Digital data collection – pilot with the KOAIKA pineapple cooperative in Rwanda. Photo Credit: Sarah Davies/Oxfam

GOING DIGITAL

Web data collection using Twitter as an example

These guidelines are published under the Going Digital series where we explore and share our learning in using new digital technologies in evaluation and research.

In this fifth instalment of Going Digital, another form of digital data is explored: web data. The paper discusses how web data, in particular social media data, can be collected and provides hands-on guidelines for harvesting Twitter data.

OXFAM

CONTENTS

Four years on from the Going Digital: Using digital technology to conduct Oxfam's Effectiveness Reviews pilots, where learning on the added value of using digital technology to conduct our impact evaluations was shared, Oxfam continues to develop its methods of data collection through the use of digital technology. The next paper in this series, Going Digital: Using and sharing real-time data during fieldwork, demonstrated how sharing real-time data during fieldwork can increase engagement and participation in communities and improve integration between different data-collection techniques. The third paper of the series, published in 2019, Going Digital: Improving data quality with digital data collection, presented and discussed features to improve data quality and accuracy enabled through the use of digital data-collection techniques. The fourth paper, Going Digital: Privacy and data security under GDPR for quantitative impact evaluation gave an overview of privacy and data security concerns from a rights-based perspective within the context of the General Data Protection Regulation (GDPR). This fifth paper focuses on another form of digital data: web data. It discusses how web data, in particular social media data, can be collected and provides hands-on guidelines for harvesting Twitter data.

1 INTRODUCTION

In 2015, Oxfam GB's team of Impact Evaluation Advisers started using digital devices to conduct surveys for Oxfam's Effectiveness Reviews[1] (a series of impact evaluations conducted on a random sample of mature projects). Using digital devices has led to improvements in data security, accountability, accuracy and timing, while also reducing costs. Making the transition from pen-and-paper surveys to a digital process, and the benefits and considerations connected to this, have been documented and shared in the Going Digital series: Going Digital: Using digital technology to conduct Oxfam's Effectiveness Reviews pilots (Tomkys and Lombardini, 2015), Going Digital: Using and sharing real-time data during fieldwork (Lombardini and Tomkys Valteri, 2017), Going Digital: Improving data quality with digital data collection (Lombardini, Pretari and Tomkys Valteri, 2018) and Going Digital: Privacy and data security under GDPR for quantitative impact evaluation (Vonk, 2019).

While the previous papers of this series have mostly focused on collecting survey data in a digital manner, this document discusses another form of digital data: web data, and in particular social media data. Online activity has become increasingly important in many areas of life. Social media specifically has the potential to profoundly impact the way society works, giving new opportunities for connecting with others around the globe, for sharing ideas and for communicating news. Using social media data for research allows the gathering of topical, naturally occurring information in real-time and has proven to be a valuable source of data (see, for example, Mayer-Schönberger and Cukier, 2013), and the evaluation context is no exception to this (see, for example, Abreu Lopes, Bailur and Barton-Owen, 2018). Even though social media data offers many new opportunities for data collection and data analysis, researchers also need to take into consideration the specific limitations that come with it. It is important to reflect on issues of limited access and inequal inclusion, as well as the opportunities of spreading misinformation in the online landscape (see, for example, Shao *et al*., 2018).

These guidelines provide a hands-on tutorial on how to collect web data and provide detailed example code on how to collect Twitter data. This methodology has previously been used as part of an impact evaluation of a project in Tanzania: Active Citizenship in Tanzania: Impact Evaluation of the 'Governance and Accountability through Digitalization' project (Pretari, 2019). This project aimed to improve community-driven governance and accountability through the use of digital technology. As part of the project, involved citizens, community animators, and other stakeholders generated Twitter data which was then collected and analysed (see Schwitter and Liebe, 2019 for details). This enabled the identification of engagement with online platforms; a better understanding of online behaviours, mobilization and interactions; and contributed to the assessment of the impact of the project. The analysis of Twitter data was shown to be a valuable complement to the qualitative case studies and the quantitative counterfactual approach. Some exemplary analyses can be found in the chapter Outlook - Data Analysis in these guidelines and more can be found in the full report.

There are a number of different approaches and tools available to extract and analyse data from social media channels and other websites. The following guide will give an introduction to the collection and analysis of Twitter as it is one of the most popular social media sites with the least restrictions regarding data accessibility. It will use the software *R* (Version 3.5.2) and the library *rtweet* (Version 0.6.8.9000), which is the most up-to-date and best maintained library for collecting Twitter data with R. Some other

packages are used for certain calculations; this is indicated in the specific code snippets. The collection of data on social media sites other than Twitter follows roughly the same procedure, but different sites employ different limitations on the scope of the available data. All explanations and restrictions mentioned in the following refer to the standard, free data access that Twitter grants to the public (premium and enterprise solutions are available for a monthly cost). These guidelines are aimed at development practitioners, technical researchers and students with an interest in the methodology of social media data collection and assumes basic knowledge of R, such as familiarity with its basic syntax and operators. For absolute beginners of R, it might be worthwhile to start with an introduction to R (for more on this, see, for example, RStudio's introductory references[2]).

2 ETHICAL CONSIDERATIONS

Research ethics are at the core of any undertaking of research projects. Collecting data from social media sites is no exception; on the contrary, it poses some new challenges that need to be considered and discussed (for a broad discussion of ethical considerations in web data research see, for example, Salganik, 2017, Chap. 6).

While often only information that is publicly available is accessed when collecting data from the web – such as from public Twitter profiles -– one has to keep in mind that social media research can include the improper access to personal information. Users of social media sites make their data available for the purpose of social networking, not necessarily to be harvested and used for research purposes. Therefore, whenever possible, it is advisable to receive consent to avoid unauthorized secondary data uses (Zimmer, 2010). Yet obtaining consent is often unrealistic, particularly in large-scale projects. For linking web data and survey data, informed consent is a prerequisite. But consent rates can be rather low (e.g. between 25% and 49% for studies in the UK, see Al Baghal *et al*., 2019) which results in low case numbers for data analyses. Still, this linked data can provide useful insights for contextualizing the web or social media data analyses. While, to our knowledge, there are no 'strict general rules' regarding ethical standards of social media research, it seems important to consider at least the following points (see Townsend and Wallace, 2016):

- *The terms and conditions of social media sites*: researchers should familiarize themselves with the terms and conditions of the social media platforms they plan to use and to evaluate whether they can agree with them; often, this means agreeing to an additional and/or different set of terms and conditions when using social media sites as a researcher as opposed to a user.
- *The ethical guidelines of the researcher's institution*: researchers need to evaluate whether their planned research as well as the terms and conditions of social media sites are in line with ethical regulations of their research intuitions or professional associations, e.g. British Sociological Association, American Association for Public Opinion Research, etc., and, if needed, seek the relevant approval of ethics boards.
- *Whether social media users can expect to be observed by strangers*: this is a crucial aspect, and data from public and open sites, such as public Twitter profiles, present fewer ethical issues than those from private sites for which access needs registration with the site and/or approval by the other user; users of the latter need to be contacted for consent in any case.
- *Whether social media users are vulnerable*: potential harm and risk of the planned research should be assessed especially carefully if the data refer to vulnerable individuals, such as children; in this case informed consent would be a strict requirement. Also, study participants should be given information about where they can receive help in case there are problems related to the research.
- *Whether the research topic is sensitive*: if the topic refers to sensitive issues, such as taboos in a society, criminal behaviour and social norm violations, the social media data analysis can be harmful for the corresponding social media users; the corresponding risks need to be considered and

evaluated.

- *Whether social media users are anonymized in published outputs*: typically, all users should be anonymized to the greatest extent possible in research output, except where there is informed consent on naming users, or the research concerns public figures or organizations who aim to share their information as widely as possible. Anonymization of publicly available data is a thoughtful undertaking – the question of how to make data untraceable needs to be critically reflected, i.e. one should not use any direct quotes and only paraphrase, etc.
- *Whether the dataset can be published or shared*: sharing data is important for the replication of research findings and is increasingly demanded by the research community. Whether social media data can be shared might depend on the terms and conditions of social media sites, which need to be checked, as well as the sensitivity of the research topic. If data cannot be shared the reason should be provided.

3 COLLECTING WEB DATA

There are two general approaches to collecting data from the web: Web scraping via the browser or using the application programming interface (API) some websites offer.

3.1 WEB SCRAPING

Web scraping refers to the general extraction of data from the surface of websites. Any content that can be viewed on a web page can be scraped since web scraping pulls data directly from the HTML (Hypertext Markup Language). HTML is the standard language for creating web pages and web applications. Web browsers receive HTML documents from a server following a request and render the documents into multimedia web pages. HTML describes the structure of a web page semantically. HTML pages are made up of HTML elements.

Web scraping thus relies on the structure of websites and needs to know where exactly information on the website can be found to extract it. Information can be identified through the HTML elements and their ID and other supporting technologies (such as using regular expressions to parse text or relying on other structural information as identified through CSS [Cascading Style Sheet] selectors, the XML [Extensible Markup Language] path [XPath] or others).

Since web scraping relies on the HTML view of a website, it is in general susceptible to redesigns of websites: If web developers change the design of websites, the whole script may break down and needs to be rewritten. Furthermore, many companies block web scraping of their websites. They implement methods that detect and disallow bots from viewing their page. Nowadays, this is the case for most of social media sites. They instantly block any account that they identify as accessing their websites in an automated fashion. Data from social media sites, such as Twitter, thus has to be collected through their APIs.

3.2 USING APPLICATION PROGRAMMING INTERFACES

An API is a set of defined methods of communication among software components. In the case of web APIs, this refers to methods of communication between the user of the site and the website itself. While web browsers render the visual content, APIs manage and organize data. Data requests are made to the API and responses are given in file formats such as JSON (JavaScript Object Notation) or XML

instead of rendered multimedia pages. APIs thus work as structured data feeds by accessing database contents directly, and they allow developers to access core data and methods for interaction. Which data are available and what kind of methods can be used depend on the specific API and is usually outlined in the particular API documentation.

Generally, there is only restricted access to a website's data via the APIs. Many companies employ a rate limitation of the number of queries that can be sent in a certain time frame so that users have to wait before sending additional requests (to not slow down the server). Furthermore, some data cannot be accessed at all via an API or might be restricted to paying users. When using APIs, the website being harvested has complete control over what information to give and what data to withhold. In contrast to web scraping, which happens through the anonymity of a web browser, APIs often require users to authenticate themselves to be granted access.

Even though these restrictions exist, APIs should be used when they are available: they are more functional, stable and consistent than web-scraping approaches that rely on the web view. However, while APIs are becoming more popular, they do not exist for every website as it is a challenging and resourceful investment to develop and maintain, which most websites lack the incentive to do. Personal websites, smaller ecommerce sites and many others do not have APIs available and need to be web scraped in the traditional way.

4 COLLECTING SOCIAL MEDIA DATA

It is possible to collect all kinds of web data, from information displayed on Wikipedia to the most recent train times shared on the website of national rail companies. Social media sites, and Twitter in particular, are especially interesting in terms of social interaction and will be used as the leading example in the following. An introduction to Twitter and its vocabulary will be given in the first section, while the available APIs and how to access them will be explained subsequently.

4.1 INTRODUCTION TO TWITTER

Twitter is a real-time social networking and information service, launched in 2006, on which users share character limited messages on their personalized, half-public news feed. The length of these messages, so-called *tweets*, was originally restricted to 140 characters; the limit was doubled in November 2017 and currently, Twitter also allows the combination of multiple tweets into one longer *thread* and the sharing of non-textual multimedia files, such as pictures and videos. Other users can subscribe to a user's newsfeed to receive new messages from them automatically; they then are a *follower* of a user. In contrast to some other popular social media websites, the follower–followee relationship does not need to be reciprocal, and it is possible to have many followers while only following few (this is, for example, the case with celebrities). Furthermore, access to most Twitter profiles is public and does not require registering with Twitter, although only registered users can directly interact with tweets and with each other. These features have led to Twitter becoming more of a microblogging platform than a pure social media networking site.

While in the early days of Twitter the exchange of information was the main focus, further features for social interactions have been developed since. It is now possible to forward tweets (*retweeting*) or *reply* to them, to *like* tweets (so they become *favourites*), to explicitly name and connect with other users in a message (*mentioning*), share pictures, links and other multimedia data, as well as to send private message to single individuals or groups (*direct messaging*). Using hash signs (#) before words is a popular mechanism to tag tweets with that specific word, turning them into *hashtags*. Hashtags are linked and can be searched for. Many of these features began as user-driven innovations that were first

implemented by third-party software developers and only later adopted by Twitter itself (Driscoll and Walker, 2014).

Over the past decade, Twitter has grown into a widespread channel of communication. It is being used in various ways, for example to interact with potential voters by political institutions (Grant, Moon and Busby Grant, 2010), to advertise new products and for promotions by companies (Mamic and Almaraz, 2013) or as a channel of communication between people known or unknown to each other to share a common (topic of) interest, both in everyday lives or during political crises (Boyd, Golder and Lotan, 2010; Lotan *et al*., 2011). All in all, the original intended purpose of Twitter – distribution of information – is still an important reason for Twitter usage (Parmelee and Bichard, 2011).

As of September 2019, Twitter comprises 330 million monthly active users that post 500 million tweets per day.

4.2 ACCESSING TWITTER DATA

Twitter offers two different families of APIs to allow the public to access their data, the *Streaming API* and the *REST API* (Representational State Transfer). They differ in regard to the functionality they offer and the ways they constrain a user. Which one has to use is a matter of research interest.

To access any one of them, certain prerequisites have to be fulfilled.

4.2.1 PREREQUISITES

To gain access to Twitter data, it is necessary to register a developer account. Any Twitter user can apply for a developer account.[3] Application requires a verified phone number and email address, as well as a detailed description of how the API will be used. Use of the Twitter API requires agreeing to Twitter's Developer Agreement and Policy, as well as their related policies, including the Display Requirements and Automation Rules. These agreements are in place to ensure a reasonable usage of the Twitter API and the data shared. Once a developer account has been established, a developer's Twitter application[4] needs to be registered by providing a name, description and domain. This application authenticates the end user for the application of the Twitter API and gives the user the necessary access key and access token through the app management dashboard.

The following script shows the typical authentication process in R (using the library rtweet[5] which is loaded in first). If the package has not yet been installed, it needs to be installed by calling `install.packages("rtweet")` for the current released rtweet from CRAN (Comprehensive R Archive Network).

```
## load rtweet package
library(rtweet)

## create authentication token
create_token(
  app = "<YOUR APP NAME>",
  consumer_key = "<YOUR CONSUMER API KEY>",
  consumer_secret = "<YOUR CONSUMER API SECRET KEY>",
  access_token = "<YOUR ACCESS TOKEN>",
  access_secret = "<YOUR ACCESS TOKEN SECRET>")
```

The consumer key, consumer secret key, access token and access token secret (as given to a user by Twitter) are passed to the function `create_token()`, which generates an authorization token and saves it as an environment variable. This has to be done once in the beginning of the script, then the token is

used to access the APIs.

4.2.2 STREAMING API

Through the Streaming API, Twitter data can be accessed as a constant real-time data stream. It gives access to (a sample of) all tweets as they are published on Twitter. After sending a certain request to the API, specific data will be sent to the user continuously. The Streaming API only sends out real-time tweets.

These streaming APIs need a search term to filter the results, such as a hashtag (e.g. #oxfam), a specific user ID (e.g. 15821039, the user ID of oxfamgb[6] or a geographical area defined by coordinates. Public statuses that match one or more of the filter predicates are returned. Multiple parameters can be specified; the standard access level allows up to 400 track keywords, 5,000 follow user IDs and 25 location boxes.

The following code snippet retrieves all public tweets that include the hashtags #oxfam or #poverty for the next five seconds. The streaming API is connected to the public streams, so all public data on Twitter is accessed.

```
## stream tweets using the hashtags
tweetshashtags <- stream_tweets("#oxfam,#poverty", timeout = 5)

## Streaming tweets for 5 seconds...

## Finished streaming tweets!
```

While the previous request streamed tweets containing certain hashtags, the following tracks a user by passing the user IDs to `stream_tweets()`. All tweets posted by a certain user – oxfamgb – for the next five seconds are streamed.

```
## stream tweets using user ID
tweetsoxfam <- stream_tweets(15821039, timeout = 5)

## Streaming tweets for 5 seconds...

## Finished streaming tweets!
```

The first parameter passed to `stream_tweets()` specifies the query which is used to select tweets. An empty query ("") returns a small random sample of all publicly available Twitter statuses. To filter by keywords or by geo-location or to track users, lists and vectors need to be provided. The variable `timeout` specifies for how many seconds the connection should be left open while capturing the tweets. It can be set to `FALSE` to stream indefinitely.

If the total result of the search term is larger than one percent of the current Twitter volume, only a sample of the result will be returned. This can be the case if major events happen that govern discussions on Twitter, such as mass catastrophes, i.e. high-volume events. How this sample is chosen is not known, but research suggests it is not a random sample but depends on popularity of a tweet (Driscoll and Walker, 2014). Specific search terms then help in guaranteeing completeness of the results.

The streaming API is appropriate when a large number of tweets should continuously be collected over a large period of time. A stable internet connection is necessary for the whole amount of time, since tweets are requested and sent back continuously in real-time. Any interruption of the script leads to a loss of data. Furthermore, the reaction towards tweets (such as likes and retweets) cannot be collected, as tweets are sent to the streaming API as soon as they have been published. It therefore captures what

users are saying, but there is no indication of how popular the content is.

4.2.3 REST API(S)

Besides the streaming API, Twitter provides REST APIs. REST (Representational State Transfer), is a widespread software architectural style that defines a set of constraints to be used for creating web services. These services allow the requesting systems to access and manipulate web resources by using a uniform and predefined set of stateless operations. In case of the Twitter APIs, two main functionalities are provided: *Getting* data *from* Twitter and *posting* data *to* Twitter (such as tweeting with your own account). These guidelines focus on getting data from Twitter by retrieving data through the *Search API* (searching for tweets containing certain words, using specific hashtags, etc.) and the *User API* (collecting a user's tweets, followers, etc.).

In contrast to the streaming API, which collects real-time data in an ongoing fashion, REST APIs are suitable for single searches of historical data. Two use cases will be presented: searching historic tweets and reading user profile information.

Historic tweets

To collect tweets, the search API is most suitable. The search API requires certain search terms and returns matching results that date back up to about a week. Furthermore, rate limits have to be considered. The API returns a maximum of 100 tweets per request with a limit of 180 requests in 15 minutes.

The following code collects the *n* most recent tweets out of all public tweets in English that include both the hashtags #oxfam and #poverty.

```
## find English tweets using the hashtags
povertyhashtag <- search_tweets("lang:en #oxfam #poverty", n = 200,
include_rts = FALSE)

## Searching for tweets...

## Finished collecting tweets!

povertyhashtagRTS <- search_tweets("lang:en #oxfam #poverty", n = 200,
include_rts = TRUE)

## Searching for tweets...
## Finished collecting tweets!
```

The first argument of search_tweets() is the query which is looked up. It is used to filter and select tweets from the REST API. It must be a character string. Spaces behave like Boolean *AND* operators; *OR* can be noted as OR. n specifies the total number of desired tweets to be returned. It defaults to 100 while the maximum from a single token is 18,000. To return more than 18,000 tweets, retryonratelimit should be set to TRUE, which forces the script to pause when reaching the rate limit and retrying automatically after waiting. Finally, include_rts is a logical that indicates whether to include retweets in the search results.

In this example, all tweets of the past 6–9 days that match the search query are returned after a single request has been made. This is in contrast to the streaming API, which will return all tweets from the point of the execution to the script until its interruption that match the search query.

It has to be noted that the search API is, as Twitter states, focused on relevance and not on completeness; some tweets and users may thus be missing and central users tend to be

overrepresented (González-Bailón *et al*., 2012).[7]

The previous example referred to a search of English tweets that use the hashtags #oxfam and #poverty. To illustrate further possibilities with the search API, other example specification for search queries are given in the following code snippet:

```
## find tweets using keyword
poverty <- search_tweets("poverty")

## Searching for tweets...

## Finished collecting tweets!

## find tweets using exact phrase
fightingpoverty <- search_tweets('"fight poverty"')

## Searching for tweets...
## Finished collecting tweets!

## find tweets using hashtag by verified users only
verifiedpoverty <- search_tweets("filter:verified #poverty")

## Searching for tweets...
## Finished collecting tweets!

## find tweets using keyword and containing a video
povertyvideo <- search_tweets("poverty filter:video")

## Searching for tweets...
## Finished collecting tweets!

## find popular tweets
povertypop <- search_tweets("poverty (min_faves:200 OR min_retweets:200)")

## Searching for tweets...
## Finished collecting tweets!
```

As the data collection streams real-time tweets, it is important to save the data once it has been collected as the same method call will always yield different results.

```
## save data frames
saveRDS(povertyhashtag,file=paste("povertyhashtag", Sys.Date(), ".rds",
sep=""))
```

`saveRDS()` saves single objects into a file in the current working directory. The name of the file is specified in the second argument (`file=`); as web data tends to change over time, the datasets should be marked with a timestamp. The code shown writes the current date into the file name by calling the function `Sys.Date()` when naming the files. These files saved can later be loaded using `readRDS()`.

User profile information

Twitter allows the collection of up to the last most recent 3,200 tweets from a user's individual timeline as long as the developer account accessing the API has access to that user (this means the user must either have a public profile or the developer follows that user; in both cases, it is necessary that the developer has not been blocked by the user).

In the following example, the collection of user data is shown, using the method `lookup_users()` which retrieves user-level information and the most recent `tweet`.`lookup_users()` takes user IDs or

screen names as an argument. The users are then looked up and data is stored in a new data frame. If the profile of only one or few users should be collected, the Twitter handle(s) of those users can simply be passed as an argument to `lookup_users()`. `c()` combines arguments into a vector or list. However, if the tweets of many users should be collected, it is more reasonable to work with a separate document. The document `listofusers.txt` contains a list of the Twitter user handles of four Oxfam accounts. This table is read into the data frame `listofusers`. The first few lines of the resulting data frame are shown by calling the method `head()`. The character @ is deleted from the usernames as lookup_users() expects user IDs or screen names without the @ as an argument. The users are then looked up and data is stored in a new data frame.

```
## get user data (few users)
oxfamuser <- lookup_users("oxfamgb")
oxfamusers <- lookup_users(c("oxfamgb", "oxfamgbpolicy"))

## get user data (many users)
listofusers <- read.table("listofusers.txt", header=TRUE, stringsAsFactors =
F, fileEncoding="UTF-8-BOM")
head(listofusers)

##       screen_name
## 1        @oxfamgb
## 2  @oxfamgbpolicy
## 3 @oxfamcampaigns
## 4   @oxfamgbpress

listofusers$screen_name <- gsub("@", "", listofusers$screen_name)
usr_df <- lookup_users(listofusers$screen_name)
```

To collect the tweets of users, the function `get_timelines()` can be called `get_timelines()` returns the last `n` (default 100, maximum 3200) tweets posted on the timeline of those users specified in the first parameter. Because `get_timelines()` does not provide the argument `retryonratelimit`, the error handling when reaching a rate limit has to be done manually. This is done by wrapping the argument in a `tryCatch()` block and specifying the system to wait for 15 minutes (900 seconds) when the function returns an error. This is not necessary when one is only collecting few users and/or the profile of users who do not have many tweets, but when working with many and/or active users, it is important to keep the restrictions employed by Twitter in mind.

```
## collect timeline of users
oxfamtml <- get_timelines("oxfamgb")
oxfamtmls <- get_timelines(c("oxfamgb", "oxfamgbpolicy"))

tmls <- tryCatch(get_timelines(usr_df$screen_name, n = 3200),
error=function(e) Sys.sleep(900))
```

The followees of a user can be obtained by employing the method `get_friends()`, while followers are collected with the method `get_followers()`. Friends thus refer to an account a given user follows while followers refer to accounts following a given user. `n` specifies the number of followers that should be returned; to collect the complete list of followers, it is set to the maximum number of followers any of the users in the list have. In the case presented, the code loops through all users and collects the complete list of all followers and writes them in a new data frame. `mutate(account = .x)` guarantees that the information about which user the followers are following is kept in the data frame.

Collecting all followers of a person can take a large amount of time with popular users. It should thus be reflected whether collecting all of the followers is necessary. If this is the case, it requires a stable internet connection and enough time. Since collecting all of the followers is not a default behaviour,

rtweet does not provide a function to retrieve all of them, but limits it to 5000 per API request. Up to 15 requests every 15 minutes are allowed, which means 75,000 is the maximum number of followers to be returned without waiting for the rate limit to reset.

The following code employs methods of the `purrr` and `dplyr` libraries. These need to be loaded in first (and installed if they have not been installed before. Again, call `install.packages()` to install any missing packages). purrr enhances R's functional programming toolkit. For example, the `map()` functions allow for efficient and easy iterations: They transform their input by applying a function to each element. `dplyr` provides additional tools for working with data frames; employing `mutate()` allows the addition of new variables and preserves existing ones.

```
library(purrr)

## Warning: package 'purrr' was built under R version 3.5.3

library(dplyr)

## Warning: package 'dplyr' was built under R version 3.5.3

## collect followees of users
followeesoxfam <- get_friends("oxfamgb", retryonratelimit = TRUE)
followees <- get_friends(usr_df$screen_name, retryonratelimit = TRUE)

## 1 friend networks collected!

## 2 friend networks collected!

## 3 friend networks collected!

## 4 friend networks collected!

## collect followers of users
followersoxfam <- map_df("oxfamgb", ~ get_followers(.x, n =
oxfamuser$followers_count, retryonratelimit = TRUE) %>% mutate(account =
.x))
followers <- map_df(usr_df$screen_name, ~ get_followers(.x, n =
max(usr_df$followers_count), retryonratelimit = TRUE) %>% mutate(account =
.x))
```

As the data collection takes time to execute and is restricted by rate limits, it is very sensible to save the data once it has been collected to not only guarantee reproducible analyses, but also to save on time. Again, `saveRDS()` is called to save single objects into files.

```
## save data frames
saveRDS(usr_df,file=paste("usr_df", Sys.Date(), ".rds", sep=""))
saveRDS(tmls,file=paste("tweets", Sys.Date(), ".rds", sep=""))
saveRDS(followees,file=paste("followees", Sys.Date(), ".rds", sep=""))
saveRDS(followers,file=paste("followers", Sys.Date(), ".rds", sep=""))
```

When a research project relies on the behaviour of specific user, the REST API offers many possibilities for data collection and analysis. Detailed data of users and their followers and friends can be collected as well as a large amount of their recent tweets, including the amount of engagement with them.

5 COLLECTED DATA

The following section gives a short overview to describe what data have been collected in the previously described process.

5.1 DATA FORMAT JSON

The Twitter APIs return tweets in JSON format. This is an open-standard file format that is based on key-value pairs, with named attributes and associated values. Both, tweets and users are served as JSON.

When accessed through a web browser in September 2019, the Twitter profile of Oxfam GB[8] looked like this:

Figure 1. Oxfam GB Twitter profile.

The data behind it in the JSON format looks like the following text:


```
{
    "contributors_enabled": false,
    "created_at": "Tue Aug 12 11:13:19 +0000 2008",
    "default_profile": false,
    "default_profile_image": false,
    "description": "Oxfam is a vibrant global movement of people who won't live
with the injustice of poverty.",
    "entities": {
        "description": {
            "urls": []
        },
        "url": {
```

```
            "urls": [
                {
                    "display_url": "oxfam.org.uk",
                    "expanded_url": "http://www.oxfam.org.uk",
                    "indices": [
                        0,
                        22
                    ],
                    "url": "http://t.co/qt7mRGgTtA"
                }
            ]
        }
    },
    "favourites_count": 4881,
    "follow_request_sent": false,
    "followers_count": 272745,
    "following": false,
    "friends_count": 8371,
    "geo_enabled": false,
    "has_extended_profile": true,
    "id": 15821039,
    "id_str": "15821039",
    "is_translation_enabled": false,
    "is_translator": false,
    "lang": null,
    "listed_count": 3476,
    "location": "",
    "name": "Oxfam",
    "notifications": false,
    "profile_background_color": "CFDDE6",
    "profile_background_image_url":
"http://abs.twimg.com/images/themes/theme1/bg.png",
    "profile_background_image_url_https":
"https://abs.twimg.com/images/themes/theme1/bg.png",
    "profile_background_tile": false,
    "profile_banner_url":
"https://pbs.twimg.com/profile_banners/15821039/1565364295",
    "profile_image_url":
"http://pbs.twimg.com/profile_images/1047149379910606849/PYdTCugb_normal.jpg",
    "profile_image_url_https":
"https://pbs.twimg.com/profile_images/1047149379910606849/PYdTCugb_normal.jpg",
    "profile_link_color": "60A333",
    "profile_location": null,
    "profile_sidebar_border_color": "FFFFFF",
    "profile_sidebar_fill_color": "7EC242",
    "profile_text_color": "000000",
    "profile_use_background_image": false,
    "protected": false,
    "screen_name": "oxfamgb",
    "status": { //key-value pairs of most recent status
    },
    "statuses_count": 86254,
    "time_zone": null,
    "translator_type": "none",
    "url": "http://t.co/qt7mRGgTtA",
    "utc_offset": null,
```

```
    "verified": true
}
```

An example tweet accessed through a web browser might look like this:

Figure 2. Oxfam GB example tweet.

In JSON, the following data is behind this tweet:

```
{
        "contributors": null,
        "coordinates": null,
        "created_at": "Fri Sep 20 15:51:12 +0000 2019",
        "entities": {
            "hashtags": [
                {
                    "indices": [
                        47,
                        54
                    ],
                    "text": "Malawi"
```

```
                },
                {
                    "indices": [
                        81,
                        95
                    ],
                    "text": "ClimateCrisis"
                }
            ],
            "symbols": [],
            "urls": [
                {
                    "display_url": "twitter.com/i/web/status/1\u2026",
                    "expanded_url":
"https://twitter.com/i/web/status/1175074940132974593",
                    "indices": [
                        117,
                        140
                    ],
                    "url": "https://t.co/zV9iFgBuOs"
                }
            ],
            "user_mentions": []
        },
        "favorite_count": 720,
        "favorited": false,
        "geo": null,
        "id": 1175074940132974593,
        "id_str": "1175074940132974593",
        "in_reply_to_screen_name": null,
        "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,
        "is_quote_status": false,
        "lang": "en",
        "place": null,
        "possibly_sensitive": false,
        "retweet_count": 318,
        "retweeted": false,
        "source": "<a href=\"https://www.spredfast.com\"
rel=\"nofollow\">Spredfast</a>",
        "text": "\"Let's take action now!!\" Jessy and Isaac from #Malawi explain
the impact of the #ClimateCrisis on their lives in f\u2026
https://t.co/zV9iFgBuOs",
        "truncated": true,
        "user": { //key-value pairs of user
        }
    }
```


JSON is a more organized and stable format to deliver data with better machine-readability than the visual representation through a web browser. Data is contained as key-value pairs, but allowing for lists and nesting. rtweet can convert the JSON data into parsed data frames and does this by default so the disentangling of the Twitter API return objects does not need to be done manually. Nested values are disentangled and keys generally become the row names of the new data frame.

Twitter makes a range of data available. For each tweet, it includes the standard information, such as the text of tweet, the user ID of the poster, a unique ID, the time it has been sent and geo-data, if available. Each tweet also contains entity objects, which are arrays of further tweet contents, such as hashtags, media or mentions. Media content can have further entities (for example, a picture has a URL and a description). In addition to the text content of a tweet, it can thus have a large number of further attributes. A thorough documentation of the keys and values included can be found here for Twitter users[9] and here for tweets.[10]

5.2 OVERVIEW OF THE DATA

As aforementioned, rtweet stores the collected data as data frames. To gain an impression of how the JSON has been transformed into a data frame and what has been collected through the sent request, `colnames()` can be called, which prints out the names of all columns of a data frame. While most columns describe the general tweet and user collected, some are specific to the owner of the access token, i.e. the user sending the request: the columns favourited and retweeted refer to whether the owner of the access token has favourited or retweeted this specific tweet. The columns `is_retweet` identifies which tweets are retweets from other accounts. Depending on whether the specific research projects want to only consider original tweets, this information can be used to filter out retweets. `dim()` further gives the number of dimensions, i.e. the number of rows and columns. More insight into the data is gained by using `summary(),` which reports mean and quantiles of variables; in the following example, the summary statistics for the date of creation of a tweet and for the number of retweets are called.

```
## describing datasets
colnames(usr_df)
```

```
##  [1] "user_id"                "status_id"
##  [3] "created_at"             "screen_name"
##  [5] "text"                   "source"
##  [7] "display_text_width"     "reply_to_status_id"
##  [9] "reply_to_user_id"       "reply_to_screen_name"
## [11] "is_quote"               "is_retweet"
## [13] "favorite_count"         "retweet_count"
## [15] "hashtags"               "symbols"
## [17] "urls_url"               "urls_t.co"
## [19] "urls_expanded_url"      "media_url"
## [21] "media_t.co"             "media_expanded_url"
## [23] "media_type"             "ext_media_url"
## [25] "ext_media_t.co"         "ext_media_expanded_url"
## [27] "ext_media_type"         "mentions_user_id"
## [29] "mentions_screen_name"   "lang"
## [31] "quoted_status_id"       "quoted_text"
## [33] "quoted_created_at"      "quoted_source"
## [35] "quoted_favorite_count"  "quoted_retweet_count"
## [37] "quoted_user_id"         "quoted_screen_name"
## [39] "quoted_name"            "quoted_followers_count"
## [41] "quoted_friends_count"   "quoted_statuses_count"
## [43] "quoted_location"        "quoted_description"
## [45] "quoted_verified"        "retweet_status_id"
## [47] "retweet_text"           "retweet_created_at"
## [49] "retweet_source"         "retweet_favorite_count"
## [51] "retweet_retweet_count"  "retweet_user_id"
## [53] "retweet_screen_name"    "retweet_name"
## [55] "retweet_followers_count" "retweet_friends_count"
## [57] "retweet_statuses_count" "retweet_location"
## [59] "retweet_description"    "retweet_verified"
## [61] "place_url"              "place_name"
```

```
## [63] "place_full_name"         "place_type"
## [65] "country"                 "country_code"
## [67] "geo_coords"              "coords_coords"
## [69] "bbox_coords"             "status_url"
## [71] "name"                    "location"
## [73] "description"             "url"
## [75] "protected"               "followers_count"
## [77] "friends_count"           "listed_count"
## [79] "statuses_count"          "favourites_count"
## [81] "account_created_at"      "verified"
## [83] "profile_url"             "profile_expanded_url"
## [85] "account_lang"            "profile_banner_url"
## [87] "profile_background_url"  "profile_image_url"

dim(usr_df)

## [1]  4 88

colnames(tmls)

##  [1] "user_id"                "status_id"
##  [3] "created_at"             "screen_name"
##  [5] "text"                   "source"
##  [7] "display_text_width"     "reply_to_status_id"
##  [9] "reply_to_user_id"       "reply_to_screen_name"
## [11] "is_quote"               "is_retweet"
## [13] "favorite_count"         "retweet_count"
## [15] "hashtags"               "symbols"
## [17] "urls_url"               "urls_t.co"
## [19] "urls_expanded_url"      "media_url"
## [21] "media_t.co"             "media_expanded_url"
## [23] "media_type"             "ext_media_url"
## [25] "ext_media_t.co"         "ext_media_expanded_url"
## [27] "ext_media_type"         "mentions_user_id"
## [29] "mentions_screen_name"   "lang"
## [31] "quoted_status_id"       "quoted_text"
## [33] "quoted_created_at"      "quoted_source"
## [35] "quoted_favorite_count"  "quoted_retweet_count"
## [37] "quoted_user_id"         "quoted_screen_name"
## [39] "quoted_name"            "quoted_followers_count"
## [41] "quoted_friends_count"   "quoted_statuses_count"
## [43] "quoted_location"        "quoted_description"
## [45] "quoted_verified"        "retweet_status_id"
## [47] "retweet_text"           "retweet_created_at"
## [49] "retweet_source"         "retweet_favorite_count"
## [51] "retweet_retweet_count"  "retweet_user_id"
## [53] "retweet_screen_name"    "retweet_name"
## [55] "retweet_followers_count" "retweet_friends_count"
## [57] "retweet_statuses_count" "retweet_location"
## [59] "retweet_description"    "retweet_verified"
## [61] "place_url"              "place_name"
## [63] "place_full_name"        "place_type"
## [65] "country"                "country_code"
## [67] "geo_coords"             "coords_coords"
## [69] "bbox_coords"            "status_url"
## [71] "name"                   "location"
## [73] "description"            "url"
## [75] "protected"              "followers_count"
```

```
## [77] "friends_count"           "listed_count"
## [79] "statuses_count"          "favourites_count"
## [81] "account_created_at"      "verified"
## [83] "profile_url"             "profile_expanded_url"
## [85] "account_lang"            "profile_banner_url"
## [87] "profile_background_url"  "profile_image_url"

dim(tmls)

## [1] 12792    88

## summary statistics
summary(usr_df$created_at)

##                   Min.                1st Qu.                 Median
## "2019-09-23 14:30:14" "2019-09-23 15:27:29" "2019-09-23 16:19:14"
##                   Mean                3rd Qu.                   Max.
## "2019-09-23 16:18:55" "2019-09-23 17:10:40" "2019-09-23 18:07:01"

summary(tmls$retweet_count)

##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##     0.00     0.00     1.00    27.72     6.00 52670.00
```

The basic components and available metadata of tweets provide a set of descriptive characteristics that can and should be reported about any collection of tweets and users. It is important to gain a first, general understanding about the collected data and its scope to refine decisions informing the analysis that follows. In any case, it is important to pre-process the data and identify outliers. Online data is particularly prone to outliers and large variations (for example, tweets that have gone viral and have received a lot of retweets and likes). How to deal with these outliers is a research question: they might need to be excluded or they might form the core of the study. Generally, it is also important to check whether outliers are real users or bots. Bots are software applications that run automated scripts. In the case of Twitter, autonomous computer programs can control a Twitter account via the API and autonomously tweet, retweet, follow users and perform other actions. This can skew statistical analysis, especially when Twitter accounts are used for spamming. The R package *tweetbotornot* can be used to identify bots. It is based on a machine learning algorithm, taking into account a number of factors, such as a user's number of followers and followed account, the bio, the use of hashtags and mentions and the user's capitalization in their tweets. The algorithm can correctly classify Twitter accounts as being bots or not bots 93.8% of the time. A manual check of suspicious, i.e. highly active, Twitter profiles can also bring certainty as to whether these are simply people enjoying Twitter or spam accounts. In any case, it is important to grasp the influence of outliers and be aware of it.

6 OUTLOOK ON DATA ANALYSIS

Once the data has been collected, it can be analysed in whatever way is suitable for the specific research study. Possible applications are broad. The following chapter aims to illustrate possible approaches to analysis and uses examples from the aforementioned impact evaluation (see Schwitter and Liebe, 2019), but it will not go into details on how to reproduce such analysis.

6.1 TWEETING BEHAVIOUR OVER TIME

For every tweet collected, the exact date and time it was posted is recorded. This allows investigation into how tweeting behaviour has changed over time and also during what day of the week or time of the day users are

particularly active. The following figure shows the relative frequency of tweets of Twitter users involved in the Oxfam project in four different regions in Tanzania (number of tweets per day divided by the number of active Twitter users), highlighting days of trainings on smartphone and social media use that were organized in the setting up of the project. The analysis shows how tweeting behaviour is spurred by these trainings and also highlights variations over time and between regions (for more details see Section 4.3.2 in Schwitter and Liebe, 2019).

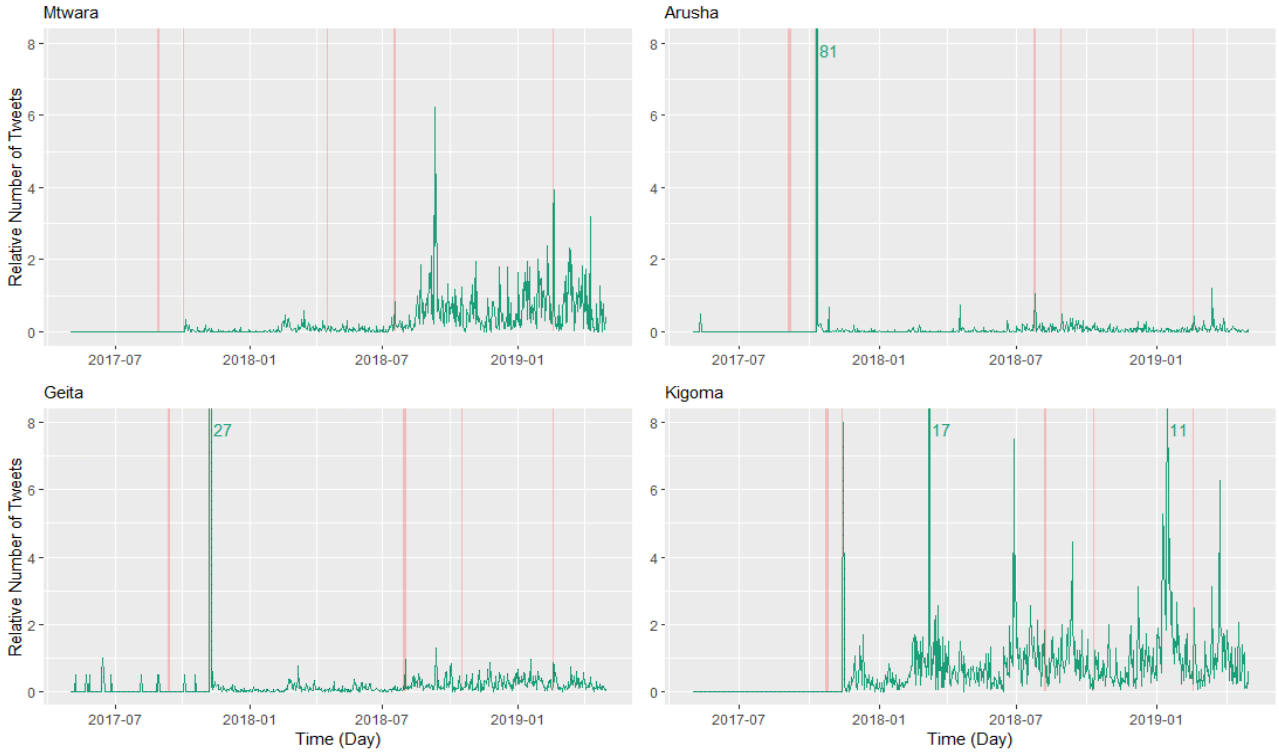

Figure 3. Tweeting behaviour over time.

6.2 WORDCLOUDS

Besides measuring the tweeting volume, it can also be of interest to analyse the actual content of tweets. A simple but helpful illustration are so-called wordclouds: visual representations of text data. The importance of words (in this case, defined as frequency of occurrence, excluding *stop words*, the most common words of a language) can be shown through font sizes and font colour. It is a useful format to quickly perceive the most prominent words in a bulk of text and thus identify the topics that seem to be of greatest relevance. However, wordclouds should be interpreted in conjunction with the raw data as they only display single words and do not give any context at all; more advanced semantic tools and algorithms that look for links or patterns in words and their connections are, for example, latent semantic analysis (see the package *quanteda* for advanced quantitative text analysis in R).

Word clouds for the four regions were generated to help understand which issues were raised by the Twitter users involved in the Oxfam project in Tanzania. One of these is presented in the following figure. We observe that Twitter posts are mostly written in Swahili, that behaviour is highly linked to the project and that different societal issues are being raised (for more details see Section 4.3.2 in Schwitter and Liebe, 2019).

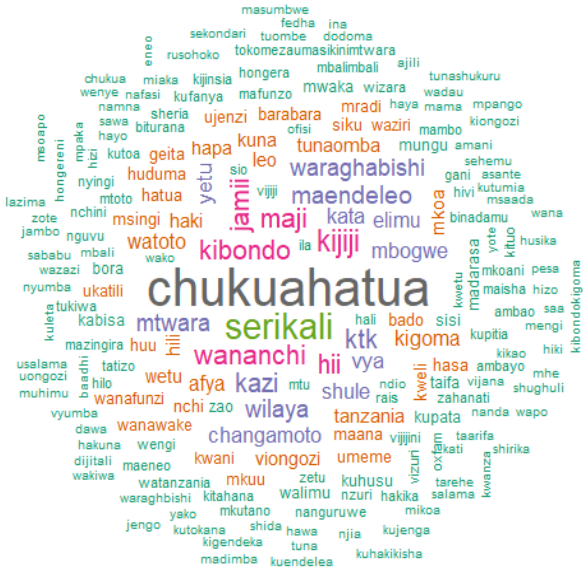

Figure 4. Wordcloud.

6.3 SOCIAL NETWORK ANALYSIS

Since Twitter allows and encourages the establishment of relationships and interaction with other users, adopting a network perspective can further inform any analysis of Twitter users.

With regard to the described project, it was of interest to analyse how the community animators – community organizers who mobilize citizens around a common advocacy agenda – and selected influencers – influential bloggers and social media users involved in the project – were engaging with key stakeholders, a number of specified other accounts that belong to government officials, service providers, NGOs and CSOs. To find out how these Twitter accounts are connected with the animators and influencers in the study, it was of interest to explore which – if any – key stakeholders followed animators and influencers. The following figure shows that the majority of animators are not followed by any official account. The plot helps to visualize which (subgroups of) officials are reached through the animators and influencers, and the general connectedness of the Twitter users (for more details see Section 4.3.4 in Schwitter and Liebe, 2019).

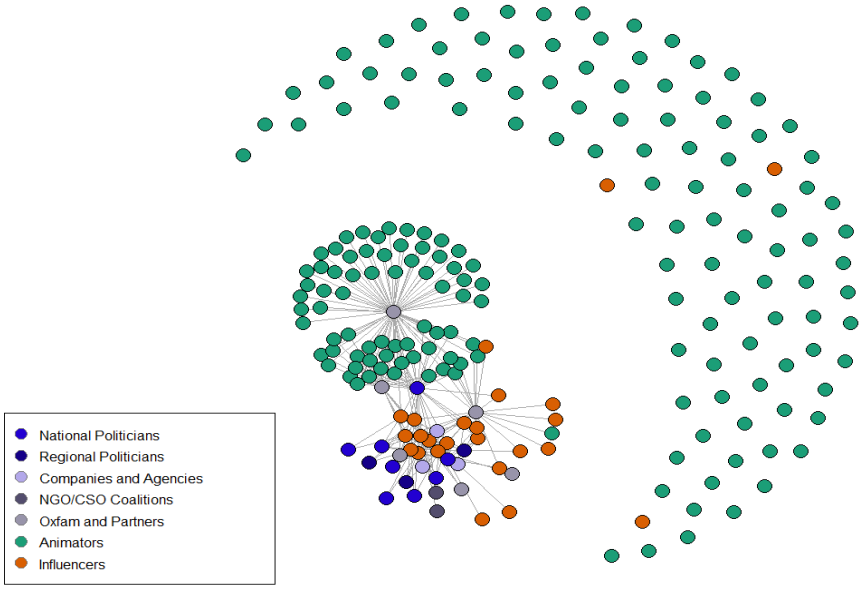

Figure 5. Bipartite social network between officials and animators/influencers.

7 CONCLUDING REMARKS

Social media has grown, and is still growing, to become an important space of social interaction in today's life in the Global North and South. As people sign up to these platforms, create content and link to other users, they generate a valuable new source of data. Collecting and analysing this data can prove to be a useful complement to traditional data sources, which might even allow the answering of new and innovative research questions. In contrast to more traditional surveys, social media data is generally produced as a by-product of someone's everyday life and not through the explicit influence of research teams, and its collection can be comparatively cost- and time-efficient. On the other hand, social media data collection and analysis also include some challenges, which need to be considered depending on the research purpose. These challenges include a population bias, transparency and replicability of social media data analysis (Ruths and Pfeffer, 2014; Lazer *et al*., 2014).

Population bias refers to the fact that users of social media typically differ in their characteristics from the general population. Next to the digital divides between countries, e.g. higher access rates to the internet and smartphone use in wealthier countries compared with poorer countries, there are also digital divides within countries, both wealthier and poorer ones (Pew Research Center, 2018). For example, Twitter and Facebook users are often younger and better educated compared to the general population, and users and non-users of social media might also differ in other characteristics, such as political interest and attitudes (Mellon and Prosser, 2017). Yet more important than the digital divide per se, i.e. the differences between users and non-users or 'haves' and 'have-nots', is the digital inequality, e.g. differences between individuals or groups in society regarding their digital skills, social support from more experienced users, and purpose of social media use (DiMaggio, Hargittai *et al*., 2001). Mere access to social media does not necessarily decrease existing social inequalities and could even strengthen them as power dynamics are also at play in online spaces and citizens may lack the agency to engage with social media (see Roberts and Hernandez, 2019).

With regard to social media data, there is also the possibility of a sample bias. For example, Twitter data used by researchers might not be representative for the platform's overall data (Ruths and Pfeffer, 2014); the publicly available data can be biased towards specific topics/content or user networks (Morstatter, Pfeffer and Liu, 2014). Results of corresponding data analyses would then not be generalizable to user behaviour of the social media platform.

As with any other type of data analysis, transparency and replicability of social media data analysis can be challenging (Lazer *et al*., 2014). This is apparent if the data cannot be shared and hence it might be impossible to re-collect the data. But even if data are shared it is important to document all steps in data preparation, such as the definition of outliers or the reporting of stop words in text analysis, in order to replicate the analysis.

These guidelines have focused on Twitter data. Collecting data from other social media sites generally follows the same logic. In a first step, access to the API has to be established by requesting a developer account and creating an application, the same way as described in these guidelines with Twitter. The API can then be used to access the data. Which data are accessible through the public APIs depend on the particular social media site. For example, the popular social media channels Facebook and Instagram provide APIs, but their access towards user data have been heavily limited in the past years. Default permissions only include name and profile picture for public profiles; all other data need specific permissions and also often require going through a review process of the company (see also this Permission Reference[11]). In contrast to this, the streaming API and the REST APIs provided by Twitter allow the collection and analysis of an only somewhat restricted amount of data for free to the public. If a less restricted amount of data is necessary for the research goal, further fee-based subscriptions exist.

Exemptions to this process are messenger services like WhatsApp or Telegram. These services are used to communicate with a restricted group of specific others. This can range from chatting to one other person to group conversation with over 200 members. While the ethical considerations are more complex in these contexts

– obtaining informed consent is absolutely vital when researching conversations in a seemingly private environment such as a group chat – the technical process of obtaining the data is easier as these services generally offer downloading of chat histories of the chats one is a member of.

The free programming language R has been used in these guidelines to collect Twitter data and to subsequently analyse it. While some technical know-how is necessary to perform these tasks, the procedure is in general straightforward and similar for many use cases so that only small adaptations are necessary once the foundation has been understood. Furthermore, the online community and documentation around R are extensive, detailed and helpful. While rtweet supports data collection on Twitter, there is often no well-maintained R package available that specifically supports data collection on specific social media sites. In these cases, other packages can be used: For example, *httr* provides general tools for working with URLs and HTTP and *jsonlite* can convert the collected JSON data to manageable data frames.

Besides the manual data collection that has been described in this guide, some free and fee-based third party service providers also exist, such as Twitonomy[12] or TAGS[13] for Twitter. While relying on these services reduces the amount of technical know-how that is necessary to perform web data collection, their functionality is more restricted.

When planning the collection of social media data, it is important to keep the restrictions of the social websites in mind as data is generally only available to a limited extent. In the case of Twitter, only the most recent 3,200 tweets of a user can be retrieved through the REST API and the free streaming API can only retrieve tweets that have been posted in the last week. The possibilities to ‘go back in time’ are thus severely limited. If the planned research concerns highly active users or aims to track changes over time, it is necessary to start early with the data collection and subsequently repeat it at regular intervals.

Social media data has a lot of potential for research and evaluation as it can provide an innovative source of (big) data, which can be collected with comparatively little effort and in a short amount of time. As simple as the data collection might seem from a technical point of view, it is important to plan carefully considering ethical concerns, be aware of restrictions, and to document and reflect on the process to handle the challenges it comes with.

APPENDIX: COMPLETE CODE

In the following section, the code for these guidelines is shown so it can easily be copied. With the exception of the access tokens, the code is complete and should run without any errors. As mentioned in the guidelines, please be aware that in particular the collection of all followers of a user can take a large amount of time; therefore, be cautious when running this code.

```
## load packages
library(rtweet)
library(purrr)
library(dplyr)

## create authentication token
create_token(
  app = "<YOUR APP NAME>",
  consumer_key = "<YOUR CONSUMER API KEY>",
  consumer_secret = "<YOUR CONSUMER API SECRET KEY>",
  access_token = "<YOUR ACCESS TOKEN>",
  access_secret = "<YOUR ACCESS TOKEN SECRET>")

## stream tweets using the hashtags
tweetshashtags <- stream_tweets("#oxfam,#poverty", timeout = 5)

## stream tweets using user ID
tweetsoxfam <- stream_tweets(15821039, timeout = 5)

## find English tweets using the hashtags
povertyhashtag <- search_tweets("lang:en #oxfam #poverty", n = 200, include_rts =
FALSE)
povertyhashtagRTS <- search_tweets("lang:en #oxfam #poverty", n = 200,
include_rts = TRUE)

## find tweets using keyword
poverty <- search_tweets("poverty")
## find tweets using exact phrase
fightingpoverty <- search_tweets('"fight poverty"')
## find tweets using hashtag by verified users only
verifiedpoverty <- search_tweets("filter:verified #poverty")
## find tweets using keyword and containing a video
povertyvideo <- search_tweets("poverty filter:video")
## find popular tweets
povertypop <- search_tweets("poverty (min_faves:200 OR min_retweets:200)")

## save data frames
saveRDS(povertyhashtag,file=paste("povertyhashtag", Sys.Date(), ".rds", sep=""))

## get user data (few users)
oxfamuser <- lookup_users("oxfamgb")
oxfamusers <- lookup_users(c("oxfamgb", "oxfamgbpolicy"))

## get user data (many users)
listofusers <- read.table("listofusers.txt", header=TRUE, stringsAsFactors = F,
fileEncoding="UTF-8-BOM")
head(listofusers)
```

```
listofusers$screen_name <- gsub("@", "", listofusers$screen_name)
usr_df <- lookup_users(listofusers$screen_name)

## collect timeline of users
oxfamtml <- get_timelines("oxfamgb")
oxfamtmls <- get_timelines(c("oxfamgb", "oxfamgbpolicy"))

tmls <- tryCatch(get_timelines(usr_df$screen_name, n = 3200), error=function(e)
Sys.sleep(900))

## collect followees of users
followeesoxfam <- get_friends("oxfamgb", retryonratelimit = TRUE)
followees <- get_friends(usr_df$screen_name, retryonratelimit = TRUE)

## collect followers of users
followersoxfam <- map_df("oxfamgb", ~ get_followers(.x, n =
oxfamuser$followers_count, retryonratelimit = TRUE) %>% mutate(account = .x))
followers <- map_df(usr_df$screen_name, ~ get_followers(.x, n =
max(usr_df$followers_count), retryonratelimit = TRUE) %>% mutate(account = .x))

## save data frames
saveRDS(usr_df,file=paste("usr_df", Sys.Date(), ".rds", sep=""))
saveRDS(tmls,file=paste("tweets", Sys.Date(), ".rds", sep=""))
saveRDS(followees,file=paste("followees", Sys.Date(), ".rds", sep=""))
saveRDS(followers,file=paste("followers", Sys.Date(), ".rds", sep=""))


## describing datasets
colnames(usr_df)
dim(usr_df)

colnames(tmls)
dim(tmls)

## summary statistics
summary(usr_df$created_at)
summary(tmls$retweet_count)
```

REFERENCES

Abreu Lopes, Claudia, Savita Bailur, and Giles Barton-Owen. (2018). Can Big Data Be Used for Evaluation? A UN Women Feasibility Study. New York: UN Women. https://www.unwomen.org/en/digital-library/publications/2018/4/can-big-data-be-used-for-evaluation.

Al Baghal, Tarek, Luke Sloan, Curtis Jessop, Matthew L Williams, and Pete Burnap. (2019). Linking Twitter and Survey Data: The Impact of Survey Mode and Demographics on Consent Rates Across Three Uk Studies. Social Science Computer Review. SAGE Publications Sage CA: Los Angeles, CA, 0894439319828011. https://journals.sagepub.com/doi/pdf/10.1177/0894439319828011.

Boyd, Danah, Scott Golder, and Gilad Lotan. (2010). Tweet, Tweet, Retweet: Conversational Aspects of Retweeting on Twitter. In System Sciences (Hicss), 2010 43rd Hawaii International Conference on, 1–10. IEEE. https://www.danah.org/papers/TweetTweetRetweet.pdf.

DiMaggio, Paul, Eszter Hargittai, et al. (2001). From the 'Digital Divide' to 'Digital Inequality': Studying Internet Use as Penetration Increases. Working Paper, Princeton: Center for Arts and Cultural Policy Studies 15(1). https://pdfs.semanticscholar.org/4843/610b79d670136e3cdd12311f91f5cc98d2ee.pdf.

Driscoll, Kevin, and Shawn Walker. (2014). Big Data, Big Questions. Working Within a Black Box: Transparency in the Collection and Production of Big Twitter Data. International Journal of Communication 8(20). https://ijoc.org/index.php/ijoc/article/view/2171.

González-Bailón, Sandra, Ning Wang, Alejandro Rivero, Javier Borge-Holthoefer, and Yamir Moreno. (2012). Assessing the Bias in Communication Networks Sampled from Twitter. arXiv Preprint arXiv:1212.1684. https://arxiv.org/abs/1212.1684.

Grant, Will J, Brenda Moon, and Janie Busby Grant. (2010). Digital Dialogue? Australian Politicians' Use of the Social Network Tool Twitter. Australian Journal of Political Science 45(4). Taylor & Francis: 579–604. https://openresearch-repository.anu.edu.au/handle/10440/1264.

Lazer, David, Ryan Kennedy, Gary King, and Alessandro Vespignani. (2014). The Parable of Google Flu: Traps in Big Data Analysis. Science 343(6176). American Association for the Advancement of Science: 1203–5. https://www.dhi.ac.uk/san/waysofbeing/data/data-crone-lazer-2014.pdf.

Lombardini, Simone, and Emily Tomkys Valteri. (2017). Going Digital: Using and Sharing Real-Time Data During Fieldwork. Oxford: Oxfam GB. https://policy-practice.oxfam.org.uk/publications/going-digital-using-and-sharing-real-time-data-during-fieldwork-620432.

Lombardini, Simone, Alexia Pretari, and Emily Tomkys Valteri. (2018). Going Digital: Improving Data Quality with Digital Data Collection. Oxford: Oxfam GB. https://policy-practice.oxfam.org.uk/publications/going-digital-improving-data-quality-with-digital-data-collection-620522.

Lotan, Gilad, Erhardt Graeff, Mike Ananny, Devin Gaffney, Ian Pearce, and Danah Boyd. (2011). The Revolutions Were Tweeted: Information Flows During the 2011 Tunisian and Egyptian Revolutions. International Journal of Communication 5: 1375–1405. https://www.researchgate.net/publication/235354320_The_Revolutions_Were_Tweeted_Information_Flows_During_the_2011_Tunisian_and_Egyptian_Revolutions.

Mamic, Lilia Ivana, and Isidoro Arroyo Almaraz. (2013). How the Larger Corporations Engage with Stakeholders Through Twitter. International Journal of Market Research 55(6). SAGE Publications Sage UK: London, England: 851–72. https://www.researchgate.net/publication/270184875_How_the_larger_corporations_engage_with_stakeholders_through_Twitter.

Mayer-Schönberger, Viktor, and Kenneth Cukier. (2013). Big Data: A Revolution That Will Transform How We Live, Work, and Think. Boston, MA: Houghton Mifflin Harcourt.

Mellon, Jonathan, and Christopher Prosser. (2017). Twitter and Facebook Are Not Representative of the General Population: Political Attitudes and Demographics of British Social Media Users. Research & Politics 4 (3). SAGE Publications Sage UK: London, England: 1–9. https://journals.sagepub.com/doi/pdf/10.1177/2053168017720008.

Morstatter, Fred, Jürgen Pfeffer, and Huan Liu. (2014). "When is it Biased? Assessing the Representativeness of Twitter's Streaming API. In Proceedings of the 23rd International Conference on World Wide Web. ACM. https://arxiv.org/pdf/1401.7909.pdf.

Parmelee, John H, and Shannon L Bichard. (2011). Politics and the Twitter Revolution: How Tweets Influence the Relationship Between Political Leaders and the Public. Lexington Books.

Pew Research Center. (2018). Social Media Use Continues to Rise in Developing Countries but Plateaus Across Developed Ones. https://www.pewresearch.org/global/2018/06/19/social-media-use-continues-to-rise-in-developing-countries-but-plateaus-across-developed-ones/.

Pretari, Alexia. (2019). Active Citizenship in Tanzania: Impact Evaluation of the 'Governance and Accountability Through Digitalization' Project. Oxford: Oxfam GB. https://policy-practice.oxfam.org.uk/publications/going-digital-using-digital-technology-to-conduct-oxfams-effectiveness-reviews-578816.

Roberts, Tony, and Kevin Hernandez. (2019). Digital Access Is Not Binary: The 5 'A's of Technology Access in the Philippines. The Electronic Journal of Information Systems in Developing Countries. Wiley Online Library, e12084. https://onlinelibrary.wiley.com/doi/full/10.1002/isd2.12084.

Ruths, Derek, and Jürgen Pfeffer. (2014). Social Media for Large Studies of Behavior. Science 346(6213). American Association for the Advancement of Science: 1063–4. https://people.cs.umass.edu/~brenocon/smacss2015/papers/Science-2014-Ruths-1063-4.pdf.

Salganik, Matthew. (2017). Bit by Bit: Social Research in the Digital Age. Princeton: Princeton University Press. https://www.bitbybitbook.com/.

Schwitter, Nicole, and Ulf Liebe. (2019). Twitter Data Analysis. In Active Citizenship in Tanzania: Impact Evaluation of the 'Governance and Accountability Through Digitalization' Project, edited by Alexia Pretari, Chapters 3.4 and 4.3. Oxford: Oxfam GB. https://policy-practice.oxfam.org.uk/publications/going-digital-using-digital-technology-to-conduct-oxfams-effectiveness-reviews-578816.

Shao, Chengcheng, Giovanni Luca Ciampaglia, Onur Varol, Kai-Cheng Yang, Alessandro Flammini, and Filippo Menczer. (2018). The Spread of Low-Credibility Content by Social Bots. Nature Communications 9(1). Nature Publishing Group: 4787. https://www.nature.com/articles/s41467-018-06930-7/.

Tomkys, Emily, and Simone Lombardini. (2015). Going Digital: Using Digital Technology to Conduct Oxfam's Effectiveness Reviews. Oxford: Oxfam GB. https://policy-practice.oxfam.org.uk/publications/going-digital-using-digital-technology-to-conduct-oxfams-effectiveness-reviews-578816.

Townsend, Leanne, and Claire Wallace. (2016). Social Media Research: A Guide to Ethics. The University of Aberdeen. https://www.gla.ac.uk/media/Media_487729_smxx.pdf.

Vonk, Jaynie. (2019). Going Digital: Privacy and Data Security Under GDPR for Quantitative Impact Evaluation. Oxford: Oxfam GB. https://policy-practice.oxfam.org.uk/publications/going-digital-privacy-and-data-security-under-gdpr-for-quantitative-impact-eval-620884.

Zimmer, Michael. (2010). ‘But the Data Is Already Public’: On the Ethics of Research in Facebook. Ethics and Information Technology 12(4). Springer: 313–25. http://www.sfu.ca/~palys/Zimmer-2010-EthicsOfResearchFromFacebook.pdf.

NOTES

[1] https://policy-practice.oxfam.org.uk/our-approach/monitoring-evaluation/effectiveness-reviews

[2] https://rstudio.com/resources/training/

[3] https://developer.twitter.com/en/apply/user

[4] https://apps.twitter.com/

[5] Consult the documentation under https://cran.r-project.org/web/packages/rtweet/rtweet.pdf for further details on any functions. When in R, use the command? to open the help window.

[6] https://twitter.com/oxfamgb

[7] See https://developer.twitter.com/en/docs/tweets/search/overview/standard.html

[8] https://twitter.com/oxfamgb

[9] https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/user-object

[10] https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/tweet-object

[11] https://developers.facebook.com/docs/facebook-login/permissions

[12] https://www.twitonomy.com/

[13] https://tags.hawksey.info/

This publication was written by Nicole Schwitter and Ulf Liebe and commissioned by Simone Lombardini and Alexia Pretari from Oxfam GB. It is part of a series of papers and reports written to inform public debate on development and humanitarian policy issues.

For further information on the issues raised in this publication please email slombardini1@oxfam.org.uk.

Published by Oxfam GB for Oxfam International under
ISBN 978-1-78748-568-6 in February 2020. DOI: 10.21201/2020.5686
Oxfam GB, Oxfam House, John Smith Drive, Cowley, Oxford, OX4 2JY, UK.

OXFAM

Oxfam is an international confederation of 20 organizations networked together in more than 90 countries, as part of a global movement for change, to build a future free from the injustice of poverty. Please write to any of the agencies for further information, or visit www.oxfam.org.

Oxfam America (www.oxfamamerica.org)

Oxfam Australia (www.oxfam.org.au)

Oxfam-in-Belgium (www.oxfamsol.be)

Oxfam Brasil (www.oxfam.org.br)

Oxfam Canada (www.oxfam.ca)

Oxfam France (www.oxfamfrance.org)

Oxfam Germany (www.oxfam.de)

Oxfam GB (www.oxfam.org.uk)

Oxfam Hong Kong (www.oxfam.org.hk)

Oxfam IBIS (Denmark) (http://oxfamibis.dk/)

Oxfam India (www.oxfamindia.org)

Oxfam Intermón (Spain) (www.oxfamintermon.org)

Oxfam Ireland (www.oxfamireland.org)

Oxfam Italy (www.oxfamitalia.org)

Oxfam Mexico (www.oxfammexico.org)

Oxfam New Zealand (www.oxfam.org.nz)

Oxfam Novib (Netherlands) (www.oxfamnovib.nl)

Oxfam Québec (www.oxfam.qc.ca)

Oxfam South Africa (http://www.oxfam.org.za/)

KEDV (Turkey) (https://www.kedv.org.tr/)

OXFAM