Codebook

v. 1.0, April 2014

---

1. Description of the study

In this study, the Twitter accounts of the National, University and Public Libraries in Germany were harvested via the Twitter API and Python code to gather information about how (larger) libraries in Germany make use of Twitter and how they interact with their network.

The study was conducted early 2014 (February: harvesting the Twitter handles, and April: harvesting the libraries' Twitter accounts, timelines and networks).

Author: Timo Glaser, timo.glaser@icloud.com, http://www.twitter.com/glaserti

Date: 04-2014

Source: This codebook, the Python code as well as the data files can be downloaded from the GitHub repo: https://github.com/glaserti/LibraryTwitter
The Codebook and the data files can be viewed and downloaded via Google Drive: http://goo.gl/X5ncYc

[1] Some part of the Python code was taken from Russell, Matthew A. *Mining the Social Web: Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More*. Second Edition. O'Reilly Media, 2013, cf. https://github.com/ptwobrussell/Mining-the-Social-Web-2nd-Edition for the code and for the license text.

1.1 Collecting the Tweeting Libraries

In a first step (02-2014), names and URLs of the homepages of the libraries were gathered via screen scraping from the corresponding websites of the German Library Statistics (DBS: Deutsche Bibliotheksstatistik: http://www.bibliotheksstatistik.de/eingabe/dynrep/adrbrowser/bibs.php).

Four different groups of libraries were chosen:

1. Deutscher Bibliotheksverband (DBV) Sektion 1 (German Library Association, Section 1): *Public libraries* from cities with a population of +400,000
2. Deutscher Bibliotheksverband (DBV) Sektion 2 (German Library Association, Section 2): *Public libraries* from cities with a population of +100,000
3. Deutscher Bibliotheksverband (DBV) Sektion 4 (German Library Association, Section 4): Academic libraries ("Wissenschaftliche Universalbibliotheken"), subsection: *University Libraries*
4. Deutscher Bibliotheksverband (DBV) Sektion 4 (German Library Association, Section 4): Academic libraries ("Wissenschaftliche Universalbibliotheken"), subsection: *National Libraries*

The data returned

- had to be cleaned manually by sorting out Twitter accounts which did not represent the library (e.g. the Twitter handle of the city or the university was frequently mentioned on the homepage of the library in case the library didn't have an own Twitter account)
- had to be complemented manually: some libraries, though having a Twitter account, did not mention their Twitter screen name on their homepage; sometimes the Twitter account is only mentioned on a subpage (e.g. Contact, PR, …), sometimes it isn't mentioned at all and could only be found via web searches.

The cleaned and complemented data was saved to one file for each of the three library groups:

1. Public Libraries (combining Section 1 and 2 of the DBV)
2. University Libraries
3. National Libraries

1.2 Harvesting Twitter Data

In 04-2014, the accounts, timelines and networks of the libraries' Twitter accounts were harvested and analyzed.

1. An API request for each of the accounts was send to gather general information about the account (e.g. when was the account created, number of tweets, number of friends and followers etc.)

2. An API request for each of the accounts was send to gather information about the network of the library, i.e. the ids of the friends and followers as well as their screen_names, descriptions, locations. Based on the information given in the description and the location, the friends and followers of each library were clustered. Only this clustered information is provided with the data in this study.

3. An API request for each of the accounts was send to harvest the tweet archive (on Monday, 04-06-2014). The archive of each library was saved as a json-file. Due to the Twitter API restrictions, only the last 3,200 tweets of each timeline could be gathered. For most of the libraries, this meant that the entire archive could be downloaded. However, there were a few libraries with more than 3,200 tweets. (The slight differences between the number of tweets and the actual tweets in the archive can result from deleted, no longer accessible tweets.) These raw data files are not part of the data provided with this study.

4. Finally, information was gathered via an API request concerning the libraries' tweets which were answers to other users' tweets. The identity of the Twitter user to whom the library responded was gathered and analyzed as well as the original tweet of this user.

Based on these API requests, the data was filtered, summarized and saved in csv files.

2. Technical Information about the Files

There are six csv files, one set of files for the library account, one set of files for the timeline; in each set there is one file for each library group (see 1.):

1. library account statistics, 45 variables, 51 observations (i.e. libraries)[2]
   - 1.1. NatBib_libTwitterStats_2014-04-09.csv: 3 observations
   - 1.2. UniBib_libTwitterStats_2014-04-09.csv: 27 observations
   - 1.3. OeBib_libTwitterStats_2014-04-09.csv: 21 observations

2. timeline statistics, 22 variables, 59,405 observations (i.e. tweets)
   - 2.1. NatBib_timelineStats_2014-04-09.csv: 2,560 observations
   - 2.2. UniBib_timelineStats_2014-04-09.csv: 23,905 observations
   - 2.3. OeBib_timelineStats_2014-04-09.csv: 32,940 observations

[2] In the Google Drive Fusion Tables version of the data, the three files for the library account statistics were merged into one single spreadsheet "GermanLibraryTwitterStats".

3. Structure of the Data[3]

3.1 NatBib_libTwitterStats Files

screen_name	string		The screen_name of the Twitter account as it is returned via the Twitter API, cf. https://dev.twitter.com/docs/platform-objects/users
id_str	string		The ID of the Twitter account as a string as it is returned via the Twitter API, cf. https://dev.twitter.com/docs/platform-objects/users
location	string		The location of the library. This information is retrieved from the DBS page, cleaned and normalized (e.g. Frankfurt/Main -> frankfurt)
created_at	date	Fri Apr 05 16:39:27 +0000 2013	The date the account was created as it is returned via the Twitter API, cf. https://dev.twitter.com/docs/platform-objects/users
created_at_sec	integer		The date converted to seconds since 1970-01-01 00:00:00
days	integer		The number of days since the account was activated
statuses_count	integer	0, 1 …	Number of tweets as it is returned via the Twitter API, cf. https://dev.twitter.com/docs/platform-objects/users
days_since_last_tweet	integer	0, 1, …	Number of days since last tweet was sent
tweets_per_day	float	0, 0.49, …	statuses_count divided by days (since activating the account).
tweets_per_year	float		statuses_count divided by years since activating the account.

[3] Due to the structure of Python dictionaries, the order of the columns in the files is not sorted.

genuine_tweet_ratio	float		Number of genuine tweets / number of tweets in the archive (i.e. including auto tweets but not retweets)
auto_tweet_ratio	float		Number of auto tweets / number of tweets in the archive
avg_RTs	float		Number of retweets / number of genuine tweets in the archive
avg_Favs	float		Number of favorited tweets / number of genuine tweets in the archive
avg_has_mention	float		Number of tweets with @mention / number of genuine tweets in the archive
avg_has_url	float		Number of tweets with url / number of genuine tweets in the archive
avg_has_hashtag	float		Number of tweets with hashtag / number of genuine tweets in the archive
avg_has_media	float		Number of tweets with embedded media / number of genuine tweets in the archive
avg_has_entity	float		Number of tweets with any of the above mentioned entities (mention, url, hashtag, media) / number of genuine tweets in the archive
replies_ratio	float	0, 0.01, …	Number of replying tweets / number of genuine tweets in the archive
avg_hours_to_answer	float	0, 0.3, 1.1, …	Sum of hours_to_answer / number of replying (not orphaned) tweets
avg_resonanceFactor	float	0, 0.01, …	Sum of resonance_Factor / number of genuine tweets in the archive
friends_count	integer		Number of “friends” as it is returned via the Twitter API, cf. https://dev.twitter.com/docs/platform-objects/users
friend_local	integer		Number of “friends” from the same location as the library
friend_other_place	integer		Number of “friends” with another location than the location of the library

friend_without_place	integer		Number of “friends” who haven't specified a location in their Twitter profile
friend_librarian	integer		Number of “friends” who can be identified as librarians
friend_library	integer		Number of “friends” who can be identified as a library
friend_publisher	integer		Number of “friends” who can be identified as working in the publishing business
friend_varia	integer		Number of “friends” who can’t be sorted into one of the clusters
friend_without_ description	integer		Number of “friends” who haven't given any description in their Twitter profile
followers_count	integer		Number of “followers” as it is returned via the Twitter API, cf. https://dev.twitter.com/docs/platform-objects/users
follower_local	integer		Number of “followers” from the same location as the library
follower_other_place	integer		Number of “followers” with another location than the location of the library
follower_without_place	integer		Number of “followers” who haven't specified a location in their Twitter profile
follower_librarian	integer		Number of “followers” who can be identified as librarians
follower_library	integer		Number of “followers” who can be identified as a library
follower_publisher	integer		Number of “followers” who can be identified as working in the publishing business
follower_varia	integer		Number of “followers” who can’t be sorted into one of the clusters
follower_without_ description	integer		Number of “followers” who haven't given any description in their Twitter profile
followBackCount	integer		Number of “followers” who are also “friends”

XFollowsNotBack	integer		Number of “followers” who the library does not follow back (who are not “friends”)
activeNotFollowRatio	float		The ratio of the library not following back their followers: XFollowsNotBack / followers_count
NotFollowXBack	integer		Number of “friends” who do not follow back the library (who are not “followers”)
passiveNotFollow Ratio	float		The ratio of the “friends” of the library who are not following back the library: NotFollowXBack / friends_count

3.2 NatBib_timelineStats Files

screen_name	string		The screen_name of the Twitter account as it is returned via the Twitter API, cf. https://dev.twitter.com/docs/platform-objects/tweets
id_str	string		The ID of the tweet as a string as it is returned via the Twitter API, cf. https://dev.twitter.com/docs/platform-objects/tweets
genuine_tweet	integer	1 0	Is it a tweet of the library (1: yes) or is it a retweet (0: retweet)?
auto_tweet	integer	1 0	Is the tweet sent via a bot? 1: yes, 0: no (i.e. was sent probably manually)
created_at	date	Fri Apr 05 16:39:27 +0000 2013	The date as it is returned via the Twitter API cf. https://dev.twitter.com/docs/platform-objects/tweets
tweet_time	integer	0, 1, 2, … 23	Hour of day a tweet was sent.
tweet_weekday	integer	1, … 7	Day of the week a tweet was send (1 = Monday, 7 = Sunday)
tweet_month	integer	1, … 12	Month a tweet was send (1 = January, 12 = December)

has_mention	integer	NA 1 0	Does the tweet mention another account (@mention)? 1 = yes, 0 = no. If the tweet is a retweet: NA
has_hashtag	integer	NA 1 0	Does the tweet use a hashtag? 1 = yes, 0 = no. If the tweet is a retweet: NA
has_url	integer	NA 1 0	Does the tweet entail a link? 1 = yes, 0 = no. If the tweet is a retweet: NA
has_media	integer	NA 1 0	Is there a media file embedded in this tweet (in most cases a photo, a link to a youtube video does not count as embedded media)? 1 = yes, 0 = no. If the tweet is an retweet: NA
is_reply	integer	1 0	Is this tweet an answer to another tweet? 1 = yes, 0 = no
orphan	integer	NA 1 0	If is_reply = 1, the original tweet sometimes is no longer accessible for analysis. If this is the case: 1 (yes), if the original tweet could be analyzed: 0 (no). If is_reply = 0: NA
is_follower	string	NA 0 nonprof prof	If is_reply = 1 and orphan = 0: Is the author of the original tweet a follower and if so, from which cluster: professional (i.e. librarian, library, publisher), otherwise: nonprofessional. If s/he is not a follower: 0. If the tweet is not a reply or an orphan: NA
follower_local	integer	NA 1 0	If is_reply = 1 and orphan = 0: Is the author of the original tweet a follower and is he from the same place as the library?
original_is_question	integer	NA 1 0 -	If is_reply = 0: NA If is_reply = 1 and orphan = 0: Has the text of the original tweet a “?”? If so, it can be considered a question (1), else (0). If orphan = 1: “-”
reply_is_answer	integer	NA 1 0	If is_reply = 0: NA If is_reply = 1 and orphan = 0:

		-	Does the original tweet address the library directly (@mention)? If so, the reply can be considered an direct answer (1), else: 0 If orphan = 1: "-"
hours_to_answer	integer	NA 0, 0.1, … -	If is_reply = 0: NA If is_reply = 1 and orphan = 0: How long did it take to reply to the original tweet (calculated the time differences of the two tweet's 'created_at' variable) in hours (e.g. 1.1 hrs = 66 min) If orphan = 1: "-"
favorite_count	integer	NA 0, 1, …	If the tweet is an auto_tweet: NA If auto_tweet = 0: Number of times the tweet was favorited by other users.
rt_count	integer	NA 0, 1, …	If the tweet is an auto_tweet: NA If auto_tweet = 0: Number of times the tweet was favorited by other users.
resonance_Factor	float	NA 0, 0.5, 1, …	If the tweet is an auto_tweet: NA If auto_tweet = 0: The resonance factor is calculated: 1.0 x rt_count + 0.5 x favorite_count

3.4 Explanation of the Description Clustering

According to the information given in the description, the "friends" and "followers" were clustered by occurrence of the following character strings in descending, exclusive order (an account with the description "Librarian working at the library…" was only clustered to "librarian", and no longer considered for the category "library").

librarian: *'bibliothekar'*, *'librarian'*, *'fachrefer'*, *'malis'*
('fachref' = Fachreferent/in [subject specialist], 'malis' = Master of Library and Information Science).

library: *'ub'*, *'bib'*, *'librar'*, *'ULB'*, *'cherei'*, *'stabi'*, *'archiv'*, *'museum'*, *'vifa'*, *'webis'*
(to avoid character encoding issues (unicode, latin1, e.a.), instead of looking for bucherei, the keyword was reduced to "cherei", knowing that this can add some noise to this cluster; 'archiv' and 'museum' were added, since they are related areas, as well as 'vifa' and 'webis').

publisher: *'buch'*, *'verleg'*, *'verlag'*, *'book'*, *'publish'*, *'medien'*, *'media'*, *'autor'*, *'author'*, *'redaktion'*, *'zeitung'*, *'press'*
(This category is even noisier than the other two categories: a “book enthusiast” or a “social media addict” would be considered a “publisher”.

3.5 Explanation of the Auto Tweet Category

The Twitter API returns a “source” field
(cf. https://dev.twitter.com/docs/platform-objects/tweets)

If in this field one of the following keywords were mentioned, the tweet was categorized as an automatically sent tweet:

'twitterfeed', 'wp.com', 'wp-to-twitter', 'tumblr', 'instagram', 'blogs.', 'SharePress', 'facebook', 'studivz', 'paper.li'

These keywords were chosen after analyzing the source field of a sample of tweets from all three categories.