UNIVERSITY OF EXETER

User guide for Twitter project

James McMurray
June 26, 2013

1 REQUIRED SOFTWARE

All of the software used in the project is Free and Open Source Software, and is freely available for many platforms, from the Internet. The underlying software required is given in the following table:

Python 2.7	The software was written in the Python programming modules, using Twython for Python 2.7.	Included in Ubuntu by default, easily obtained for other operating systems.
Sqlite3	Sqlite3 was used to produce the databases.	This is included with Python by default.
Twython	The Twython module was used to obtain the data from the Twitter API.	This must be installed via the *pip* package management system for python. Running *sudo pip install twython* after installing *pip* via your package manager should install the module.
NetworkX	The NetworkX module was used to produce the GML files.	NetworkX can be installed via the *pip* package management system.
Gephi	Gephi was used to visualise the graphs (although other software could be used instead without any difficulty).	Gephi is available at `https://gephi.org/`.

2 DATABASE STRUCTURE

tweetsdb.db:

The database file *tweetsdb.db* contains the tables for the queries on *#climatechange, #climate, #globalwarming, #climaterealists,"Climate Change", "Global Warming", #agw.*

Tables: *htagw, htclimatechange, htglobalwarming, htclimate, ClimateChange, GlobalWarming, htclimaterealists*
Each table contains the tweets for that hashtag/query, all have the same structure.

Id	INT	Unique ID number of the tweet.
ScreenName	TEXT	User name of the posting user.
FullName	TEXT	Full name of the user (if given).
Tweet	TEXT	Text of the tweet (sometimes truncated due to URLs, etc. added in)
Timestamp	TEXT	Timestamp of the tweet in the form: "Mon Jun 17 22:38:21 +0000 2013"
RetweetCount	INT	Number of times this tweet has been retweeted.
InReplyToStatusId	INT	If this tweet is a reply to a status, returns the ID number of that status.
InReplyToUserId	INT	If this tweet is a reply, returns the ID of the user to whom the reply is.
Truncated	INT	Always returns 0.
Retweeted	INT	Always returns 0.
FriendsCount	INT	Returns the number of users that the tweeting user is following, at the time of collecting the data.
FollowersCount	INT	Returns the number of users that the are following the tweeting user, at the time of collecting the data.
IsRetweet	INT	Field added by myself. Boolean 1 or 0 - whether the tweet is a retweet or not by checking for presence of "RT: @X".
RetweetSource	TEXT	Field added by myself. Returns the username of the original source, if the tweet is a retweet, otherwise returns "-".
ConvertedTime	INT	Field added by myself. The timestamp converted to UNIX time.
RetweetTweet	TEXT	The text only of the Retweet (i.e. "RT: @X" removed) , if it is a retweet, otherwise "-".

userdb.db:

The user database *userdb.db* contains a table of descriptions for many users, a table of followers for many users - where multiple records for the source user construct a list of followers, a table of friends structured the same way as for followers, a table mapping the user ID number to the screen name of the user.

Tables: *descriptions, followers, friends, usermap*

descriptions:

ScreenName	TEXT	The user name of the user.
Description	TEXT	The description of the user.

followers:

ScreenName	TEXT	The user name of the user.
FollowerId	INT	The ID number of a follower of the user.

friends:

ScreenName	TEXT	The user name of the user.
FriendId	INT	The ID number of a friend (i.e. other user whom the user is following) of the user.

usermap:

ScreenName	TEXT	The user name of the user.
UserId	INT	The User ID number of the user.

IPCCdb.db:

The database file *IPCCdb.db* contains the tables for the queries on the hashtags *#AR5, #HadCRUT, #LTFchat, #Pages2k, #WGII, #GISS, #IPCC, #Pages, #UNFCCC, #WGIII*

Tables: *AR5, HadCRUT, LTFchat, Pages2k, WGII, GISS, IPCC, Pages, UNFCCC, WGIII*
Each table contains the tweets for that hashtag/query, all have the same structure. This is the same as for the *tweetsdb.db* file, except with the additional field UserId which stores the User ID number of the posting user directly (this field was accidentally omitted in the construction of the other database).

Id	INT	Unique ID number of the tweet.
ScreenName	TEXT	User name of the posting user.
UserId	INT	The User ID number of the posting user.
FullName	TEXT	Full name of the user (if given).
Tweet	TEXT	Text of the tweet (sometimes truncated due to URLs, etc. added in)
Timestamp	TEXT	Timestamp of the tweet in the form: "Mon Jun 17 22:38:21 +0000 2013"
RetweetCount	INT	Number of times this tweet has been retweeted.
InReplyToStatusId	INT	If this tweet is a reply to a status, returns the ID number of that status.
InReplyToUserId	INT	If this tweet is a reply, returns the ID of the user to whom the reply is.
Truncated	INT	Always returns 0.
Retweeted	INT	Always returns 0.
FriendsCount	INT	Returns the number of users that the tweeting user is following, at the time of collecting the data.
FollowersCount	INT	Returns the number of users that the are following the tweeting user, at the time of collecting the data.
IsRetweet	INT	Field added by myself. Boolean 1 or 0 - whether the tweet is a retweet or not by checking for presence of "RT: @X".
RetweetSource	TEXT	Field added by myself. Returns the username of the original source, if the tweet is a retweet, otherwise returns "-".
ConvertedTime	INT	Field added by myself. The timestamp converted to UNIX time.
RetweetTweet	TEXT	The text only of the Retweet (i.e. "RT: @X" removed) , if it is a retweet, otherwise "-".

3 TABLE OF SCRIPTS

dbgettweets#climatechange.py	Script to get tweets for #climatechange hashtag - writes to tweetsdb.db and userdb.db.
dbgettweetsClimateChange.py	Script to get tweets for "Climate Change" query - writes to tweetsdb.db and userdb.db.
dbgettweetsGlobalWarming.py	Script to get tweets for "Global Warming" query - writes to tweetsdb.db and userdb.db.
dbgettweets#climate.py	Script to get tweets for #climate hashtag - writes to tweetsdb.db and userdb.db.
dbgettweets#globalwarming.py	Script to get tweets for #globalwarming hashtag - writes to tweetsdb.db and userdb.db.
dbgettweets#agw.py	Script to get tweets for #agw hashtag - writes to tweetsdb.db and userdb.db.
dbgettweets#climaterealists.py	Script to get tweets for #climaterealists hashtag - writes to tweetsdb.db and userdb.db.
gvscript.py	Script to produce GraphView file for plotting timeline of retweets on a specific article.
classtweetreader.py	Class which handles some functions for analysis of data, mostly unused.
fillusermap.py	Script to get ID numbers for users and enter them in to the database, if missing.
retweettest.py	Script to build full retweet graph, inferring most likely intermediate sources from the list of followers and previous retweeters.
mantag2.py	Base script for manual tagging of users from user list as skeptic, activist, etc., requires creation of smaller database first.
cumtime.py	Script to create cumulative plots of users, tweets, and the tweet activity per day.
ipcctags.py	Script which was used to create new database for new hashtags data.
getipcctweets.py	Script which gets tweets for the IPCC related hashtags, writes to IPCCdb.db and userdb.db.
networkxstuff.py	Script to produce GML files for Friend-Follower graph, naive RT graph and mentions graphs.
classtweetgetter.py	Class which handles the obtaining of the data via Twython - hashtag queries, friend/follower queries, etc.
taggraphs.py	Adds colour tags to GML file from tag file.
removeweights.py	Removes edge weights from GML file.
stats.py	Calculates number of users, tweets, mentions, gini coefficient for tweets per user, etc.
extractusers.py	Extracts list of users from GML file.
mentionstag.py	Base script for tagging individual mention tweets - requires produced smaller database.
creatementiondb.py	Creates smaller database of data needed for mention tagging from user list.
createtempdb.py	Creates smaller database of data needed for user tagging from user list.
dbfs.py	Script for calculating path lengths, betweenness.
dbcronscript.sh	Script which runs all of the gettweets scripts.
igraphscript.R	Script to add hierarchical clusters to gml file using igraph in R - didn't work very well, but included in case it is needed in the future.
plotcp.R	Script for plotting cumulative plot graphs in R.