Harvesting of Social Media Collections in the BESOCIAL project - User documentation

Editors	Sven Lieber
Responsible partners	IDLab
Version	1.0 - Added Social Feed Manager documentation

1 Introduction

This user documentation explains how social media collections can be created, harvested and exported via **Social Feed Manager** for Twitter and via **Instaloader** for Instagram based on provided seeds. A seed is something to be harvested such as an account identified by an account name or ID, or a search term for example based on hashtags or locations.

2 Twitter - Social Feed Manager

The tool Social Feed Manager (SFM) provides a web interface which can be used to harvest social media content from different social media providers based on provided seed lists and API credentials. We explain how the tool can be configured with API credentials and how different types of Twitter harvests can be performed in a repeating schedule.

2.1 Provide API credentials

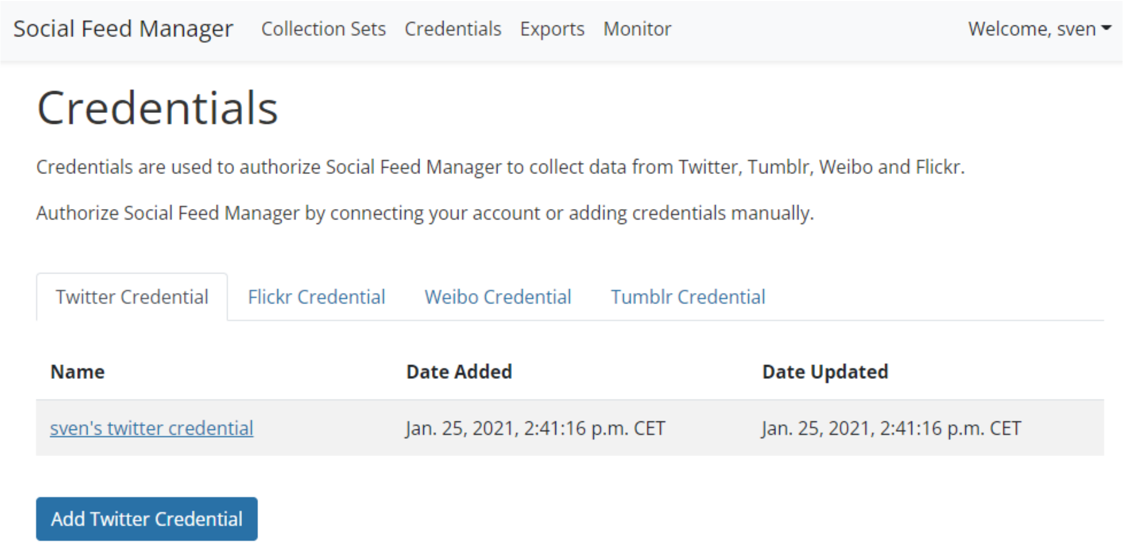
SFM is able to manage several API credentials (if you have more than one), they can be entered via a user interface and when creating a collection one of the credentials can be selected.

For Twitter API credentials can be requested online[1]. These credentials allow you to harvest a certain number of tweets per month. Recently an academic version of the API was introduced, however, the application process consisting of several questions regarding your plans with the data will be reviewed by a person and thus this process may take longer.

Within SFM you can click on “Credentials” in the top navigation bar (figure 1) to add new credentials. Since SFM is a framework which is able to harvest social media from different social media providers, you have to further select for which provider you would like to add credentials. For Twitter you can click “Add Twitter Credential” and provide the requested values such as Consumer key and Consumer secret. The process to request API keys from Twitter is also explained in the SFM form to add new credentials. It is possible to add more than one credential, you can specify a name for a credential. In case of many collections harvested around the same time, this allows you to use different credentials for different collections and helps not to overuse one API key[2].

[1] https://developer.twitter.com/en/products/twitter-api

[2] There is a rate limitation about how many tweets can be requested per month or in a specific time frame with one API key.



Credentials

Credentials are used to authorize Social Feed Manager to collect data from Twitter, Tumblr, Weibo and Flickr.

Authorize Social Feed Manager by connecting your account or adding credentials manually.

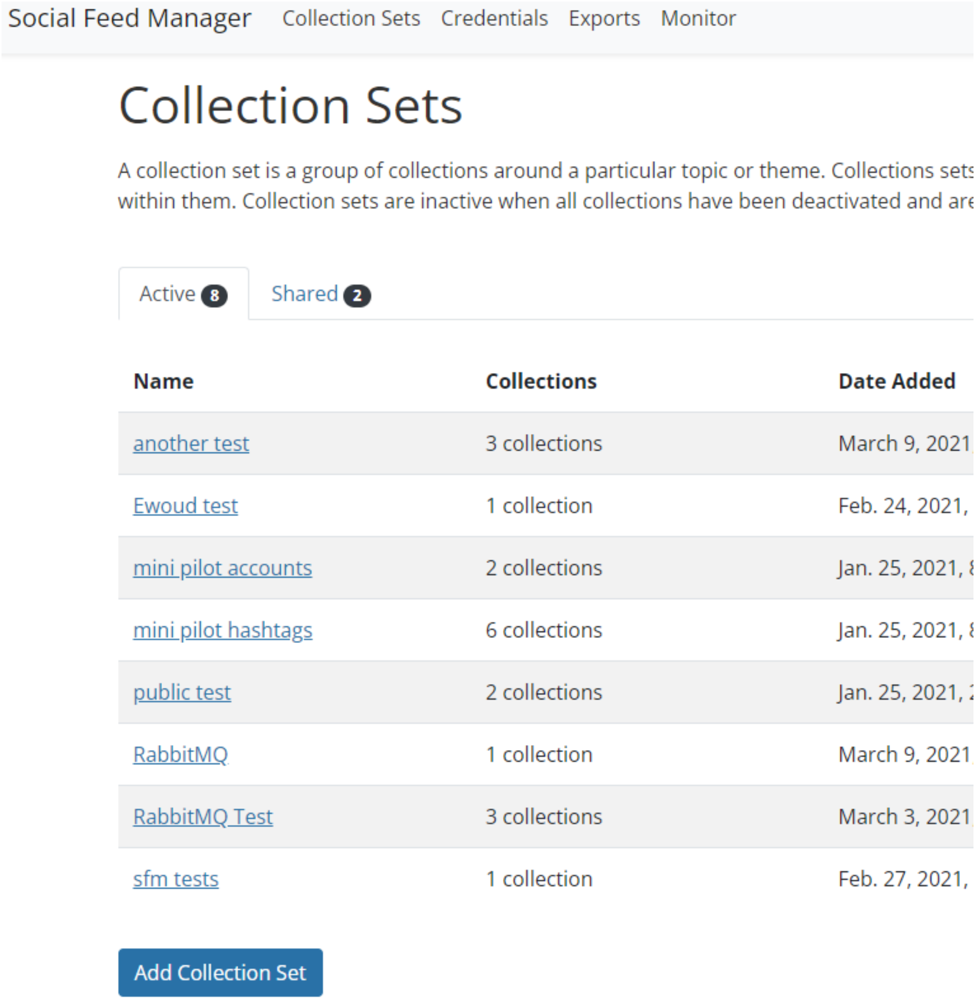
Name	Date Added	Date Updated
sven's twitter credential	Jan. 25, 2021, 2:41:16 p.m. CET	Jan. 25, 2021, 2:41:16 p.m. CET

Figure 1: Social media API credentials can be added to SFM via the “Credentials” button in the top navigation bar.

2.2 Create a collection set

The starting point to create harvests is to first create a collection set, it serves as a container for different thematically related collections.

If you click on “Collection sets” in the top navigation bar, you see a list of already created collection sets (figure 2). The blue “Add collection set” button on the bottom of the page opens a form in which you can enter information about the new collection set.



Collection Sets

A collection set is a group of collections around a particular topic or theme. Collections sets
within them. Collection sets are inactive when all collections have been deactivated and are

Name	Collections	Date Added
another test	3 collections	March 9, 2021
Ewoud test	1 collection	Feb. 24, 2021,
mini pilot accounts	2 collections	Jan. 25, 2021,
mini pilot hashtags	6 collections	Jan. 25, 2021,
public test	2 collections	Jan. 25, 2021,
RabbitMQ	1 collection	March 9, 2021
RabbitMQ Test	3 collections	March 3, 2021
sfm tests	1 collection	Feb. 27, 2021,

Figure 2: A list of existing collection sets the logged-in user can see, with the blue button on the bottom of the page a new collection set can be created.

2.3 Create a collection

In SFM, a collection relates to a specific social media provider and API endpoint, for example, a collection can be of type “Twitter user timeline”, “Twitter search” or “Flickr user”, see also figure 3 for an overview of currently supported collection types.

For each collection one can specify a name, a description and an optional link which could link to a public version of data from this collection. Different types of collections have different options which we explain in the following subsections where we focus on “Twitter user timeline” and “Twitter search” collections.

A collection can be **on, off or inactive**. After creation, a collection is turned off but activated. After seeds are added, the collection can be turned on using the button “Turn on” which means data is collected from the specified seeds in the specified schedule (see next sections). A turned off collection can also be deactivated which should be used to indicate that the data collection is completed. Deactivated collections do not appear in some overview lists nor in harvest status emails. Turned on collections cannot be deactivated.

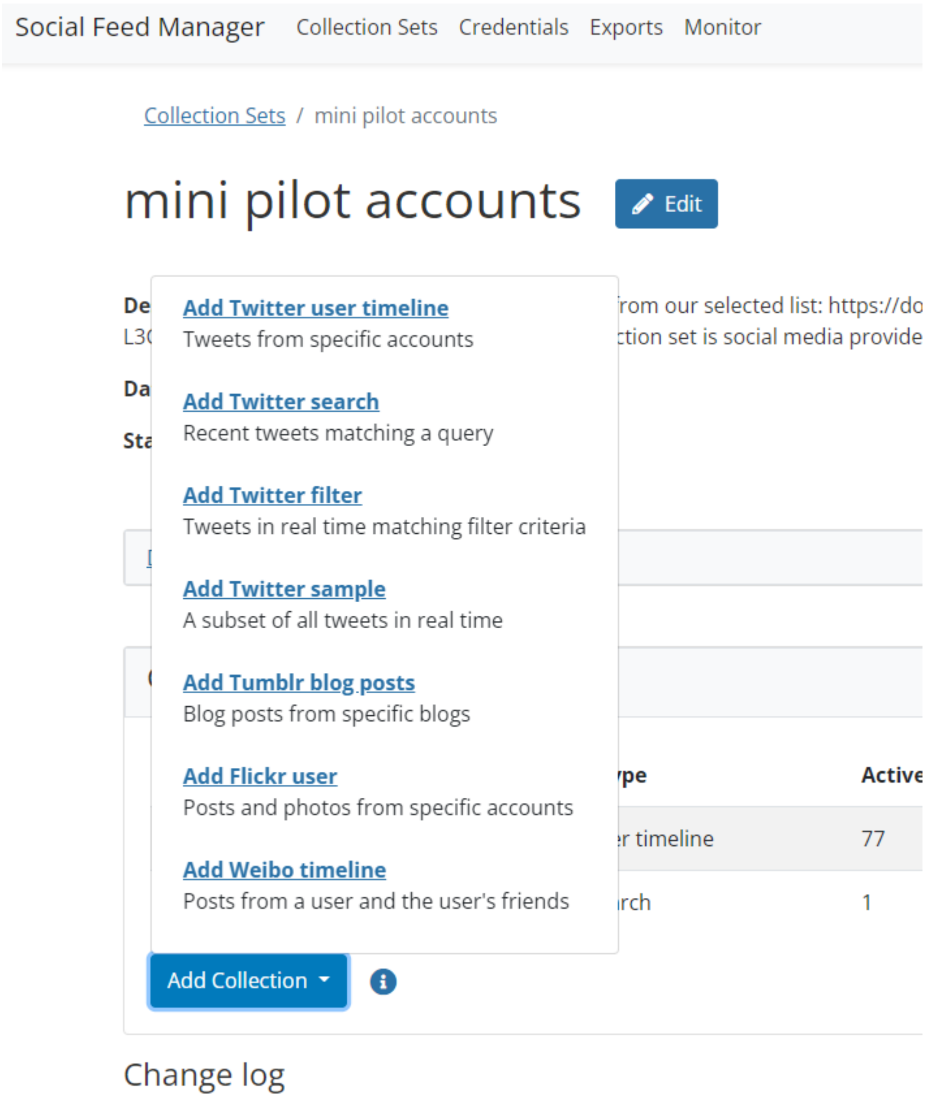

Social Feed Manager
Collection Sets
Credentials
Exports
Monitor
Collection Sets / mini pilot accounts
mini pilot accounts
Edit
De
L3(
Da
Sta
Add Twitter user timeline
Tweets from specific accounts
Add Twitter search
Recent tweets matching a query
Add Twitter filter
Tweets in real time matching filter criteria
Add Twitter sample
A subset of all tweets in real time
Add Tumblr blog posts
Blog posts from specific blogs
Add Flickr user
Posts and photos from specific accounts
Add Weibo timeline
Posts from a user and the user's friends
rom our selected list: https://do
ction set is social media provide
pe
Active
r timeline
77
rch
1
Add Collection
Change log

Figure 3: An overview of all collections of a selected collection set. The button "Add Collection" was pressed which reveals a menu of possible collection types to choose from.

2.3.1 Create a Twitter user timeline collection

A user timeline collection aims to harvest Tweets from specified account seeds where up to 3,200 Tweets can be obtained per account. In figure 4 we show the possible options which we also discuss in this section: select API credentials, provide a harvesting schedule and harvesting behavior, determine access permissions for the collection, save the collection and add seeds.

Used API credentials

A select list shows available credentials which are stored in SFM (see section 2.1). One of the credentials has to be chosen for the collection. Additionally, a warning is shown in case the selected API credentials are already in use for other collections.

Harvesting schedule

A collection can be configured to harvest content from the seeds once or in a specified schedule. A select list shows the options "one time harvest", and seven other possibilities ranging from "30 minutes", to "every 4 weeks". Optionally the date of an end harvest can be provided. Unfortunately, no "starting time" can be provided, thus the schedule is relative to the time you initially activate the collection.

Harvest behavior

When harvesting account content from Twitter, the harvester is notified if an account is protected or (meanwhile) deleted or suspended. One can select with checkboxes if an account should be removed from the seed list if one of these events are detected. Additionally one can specify if the harvest is incremental, i.e. only new content since the last harvest is harvested (recommended).

Access permissions

SFM provides a rudimentary user management where the logged-in user is part of one or more groups. With the "sharing" option, one can specify if only users from the same group can view and export content from the collection or all logged-in users.

Save the collection

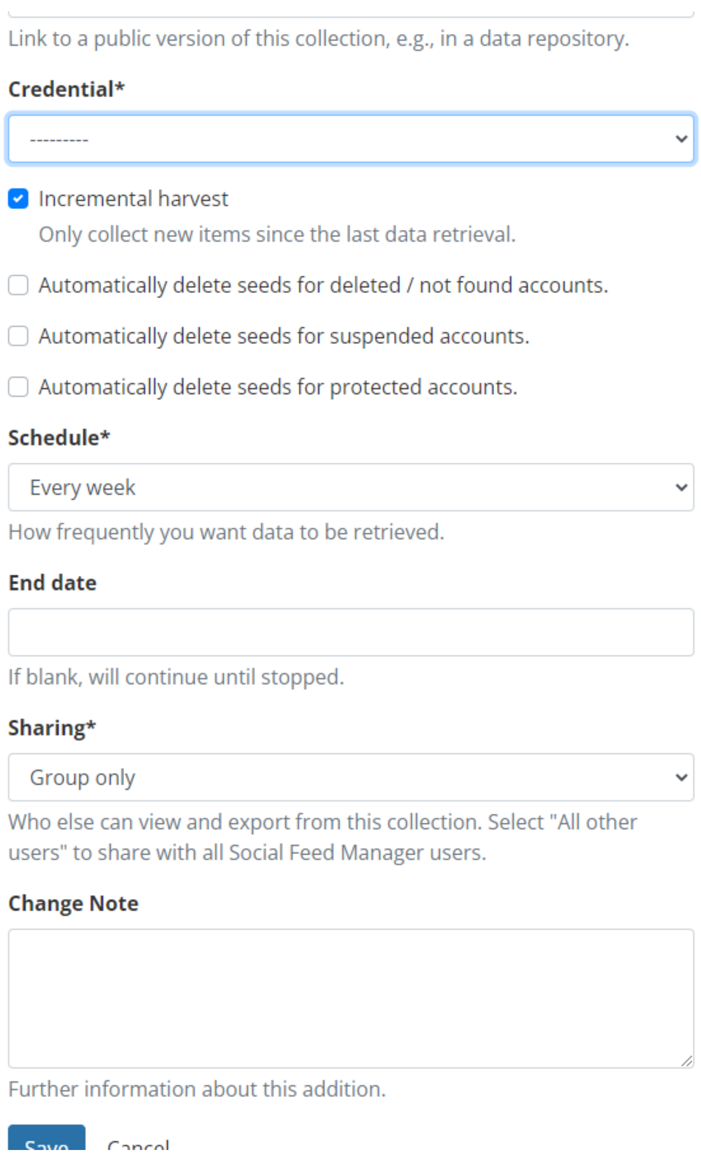
After the collection is created the collection needs to be saved with the blue "save" button on the bottom of the page before seeds can be added to the collection.

Adding account seeds or in bulk

After selecting a collection one can add seeds via the buttons "Add Seed" or "Bulk Add Seeds". In the form which pops up, one can provide seeds in a text field (adding seeds via a configuration file is not possible). When adding a single seed you have to fill in either a screen name or a user ID in the respective fields which appear. In case you provide only the screen name, the user ID will be completed automatically after the first/next harvest. When adding seeds in bulk, you have to choose either screen name or user id for the whole list and then provide one such value per line in the "Bulk Seeds" field which appears. In both cases, an optional change log text can be provided. No limit in the number of accounts which can be provided is documented for SFM[3] nor the underlying Twarc[4].

[3] https://sfm.readthedocs.io/en/stable/collections.html#twitter-user-timeline

[4] http://digitalcollecting.lib.virginia.edu/toolkit/docs/social-media/twarc-commands/#timeline



Link to a public version of this collection, e.g., in a data repository.

Credential*

☑ Incremental harvest
Only collect new items since the last data retrieval.

☐ Automatically delete seeds for deleted / not found accounts.

☐ Automatically delete seeds for suspended accounts.

☐ Automatically delete seeds for protected accounts.

Schedule*

Every week

How frequently you want data to be retrieved.

End date

If blank, will continue until stopped.

Sharing*

Group only

Who else can view and export from this collection. Select "All other users" to share with all Social Feed Manager users.

Change Note

Further information about this addition.

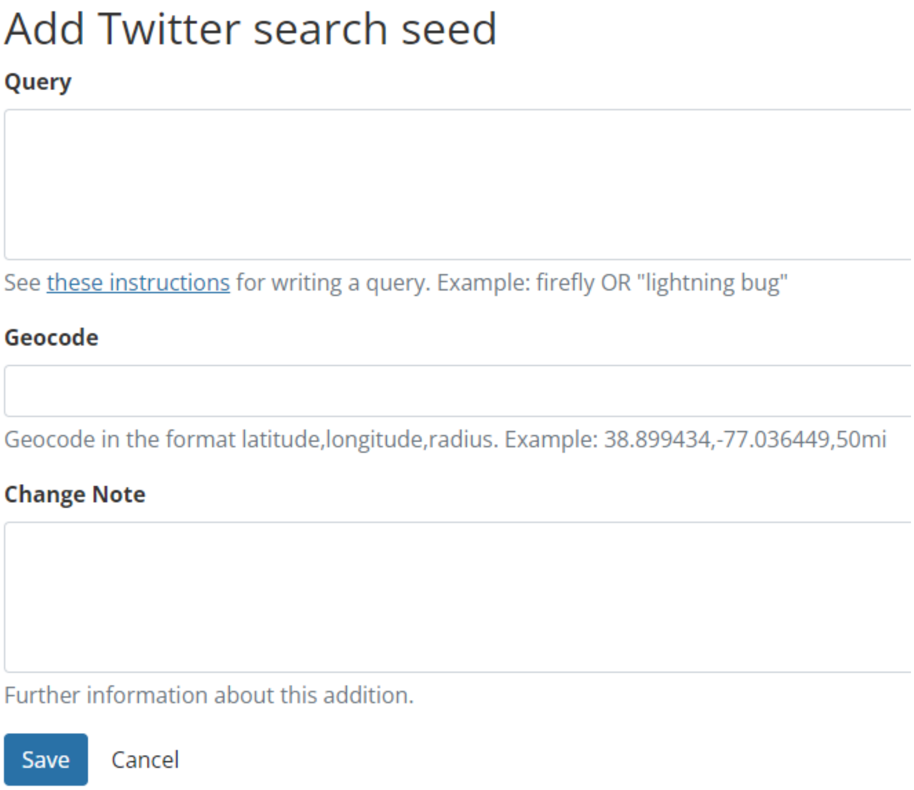
Figure 4: The creation form of a Twitter user timeline collection: API credentials and harvesting options need to be provided, the collection has to be saved and then seeds can be added.

2.3.2 Create a Twitter search collection

A Twitter search collection provides functionality to harvest results of a search query from the last 7 to 9 days. The options to be configured are the same as for user timeline collections (section 2.3.1) but without options for accounts. Thus before continuing, a Twitter search collection needs to be created and saved.

Add Twitter search seeds

In contrast to user timeline collections where several seeds can be provided, a Twitter search collection can only have one seed. A query, geocode or change note can be provided for a seed (see figure 5). The geocode “location is preferentially taking from the Geotagging API, but will fall back to their Twitter profile”[5]. Other operators can be used too, for example to filter for language or negative attitude, a full list of operators is available at the bottom of the Twitter API documentation page[6].



Add Twitter search seed

See these instructions for writing a query. Example: firefly OR "lightning bug"

Geocode in the format latitude,longitude,radius. Example: 38.899434,-77.036449,50mi

Further information about this addition.

Figure 5: The seed adding form for a Twitter search collection: a query containing different Twitter search operators can be provided and optionally also a geocode.

2.4 Starting and monitoring harvests

So far API credentials are provided and collections with seed lists are created, however no harvesting was performed so far. In this section we explain how the harvesting can be started and monitored.

Start harvesting

[5] https://developer.twitter.com/en/docs/twitter-api/v1/tweets/search/api-reference/get-search-tweets

[6] https://developer.twitter.com/en/docs/twitter-api/v1/tweets/search/guides/standard-operators

As explained in section 2.3, a collection is turned off after creation. Using the button “Turn on” on the top right of a collection page, a collection is turned on which means harvests are performed in the specified interval using the specified credentials.

Monitor harvesting

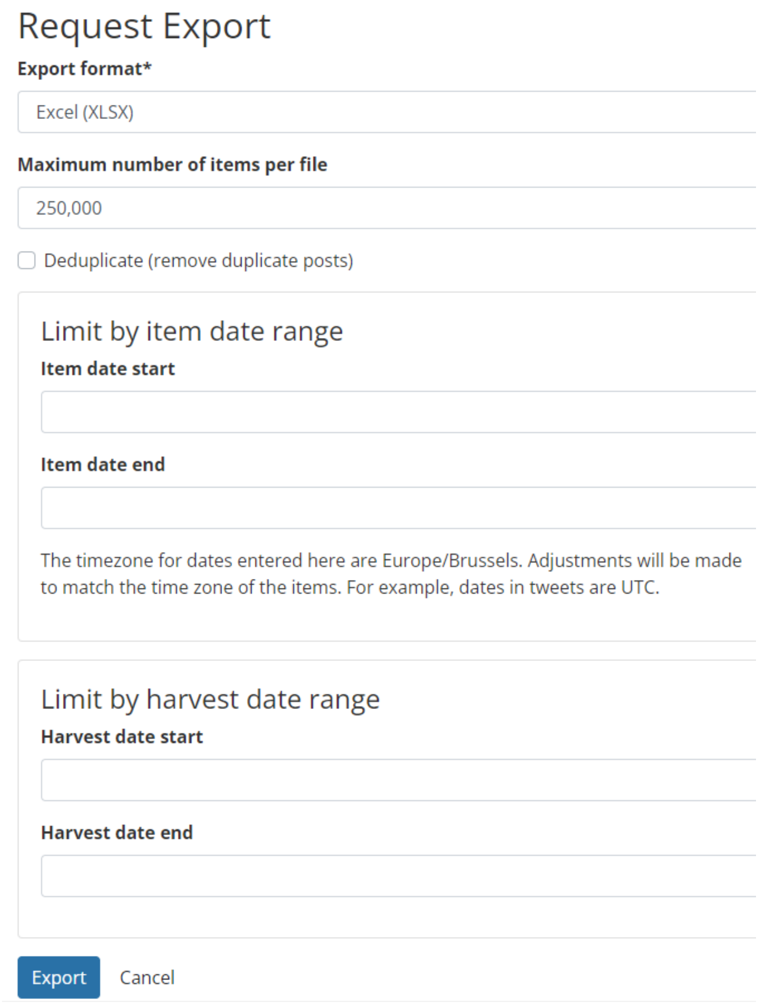
Different ways exist to monitor harvests. Firstly, when clicking the “Monitor” entry in the top navigation bar, the status of the most recent harvests (and exports, see next section) is listed. Secondly, a detailed log of harvests is shown on each collection page below the seed list. Thirdly, harvesting status emails are sent from SFM if configured.

2.5 Export harvesting results

SFM stores all harvested content in WARC files which are stored in collection sets and collection-related subfolders. (Parts of) content of a collection can be exported using the “Export” button on a collection page (see section 2.5.1). There is no direct access to WARC files possible via the SFM user interface. However, the WARC files are stored on the SFM server in a folder structure representing the hierarchy of collection sets and collections, thus WARC files need to be fetched directly from the server’s data directory (see section 2.5.2).

2.5.1 Export collection content via the UI from WARC files

Harvested data can be exported from SFM using the user interface. When the “Export” button on a collection page is pressed, one can configure which content of the collection and how it should be exported. Figure 6 shows the export dialog which we also explain in this section.



Request Export

Export format*

Excel (XLSX)

Maximum number of items per file

250,000

□ Deduplicate (remove duplicate posts)

Limit by item date range

Item date start

Item date end

The timezone for dates entered here are Europe/Brussels. Adjustments will be made to match the time zone of the items. For example, dates in tweets are UTC.

Limit by harvest date range

Harvest date start

Harvest date end

Figure 6: The export dialog which appears when clicking "Export" on a collection page. One can configure the export before requesting its automatic creation using the blue "Export" button at the bottom.

Which content

Selections regarding seeds and duplicate entries can be made. On the one hand, one can select which seeds should be exported using the "Seed choice" radio button. Either all seeds, only active seeds, or multiple choice selected seeds can be selected. On the other hand, one can use the "Deduplicate" checkbox to indicate that possibly duplicate posts should be removed for the export. Please note: this export functionality considers a single collection, yet duplicate tweets might be possible for certain collection types, hence the checkbox.

Which format and how many posts per file

It can be configured how many social media posts should be exported per file as well as the format of the file. With respect to the file format, one can select that only tweet IDs should be exported in a text file (dehydration) which is a common exchange format allowed by Twitter. Besides this, content can be exported in tabular formats, i.e. Excel (XLSX), Comma Separated Values (CSV) and Tab Separated Values

(TSV), or in JSON format, i.e. the full JSON content as obtained from the API or a JSON with less fields. With respect to the number of export files, one can select to export everything in one file, or having 250k, 500k or 1mio posts per file.

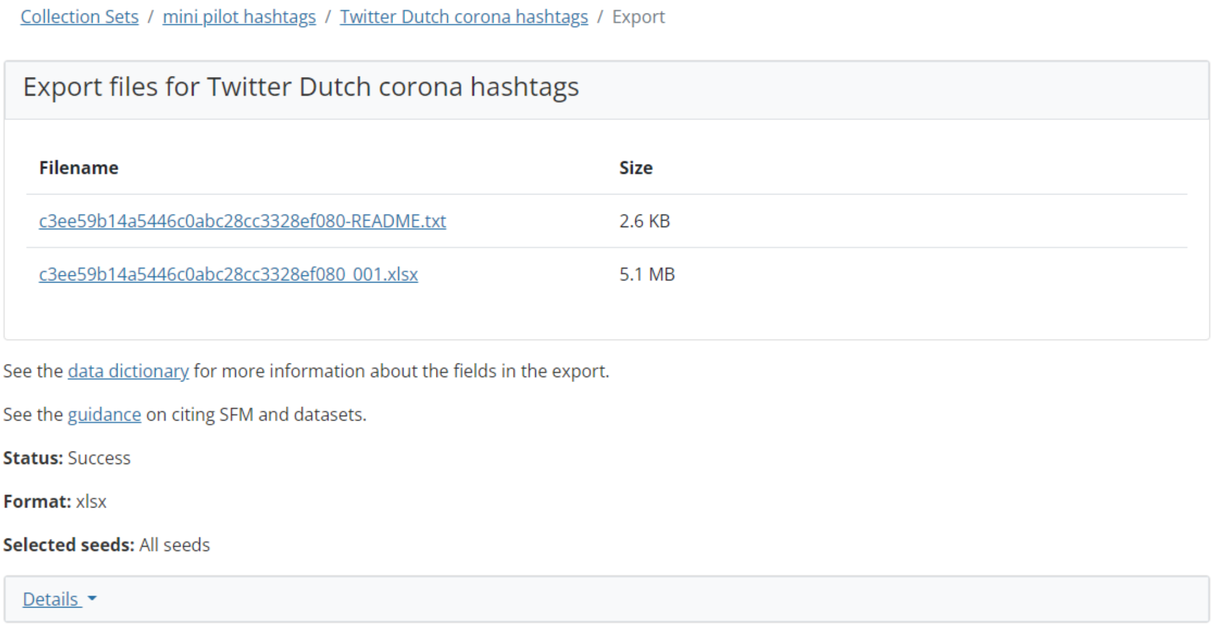
Date range limitation

The export can be limited in date range either by the creation date of the social media posts or by the harvesting date. For example, since a Twitter user timeline collection harvests up to 3,200 tweets per account it could be that Tweets from the distant past are collected. One may limit the date range with respect to the creation date of Tweets, but also based on the date of the harvest execution.

After clicking the export button, the export is prepared.

Downloading the export

SFM extracts content from WARC files and stores the files in the selected format for download on the server. One can download finished exports via the user interface (also including previously created exports). When clicking on "Exports" of the top navigation bar, one can see a list of requested exports, their status and when they were requested. When clicking on such an export one is forwarded to an overview page of this export, the actual file can be downloaded by clicking on the file



Collection Sets / mini pilot hashtags / Twitter Dutch corona hashtags / Export

Export files for Twitter Dutch corona hashtags

Filename	Size
c3ee59b14a5446c0abc28cc3328ef080-README.txt	2.6 KB
c3ee59b14a5446c0abc28cc3328ef080_001.xlsx	5.1 MB

See the data dictionary for more information about the fields in the export.

See the guidance on citing SFM and datasets.

Status: Success

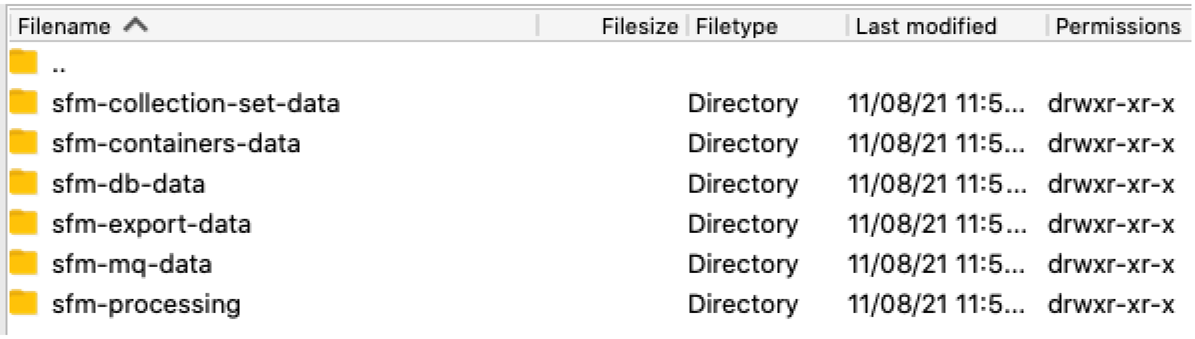
Format: xlsx

Selected seeds: All seeds

Figure 7: The overview page of a requested export. If successful, one can download (i) the export by clicking on the filename, and (ii) meta information of the export by clicking on the filename with README.txt postfix.

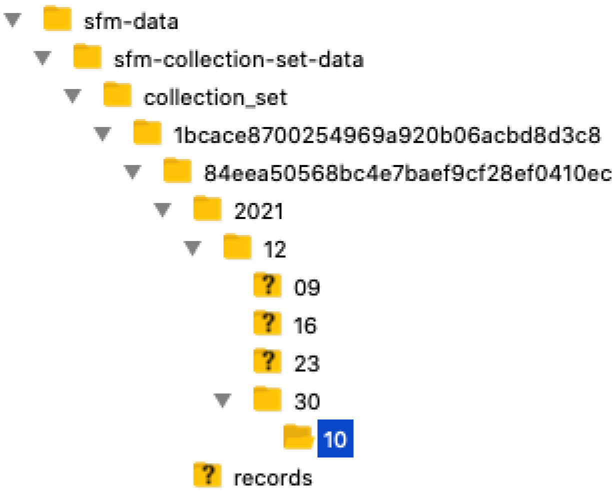
2.5.2 Export collection WARC files

WARC files cannot be exported via the UI, but can be retrieved directly from the SFM servers data directory. Figure 8a shows a screenshot of SFM's data directory structure. Different standard tools can be used to access the files on the server, we describe how to retrieve WARC files using the tool FileZilla[7] which provides a user interface (see figures 9 and 10). Side note: collection exports requested via the SFM UI (section 2.5.1) are also just stored in this data directory, the download link presented in the UI simply points to the created export file in the export directory.



Filename ^	Filesize	Filetype	Last modified	Permissions
..				
sfm-collection-set-data		Directory	11/08/21 11:5...	drwxr-xr-x
sfm-containers-data		Directory	11/08/21 11:5...	drwxr-xr-x
sfm-db-data		Directory	11/08/21 11:5...	drwxr-xr-x
sfm-export-data		Directory	11/08/21 11:5...	drwxr-xr-x
sfm-mq-data		Directory	11/08/21 11:5...	drwxr-xr-x
sfm-processing		Directory	11/08/21 11:5...	drwxr-xr-x

Figure 8a: An illustration of the folder structure of SFM's data directory

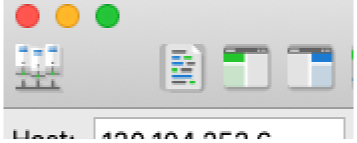


```
▼ sfm-data
  ▼ sfm-collection-set-data
    ▼ collection_set
      ▼ 1bcace8700254969a920b06acbd8d3c8
        ▼ 84eea50568bc4e7baef9cf28ef0410ec
          ▼ 2021
            ▼ 12
                09
                16
                23
              ▼ 30
                  10
            records
```

Figure 8b: An illustration of the folder structure of SFM's collection sets

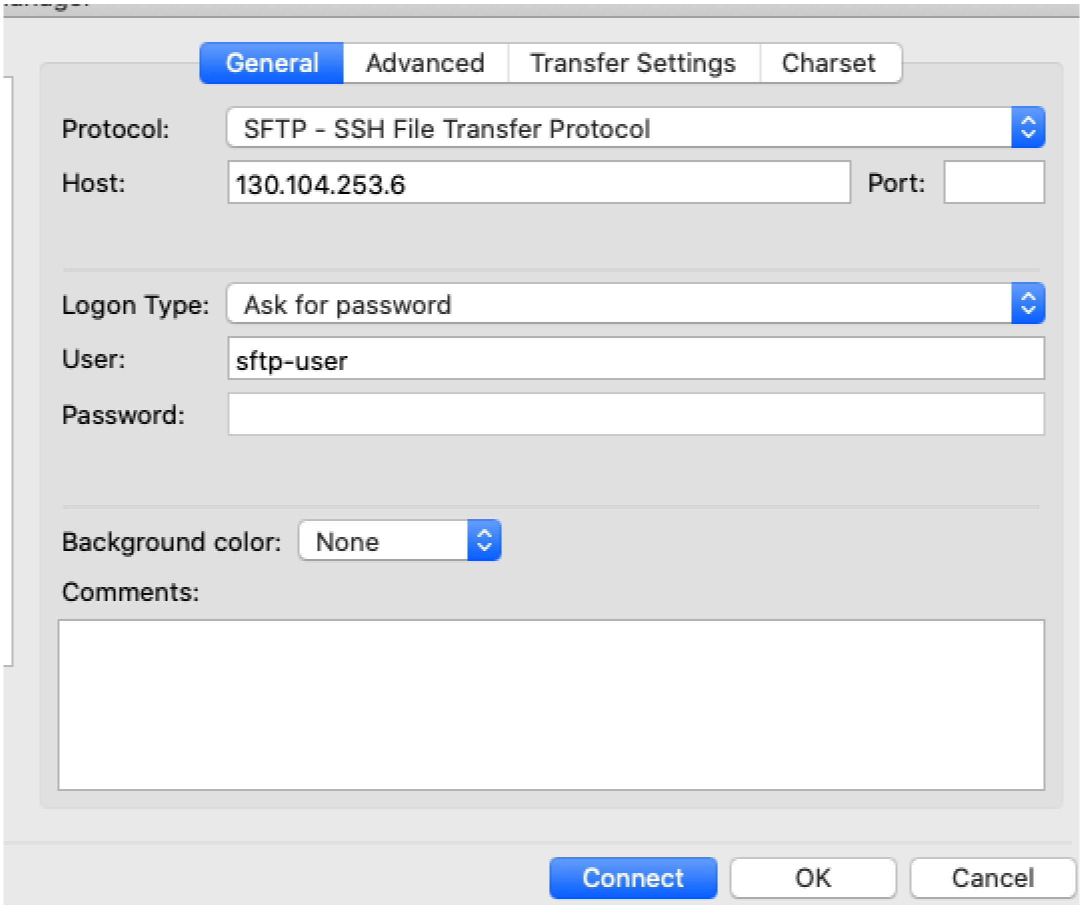
As seen in figure 8b, SFM's data about collections is hierarchically organized: collection set folders contain folders for collections which contain folders per year and month which contain the actual compressed WARC files. Since names of collections may change, the names of the folder are based on unique identifiers. Metadata about these collections is stored in the SQL database of SFM, but also as JSON files within the "records" subfolder of a collection.

[7] https://filezilla-project.org/



Host: 130.104.253.6

Figure 9a: Creating a new connection via FileZilla: first click on the icon on the top left to open the Site Manager.

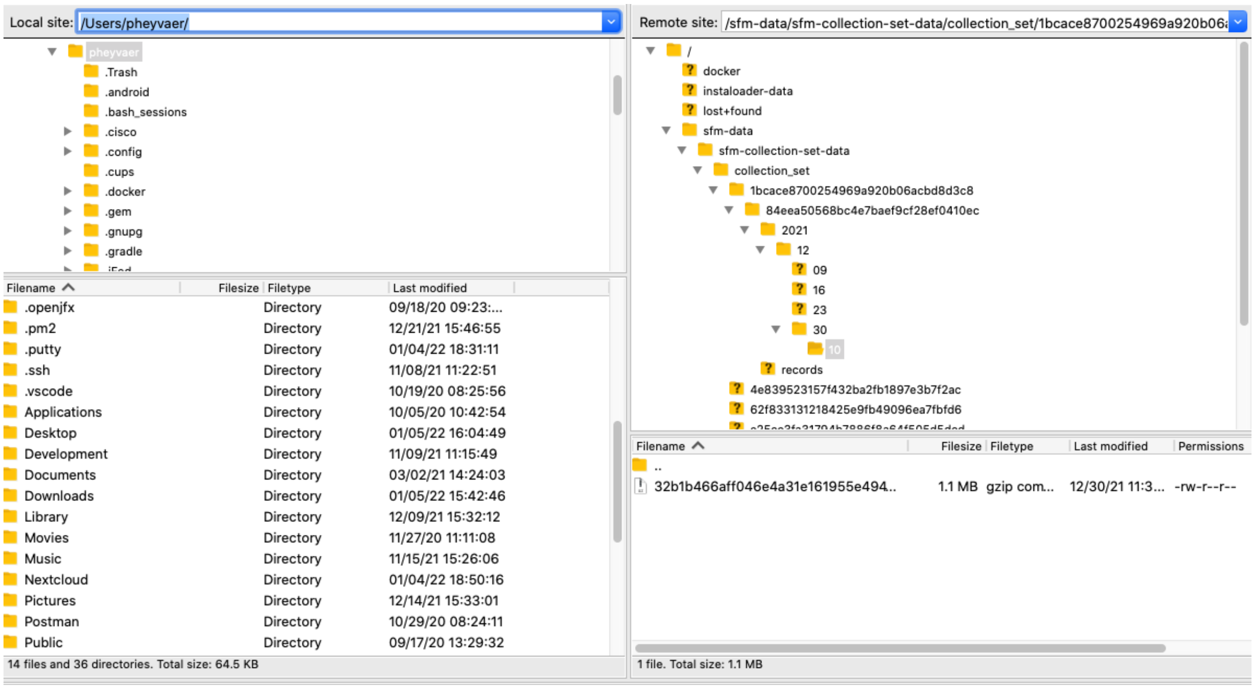


Protocol: SFTP - SSH File Transfer Protocol
Host: 130.104.253.6 Port:
Logon Type: Ask for password
User: sftp-user
Password:
Background color: None
Comments:

Figure 9b: Creating a new connection via FileZilla: one has to select the protocol, hostname or IP address, username and password.

The data directory of SFM on the server can be accessed using the free tool FileZilla. When opening the tool you click on the icon on the top left to open the Site manager (figure 9a). A new window opens in which connection information needs to be entered (figure 9b). The protocol is SFTP. The host is 130.104.253.6 The user is “sftp-user”. Next, click on “Connect” at the bottom of the window.

After the connection is established you see the directories of your computer on the left side and the servers directories on the right side (figure 10). You need to locate the sfm-data directory on the right side and based on the folder hierarchy of SFM (figure 8b) you can find the WARC files you would like to export. To download the data you drag and drop the file from the right (server) to your computer (left).



Local site: /Users/pheyvaer/

```
▼ pheyvaer
    .Trash
    .android
    .bash_sessions
  ▶ .cisco
  ▶ .config
    .cups
  ▶ .docker
  ▶ .gem
  ▶ .gnupg
  ▶ .gradle
```

Filename	Filesize	Filetype	Last modified
.openjfx		Directory	09/18/20 09:23:...
.pm2		Directory	12/21/21 15:46:55
.putty		Directory	01/04/22 18:31:11
.ssh		Directory	11/08/21 11:22:51
.vscode		Directory	10/19/20 08:25:56
Applications		Directory	10/05/20 10:42:54
Desktop		Directory	01/05/22 16:04:49
Development		Directory	11/09/21 11:15:49
Documents		Directory	03/02/21 14:24:03
Downloads		Directory	01/05/22 15:42:46
Library		Directory	12/09/21 15:32:12
Movies		Directory	11/27/20 11:11:08
Music		Directory	11/15/21 15:26:06
Nextcloud		Directory	01/04/22 18:50:16
Pictures		Directory	12/14/21 15:33:01
Postman		Directory	10/29/20 08:24:11
Public		Directory	09/17/20 13:29:32

14 files and 36 directories. Total size: 64.5 KB

Remote site: /sfm-data/sfm-collection-set-data/collection_set/1bcace8700254969a920b06a

```
▼ /
    docker
    instaloader-data
    lost+found
  ▼ sfm-data
    ▼ sfm-collection-set-data
      ▼ collection_set
        ▼ 1bcace8700254969a920b06acbd8d3c8
          ▼ 84eea50568bc4e7baef9cf28ef0410ec
            ▼ 2021
              ▼ 12
                  09
                  16
                  23
                ▼ 30
                    10
              records
          4e839523157f432ba2fb1897e3b7f2ac
          62f833131218425e9fb49096ea7fbfd6
```

Filename	Filesize	Filetype	Last modified	Permissions
..				
32b1b466aff046e4a31e161955e494..	1.1 MB	gzip com...	12/30/21 11:3...	-rw-r--r--

1 file. Total size: 1.1 MB

Figure 10: The interface of FileZilla after the connection to a server is established: the local computer is on the left side and the directories of the remote server on the right. The shown WARC file (or whole folders) can be copied to the local computer via drag-and-drop functionality.

2.6 Harvesting and exporting images

Out of the box, SFM does only harvest JSON files from Twitter, thus no images. However, images can be harvested in a separate step using utility scripts from the SFM underlying Twitter harvester Twarc[8]. We describe how an adapted version of the Twarc utility script `media2warc.py`[9] can be used to harvest images as WARC files. The script was adapted such that it can be configured if videos should be excluded from harvesting; videos are not in scope of BESOCIAL.

Harvesting

The input of the tool are line-based JSON files (.json, .jsonl) or alternatively compressed line-based json files (.json.gz). Therefore for an image harvest, the relevant JSON data needs first to be extracted from SFM (see section 2.5.1). To harvest the images of such a JSON file you can use the python command of listing 1 which will download all images from the tweets excluding videos and profile media (namely

[8] https://github.com/DocNow/twarc
[9] https://github.com/SvenLieber/twarc/blob/feature/media-download-filter/utils/media2warc.py

profile picture and profile background image[10]). We do not prescribe where you execute this command, you can either copy the JSON file to your own computer and download/install the `media2warc.py` script, or you can connect to the server to use an installed version there.

```
# it is recommended to execute the following command in its own virtual
python environment

python media2warc.py --no-videos --no-profile-media <json file>
```

Listing 1: Downloading images listed in Twitter JSON files and store them in WARC files.

Exporting

The media2warc.py script will download images and store them in a WARC File. This WARC file can be downloaded to your computer as described for other server-side files (see section 2.5.2). To get the actual images you can either use a tool such as warcio[11] to extract content from WARC files, or use another harvesting script in the first place. An alternative to `media2warc.py` is the script `media_urls.py`[12] which will extract all links of media to a text file which can then be downloaded in a subsequent step using for example the linux command line tool `wget`. Please note, when using `media_urls.py` instead of our adapted version of `media2warc.py` you also get links for videos and profile-related images. Similarly to `media2warc.py` you could also adapt this script to only extract non-profile and non-video media links.

3 Instagram - Instaloader

In contrast to Twitter, there exists no straightforward functionality to harvest data via an API. There is currently also no support for Instagram harvesting as part of SFM. However, the tool Instaloader[13] can be used to harvest content from Instagram using a command line interface. We explain how the tool can be configured with login credentials and how different types of harvests can be performed in a repeating schedule[14]. For this section we assume that you are connected to the server on which Instaloader is installed.

The Instaloader tool can be executed with different command line options to specify which seeds to harvest and where to store it. Harvested content is stored in folders following a name structure which can be configured, usually consisting of the seed name. Based on collected metadata inside these

[10] For BESOCIAL it was chosen NOT to harvest videos or profile pictures
[11] https://github.com/webrecorder/warcio
[12] https://github.com/SvenLieber/twarc/blob/feature/media-download-filter/utils/media_urls.py
[13] https://instaloader.github.io/
[14] More documentation about Instaloader is also available as part of our tool testing activity: https://github.com/RMLio/social-media-archiving/tree/master/harvesting-tool-comparison/instaloader

directories and certain command line options, Instaloader will continue harvesting “where it stopped” for each given seed-based content folder.

3.1 Provide login credentials

The harvester can be used without Instagram login but there are several limitations: a stricter rate limiting compared to logged-in users, pictures only in low quality, and additionally for some search queries one needs to be logged in. We recommend login one time and then use a created session cookie for further authentication. In this section we introduce different ways to login and related information. Please note that the first login method via command line may result in errors, using the second metho should avoid this issue.

First time login via command line

Login details can be provided using the `--login` option, see listing 1. After successful authentication, a session cookie is stored as a file in the same directory which avoids that username and password need to be provided the next time the tool is used with the `--login` option. By default the file is stored in ~/.config/instaloader-<name>. But you can provide another location to store the session using the option `--sessionfile` (when using this option without `--login`, it is interpreted as the location of an existing session file which should be used for login).

```
instaloader --login <username> –sessionfile <path-to-store-file>
```

Listing 2: Logging in with Instaloader which creates a session cookie avoiding that for future calls username and password need to be provided.

First time login via Browser and cookie export

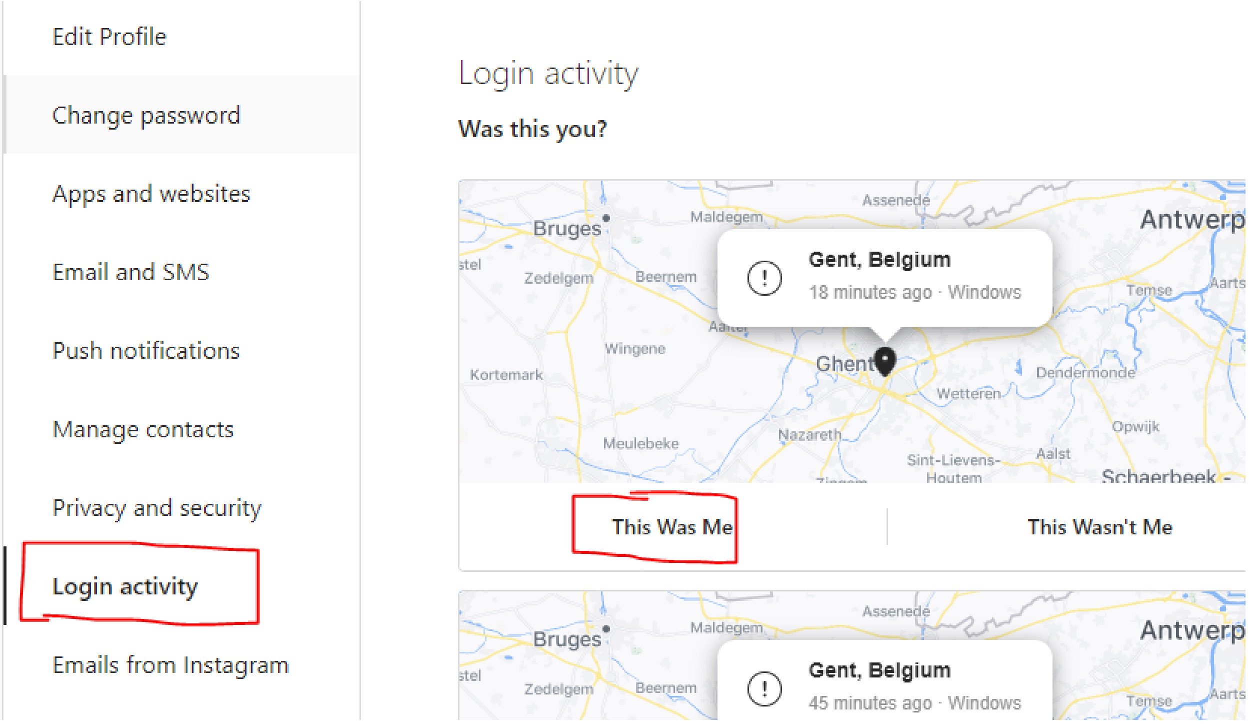
You can login to your Instagram account using the regular Instagram website. Using a Browser extension such as EditThisCookie[15], you can export all stored Instagram cookies created after you login in JSON format. You can copy/paste this JSON content in a file which you can use via the `--sessionfile` command line option *without* specifying `--login`.

Stay logged-in

After logging in via either of the methods, it might be helpful to login to the used Instagram profile and confirm in the login history that this login was indeed you, see figure 11. This may help preventing that the account is blocked frequently[16].

[15] https://www.editthiscookie.com/

[16] https://github.com/instaloader/instaloader/issues/828#issuecomment-704418101



Login activity

Was this you?

Gent, Belgium
18 minutes ago · Windows

Gent, Belgium
45 minutes ago · Windows

Figure 11: After your initial Instaloader login, confirm that this was you via the Instagram menu Account->Settings->Login activity.

3.2 Create a collection

The tool itself does not know the concept of a collection, but different collections can be realized by using different configuration files. Since there is no straightforward API, Instagram is mainly about images and the tool has to take rate limitations into account, even harvests of a small number of accounts may take significantly longer compared to Twitter. Therefore, we recommend creating smaller collections and leaving enough time in between harvests. In this section we exemplify how two different collections can be created: a user timeline collection and a hashtag search collection. Each collection is realized using a respective configuration file, see listing 3 and 5 which is executed as shown in listing 4.

3.2.1 Create an Instagram user timeline collection

To harvest posts from selected user accounts simply a list of the account names needs to be provided. In listing 3 we list the content of a configuration file consisting of profile names and command line options.

```
--sessionfile your-session-cookie.json
--dirname-pattern=”myUserTimelineCollection/{target}”
--latest-stamps ./myUserTimelineCollection.ini
```

```
--no-profile-pic
--no-video-thumbnails
--no-videos
profile1
profile2
profile3
profile4
```

Listing 3: An Instaloader configuration file which can be used to harvest the specified profiles with the specified options.

The file of listing 3 can be used using the command in listing 4. Using the file in listing 3, all provided seeds (profile1 - profile 4) are stored within the *myUserTimelineCollection* folder. Furthermore, *no videos* or *profile images* are harvested and the tool will use a metadata file with timestamps to keep track which content was already harvested.

```
Instaloader +myUserTimelineCollection.txt
```

Listing 4: Calling Instaloader with pre-configured command line options, thus realizing “a collection”.

3.2.2 Create an instagram search collection

For hashtags you use the same approach as above for profiles, but you use ‘#hashtag1” instead of “profile1”.

3.3 Starting and monitoring harvests

The previously created configuration files can be used to start Instagram harvests. You can find a script combining the aforementioned approach on the server at “/home/pieter/instaloader-scripts”. You find more information about the script in “README.md” in that folder. A cron job[17] [18] is found in “cron-job.sh” it is executed once a week. The details of the job on the server are “44 12 * * 1 /home/pieter/instaloader-scripts/cron-job.sh &> /home/pieter/instaloader-scripts/cron-job.log” and are set via “crontab -e”. You find the logs outputted by the script at “/home/pieter/instaloader-scripts/cron-job.log”

[17] https://en.wikipedia.org/wiki/Cron

[18] https://github.com/instaloader/instaloader/issues/828#issuecomment-822715928

3.4 Export harvesting results

Harvested content from Instagram, photos and metadata files, are directly stored in folders. Thus, you export the results by copying them via FileZilla, analogous to the SFM WARC files. The folder where all the Instagram data is stored is called “instaloader-data”.