760052</td>
<td>2013-06-20 23:13:40</td>
<td>Cococolin11</td>
<td>35.4866</td>
<td>-96.6859</td>
<td>None</td>
</tr>
<tr>
<td>336820771843661824</td>
<td>2013-06-20 23:13:40</td>
<td>kellicolleen</td>
<td>36.1333</td>
<td>-95.9756</td>
<td>None</td>
</tr>
<tr>
<td>336820772146627137</td>
<td>2013-06-20 23:13:40</td>
<td>rose_hisayo</td>
<td>10000</td>
<td>10000</td>
<td>None</td>
</tr>
</tbody>
</table>

**Figure 13: Merged Collection Details View**
## Tweets

2828612 total

<table>
<thead>
<tr>
<th>ID</th>
<th>Created</th>
<th>Screen Name</th>
<th>Latitude</th>
<th>Longitude</th>
<th>URL Mentions</th>
</tr>
</thead>
<tbody>
<tr>
<td>29545066539064321</td>
<td>2013-01-27 08:37:04</td>
<td>Fachrurozy__</td>
<td>10000</td>
<td>10000</td>
<td><a href="http://twitter.com/Fachrurozy__/">http://twitter.com/Fachrurozy__/</a></td>
</tr>
<tr>
<td>295451140710084808</td>
<td>2013-01-27 08:40:15</td>
<td>JaydenLstrange</td>
<td>-25.2744</td>
<td>133.775</td>
<td>None</td>
</tr>
<tr>
<td>295451317381602560</td>
<td>2013-01-27 08:40:57</td>
<td>ahmadhafiz2010</td>
<td>10000</td>
<td>10000</td>
<td>None</td>
</tr>
<tr>
<td>295451406361209856</td>
<td>2013-01-27 08:41:19</td>
<td>geosurfn\news</td>
<td>10000</td>
<td>10000</td>
<td><a href="http://twitter.com/GeosurfNews/">http://twitter.com/GeosurfNews/</a></td>
</tr>
<tr>
<td>295451574178824192</td>
<td>2013-01-27 08:41:59</td>
<td>tiegan_brooks</td>
<td>10000</td>
<td>10000</td>
<td>None</td>
</tr>
<tr>
<td>295451781678783616</td>
<td>2013-01-27 08:42:43</td>
<td>Ryhand\hCranney</td>
<td>-25.2744</td>
<td>133.775</td>
<td>None</td>
</tr>
<tr>
<td>295451812038535664</td>
<td>2013-01-27 08:42:56</td>
<td>Mellyjanee\ee</td>
<td>-25.2744</td>
<td>133.775</td>
<td>None</td>
</tr>
<tr>
<td>2954522492562259104</td>
<td>2013-01-27 08:44:40</td>
<td>MarcusForbes\e</td>
<td>-36.8474</td>
<td>174.766</td>
<td>None</td>
</tr>
<tr>
<td>29545270915648440</td>
<td>2013-01-27 08:44:46</td>
<td>AussieClaire\e</td>
<td>-35.282</td>
<td>148.129</td>
<td>None</td>
</tr>
<tr>
<td>295452527695122432</td>
<td>2013-01-27 08:45:46</td>
<td>Switcht197\e</td>
<td>37.6873</td>
<td>-122.401</td>
<td><a href="http://fb.me/1gmrC466R">http://fb.me/1gmrC466R</a></td>
</tr>
<tr>
<td>295452846878903040</td>
<td>2013-01-27 08:47:02</td>
<td>amydegnan\e</td>
<td>-37.8143</td>
<td>144.363</td>
<td>None</td>
</tr>
<tr>
<td>295454740510565816</td>
<td>2013-01-27 08:57:28</td>
<td>Joxro\x</td>
<td>-21.9113</td>
<td>-49.3653</td>
<td>None</td>
</tr>
<tr>
<td>29545486273269760</td>
<td>2013-01-27 08:57:31</td>
<td>WildWindTornado\e</td>
<td>10000</td>
<td>10000</td>
<td>None</td>
</tr>
<tr>
<td>295455581561247744</td>
<td>2013-01-27 08:59:06</td>
<td>CoastCath\e</td>
<td>-26.8754</td>
<td>153.05</td>
<td>None</td>
</tr>
<tr>
<td>295456085855622220</td>
<td>2013-01-27 08:59:12</td>
<td>client\ost\e</td>
<td>37.6873</td>
<td>-122.401</td>
<td>None</td>
</tr>
<tr>
<td>295456835918873285</td>
<td>2013-01-27 09:02:48</td>
<td>jacmos\s</td>
<td>10000</td>
<td>10000</td>
<td>None</td>
</tr>
<tr>
<td>295456835918388553</td>
<td>2013-01-27 09:02:53</td>
<td>Moyes_88</td>
<td>-31.9554</td>
<td>118.385</td>
<td>None</td>
</tr>
<tr>
<td>295469057858988580</td>
<td>2013-01-27 09:03:10</td>
<td>grant_mitchell</td>
<td>-37.8143</td>
<td>144.363</td>
<td>None</td>
</tr>
<tr>
<td>295467107846722112</td>
<td>2013-01-27 09:03:58</td>
<td>jessica_howe</td>
<td>-19.2581</td>
<td>146.318</td>
<td>None</td>
</tr>
</tbody>
</table>

**FIGURE 14: TWEETS SCREEN**
Tweet ID 295450465339064321

ID
295450465339064321

Created
2013-01-27 08:37:34

Screen Name
Fachrurrozy__

Latitude
10000

Longitude
10000

URL Mentions
http://dlvr.it/2rv5PP

Collections
Queensland Floods

JSON
{
  "twitter": {
    "entities": {
      "urls": [
        {
          "expanded_url": "http://dlvr.it/2rv5PP"
        }
      ],
      "created_at": "2013-01-27 08:37:34",
      "id_str": "295450465339064321",
      "coordinates": {
        "type": "Point",
        "coordinates": [
          10000.0,
          10000.0
        ]
      },
      "screen_name": "Fachrurrozy__"
    },
    "tweetid": {
      "collections": [
        "Queensland Floods"
      ]
    }
  }
}

Figure 15: Tweet Detail Screen