Article

Redundancy Reduction in Twitter Event Streams

Nane Kratzke

Lübeck University of Applied Sciences; nane.kratzke@th-luebeck.de

Abstract: The data from social networks like Twitter is a valuable source for research but full of redundancy, making it hard to provide large-scale, self-contained, and small datasets. The data recording is a common problem in social media-based studies and could be standardized. Sadly, this is hardly done. This paper reports on lessons learned from a long-term evaluation study recording the complete public sample of the German and English Twitter stream. It presents a recording solution proposal that merely chunks a linear stream of events to reduce redundancy. If events are observed multiple times within the time-span of a chunk, only the latest observation is written to the chunk. A 10 Gigabyte Twitter raw dataset covering 1,2 Million Tweets of 120.000 users recorded between June and September 2017 was used to analyze expectable compression rates. It turned out that resulting datasets need only between 10% and 20% of the original data size without losing any event, metadata or the relationships between single events. This kind of redundancy reduction recording makes it possible to curate large-scale (even nation-wide), self-contained, and small datasets of social networks for research in a standardized and reproducible manner.

Keywords: Twitter, dataset, redundancy, reduction, archive

1. Introduction

The data-drivenness and systematic analysis of social media data get more and more common in the (social) sciences and is, besides other domains, applied to understand the influence of social media on our everyday life. E.g. Barberá and Rivero emphasize *"the opportunities offered by Twitter for the analysis of public opinion: messages are exchanged by numerous users in a public forum, and they may contain valuable information about individual preferences and reactions to different (...) events in an environment that is fully accessible to the researcher" [1].*

Twitter provides samples of these data for free via its streaming APIs. At least for large datasets [2], these samples *"truthfully reflect the daily and hourly activity patterns of the Twitter users (...) and preserve the relative importance (...) of content terms"* [3]. So, although Twitter might not be the biggest influencing social network, it is with more than 330 million monthly, and approximately 145 million daily active users in 2019 a valid and free to use data source for research. Therefore, Twitter data has been used for a variety of interesting studies:

- Analysis of the political representativeness of Twitter users [3]
- Real-time Twitter analysis [4–6]
- Democratic elections [7–13]
- The uses of Twitter by populists [14]
- Misinformation dissemination and event detection in social networks [15–17]
- Online public shaming [18]
- and many more

Furthermore, there exist plenty of public available datasets. E.g., a curated collection [19] is hosted on Zenodo [20] and covers several datasets of political campaigns [8,9], online misinformation networks [21], event detection in temporal networks [22], public shaming [23], Twitter-related word vectors [24], retweeting timeseries [25], and even continuously updated samples of a nations-wide Twitter usage [26].

So, there is no problem with a lack of data. The problem lays more in the variety of data, and in the variety of different collection methodologies and software stacks used to collect the data. The mentioned datasets are provided in varying formats (CSV, JSON, TXT, XML, and further proprietary – even binary – data formats). What is more, almost none of the above-referenced studies reported in repeatable details what the exact methodology and tool-suite was to archive the data? This mix of data formats and vaguely described collection methodologies makes it hard to compare different studies and datasets. However, we can distinguish two major data collection approaches:

- Streaming approaches like the programmable Tweepy API [27] or BotSlayer [28] make use of the Twitter streaming API. They make it possible to record large amounts of data in real-time. However, the filtering must be specified in the upfront of a study - which can be tricky if the specific filtering can be hardly predicted. E.g. in case of sudden and unpredictable events like earthquakes [5] or terror attacks. However, the programmable libraries let plenty of room for very proprietary recording solutions that are hardly reproducible by other researchers.
- Scraping approaches like TWINT [29] make use of Web scraping techniques. Because of the scraping approach, these tools are more limited regarding the amount of recordable data. However, the data can be collected even in the aftermath of an event. Because they are "backwards"-looking, they are hard to analyze real-time effects. What is more, scraping technologies need an initial set of search terms or user accounts to start scraping. A slightly different set of search terms may result in substantially different datasets. So, scraping based approaches are vulnerable for unaware and non-obvious biases. Therefore, it is hard to use them to collect datasets the can be used as an objective and unbiased "ground-truth" of a social network.

So, to create a large-scale, objective, and un-biased "ground-truth," it is necessary to archive even nation-scale social-network traffic with as less up-front filters as possible. Thus, the resulting archives can be filtered for research questions in the aftermath of events independently by different researchers. However, the resulting sizes of datasets must be manageable and reasonable. This paper focuses on the inherent redundancy of social network event streams that are mainly repeating content. If archiving solutions would effectively eliminating the intrinsic redundancy in social network streams, large-scale, self-contained, (relatively) small datasets would become possible, that could be used for a broad range of research. A long-term evaluation study archiving the German Twitter stream demonstrates that this is possible. The monthly updated dataset is provided as open-source and might be inspected by the reader [26].

The rest of the paper is **outlined** as follows. Section 2 explains the overall problem that the inherent redundancy of Social network event streams makes it necessary to find solutions for self-containing but redundancy-reduced data formats for archiving. It proposes furthermore a redundancy reduction solution that can be configured using different chunk sizes of events. This solution proposal is implemented in a recording solution called Twista [30]. The reader might inspect Twista on GitHub. Section 3 explains the methodology of the evaluation of Twista's recording capabilities that are based on replaying already existing raw data of social network event streams. Section 4 shows different compression rate results. It turns out that social network event streams can be archived consuming less than 20% of the raw data space without losing information. Section 5 discusses the results critically and addresses internal and external threats on the validity of this study. The paper closes with a conclusion on the results of this study in Section 6.

2. Problem Statement

The Twitter content dissemination mechanics is well structured. Figure 1 presents the conceptual metamodel as a UML class diagram.

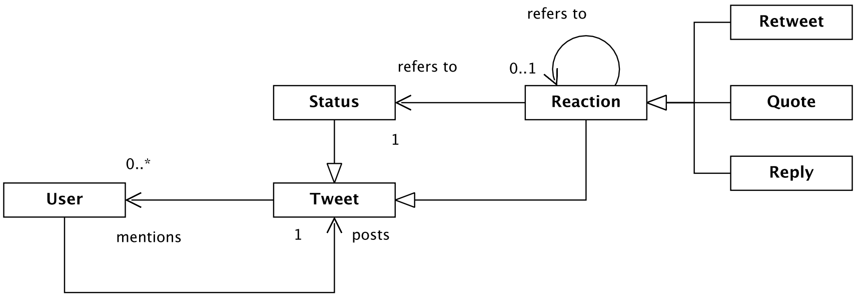


Figure 1. UML data model of Twitter events

Every user interaction on Twitter starts with a `Status` post. `Status` posts are used to disseminate some updates or news. Other users are informed about such `Status` posts and can interact using special kind of `Reactions` to comment on or support the dissemination of `Tweets`. These `Reactions` refer to this initial `Status` or observed follow-up `Reactions`. Such `Reactions` that flow as events through the streaming API are:

- `Retweet` (this is used to broadcast other `Tweets` to own followers; usually this expresses a kind of support for the content)
- `Quote` (similar to a `Retweet`, but own content is attached to the original `Tweet`; this might change the tenor of the original tweet, e.g. by sarcastic comments)
- `Reply` (to comment on different kind of posts; this can be supportive, neutral or contradictive comments)

A further reaction is a "Like". A "Like" expresses some support for the content of a `User`. Nevertheless, "Likes" do not flow as events through the Twitter streaming API. So, for this paper, "Likes" are not considered. However, each event contains metadata that counts how many "Likes" a Tweet got. So, "Likes" are recorded and can be analyzed, although they are not flowing as events through the Twitter streaming API.

Except for `Replies` (for Twitter API historical and backward compatibility reasons), `Quotes` and `Retweets` contain the referring content completely. For instance, a `Quote` Q of a `Retweet` R of a `Status` S contains R and S (the `Retweet` R that includes the `Status` S). On the Twitter streaming API, this looks like a linear stream of events (see Figure 2, TOP). However, there exist referrals between some events. Figure 2 (BOTTOM) presents a more precise representation of the situation. Moreover, every refer-to link is accompanied by redundancy (the referring event always includes the referred event). This redundancy is very comfortable for (mobile) streaming applications because every streaming event contains its full context. So, no expensive follow-up queries are necessary to fetch the context of a Tweet. However, for archiving Twitter content, this self-containing is a redundancy nightmare. According to studies focussing on political campaigning [9], almost 2/3 of all Tweets are `Retweets` or `Quotes`, and only 6% are `Status` posts (see Figure 7). Although these percentages depend to some degree on the language (German, English, etc.) or the context (political campaigns, general, lifestyle, etc.), they can be observed to some degree similarily in different contexts (see Figure 7). So, more than 60% of all streaming events repeat 6% of the original content. That is a factor of 10!

doi:10.20944/preprints202002.0170.v1

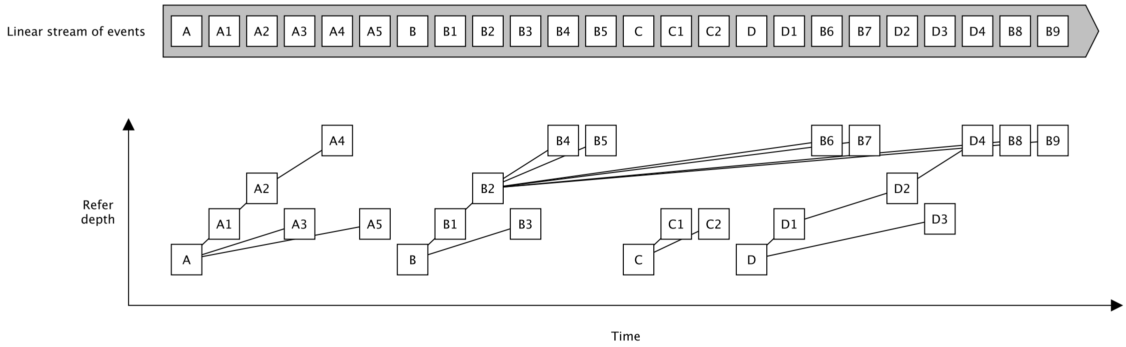


Figure 2. Stream of Twitter events

However, this provides plenty of opportunities to reduce redundancy. Like the original streaming API, we want to record events as chunks of a self-contained set of events but without unnecessary redundancy. Twista is doing this via its recording component by logging a Twitter stream as a linear stream of events. Every n-th event (a chunk), all so far recorded events in this chunk are written to a log, but duplicate events are eliminated. So each log contains unique events, but all referrals between these events are preserved. Figure 3 shows this effect by a constructed example for different chunk sizes. So, we can create much smaller records that are still self-contained (containing all referrals).

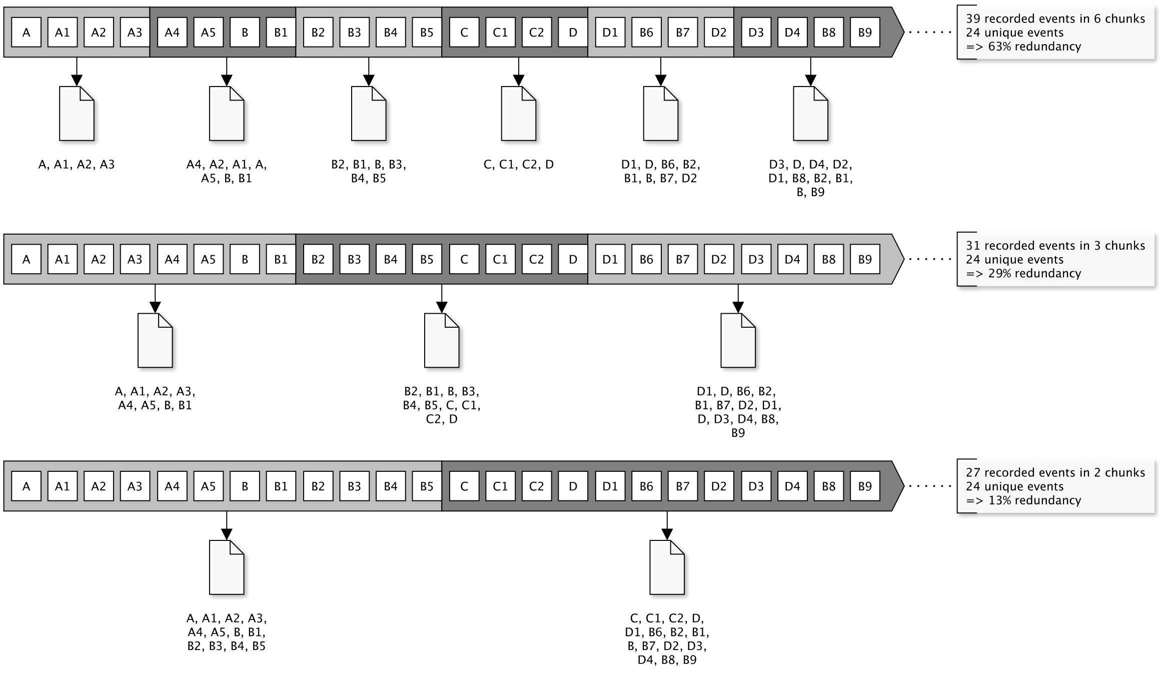

A A1 A2 A3 A4 A5 B B1 B2 B3 B4 B5 C C1 C2 D D1 B6 B7 D2 D3 D4 B8 B9
39 recorded events in 6 chunks
24 unique events
=> 63% redundancy
A, A1, A2, A3
A4, A2, A1, A, A5, B, B1
B2, B1, B, B3, B4, B5
C, C1, C2, D
D1, D, B6, B2, B1, B, B7, D2
D3, D, D4, D2, D1, B8, B2, B1, B, B9
A A1 A2 A3 A4 A5 B B1 B2 B3 B4 B5 C C1 C2 D D1 B6 B7 D2 D3 D4 B8 B9
31 recorded events in 3 chunks
24 unique events
=> 29% redundancy
A, A1, A2, A3, A4, A5, B, B1
B2, B1, B, B3, B4, B5, C, C1, C2, D
D1, D, B6, B2, B1, B7, D2, D1, D, D3, D4, B8, B9
A A1 A2 A3 A4 A5 B B1 B2 B3 B4 B5 C C1 C2 D D1 B6 B7 D2 D3 D4 B8 B9
27 recorded events in 2 chunks
24 unique events
=> 13% redundancy
A, A1, A2, A3, A4, A5, B, B1, B2, B3, B4, B5
C, C1, C2, D, D1, B6, B2, B1, B, B7, D2, D3, D4, B8, B9

Figure 3. Effect of different chunk sizes, larger chunks reduce redundancy

Figure 3 shows furthermore that this redundancy reduction is correlated with the chunk size. The redundancy is shrinking for larger chunk sizes. However, Figure 3 is only presenting a constructed example to demonstrate the effect. The question is, whether this principle holds for real-world datasets? We will address these questions in the following Sections.

3. Methodology

The principle demonstrated in Figure 2 and 3 has been implemented in Twista [30] to measure the effect of redundancy elimination. The reader can study the straightforward implementation of the recording component (`twista/recorder.py`) on GitHub to introspect the solution proposal.

In a second step, some Twitter streaming API raw data has been selected for evaluation. For this study, it was decided to work with the #BTW17 dataset [9]. It compromises approximately 10 GB raw data of German tweets (1GB if compressed as ZIP archive), that has been recorded while the political campaigns for the 2017 German Federal Elections (Bundestag). Other raw datasets would have been possible as well. However, no further documented Twitter streaming raw data sets in public dataset repositories were found. The collection of the #BTW17 dataset is explained and described detailly in [8].

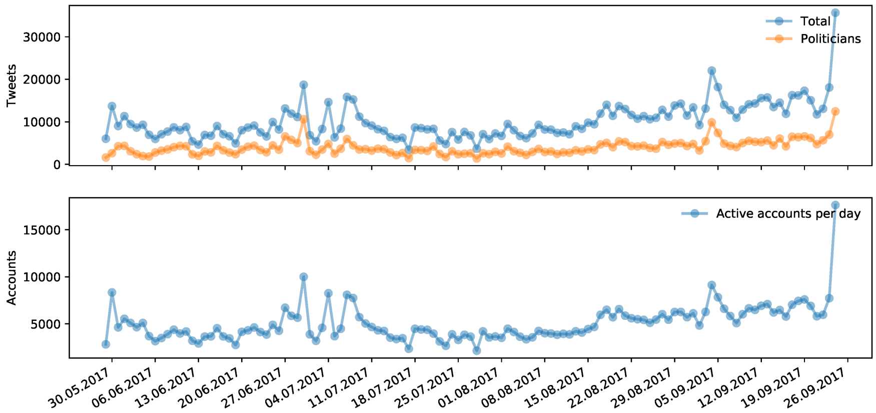


Figure 4. #BTW17 dataset, active accounts, taken from [9], the 26th Sep. 2017 was the election day

The #BTW17 dataset [8,9] comprises more than 1,200,000 tweets from 120,000 users recorded between June and September 2017. These recorded tweets and users are stored in precisely the JSON-based API (raw) format provided by the public Twitter streaming API. Figure 4 shows the observed tweeting activity throughout the recording. This dataset can be taken as a typical sample of what is "going on" on Twitter every day.

Therefore, the #BTW17 dataset was time-ordered injected into the Twista recording engine with different recording chunk sizes. A chunk size of 1 means effectively to store every event immediately and is therefore very similar to the behaviour of the Twitter streaming API itself. A chunk size of 10.000 means to eliminate all duplicates until 10.000 unique entities could be collected. The Twista recording engine recorded this stream as it would have recorded a live stream. In the aftermath, the resulting data sizes of the records resulting from different chunk sizes could be compared to reason about the redundancy reduction efficiency of varying chunk sizes.

4. Results

Since April 2019 Twista records a sample of the complete German Twitter stream [26]. This long-term evaluation demonstrates Twista's large-scale recording capabilities. Figure 4 shows the number of recorded tweets and active users per month using the presented redundancy reduction approach. If the reader compares Figure 5 with Figure 4, it becomes evident that the given recording solution is capable to archive much larger datasets than the selected #BTW17 reference dataset. The evaluation turned out, that is possible to archive the complete public sample of the German language Twitter stream. Even the full public sample of the English language Twitter stream has been recorded for several days without problems. This data has been used to compile Figure 7 and to deduce language-dependent Twitter usage characteristics.

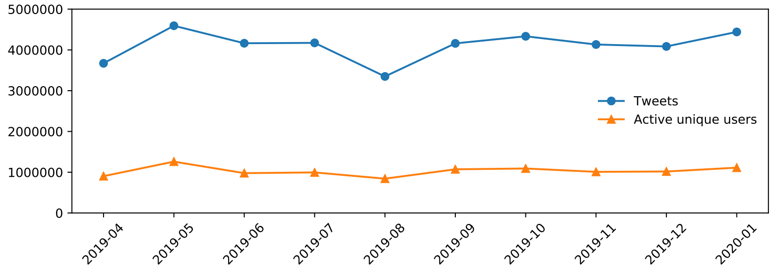
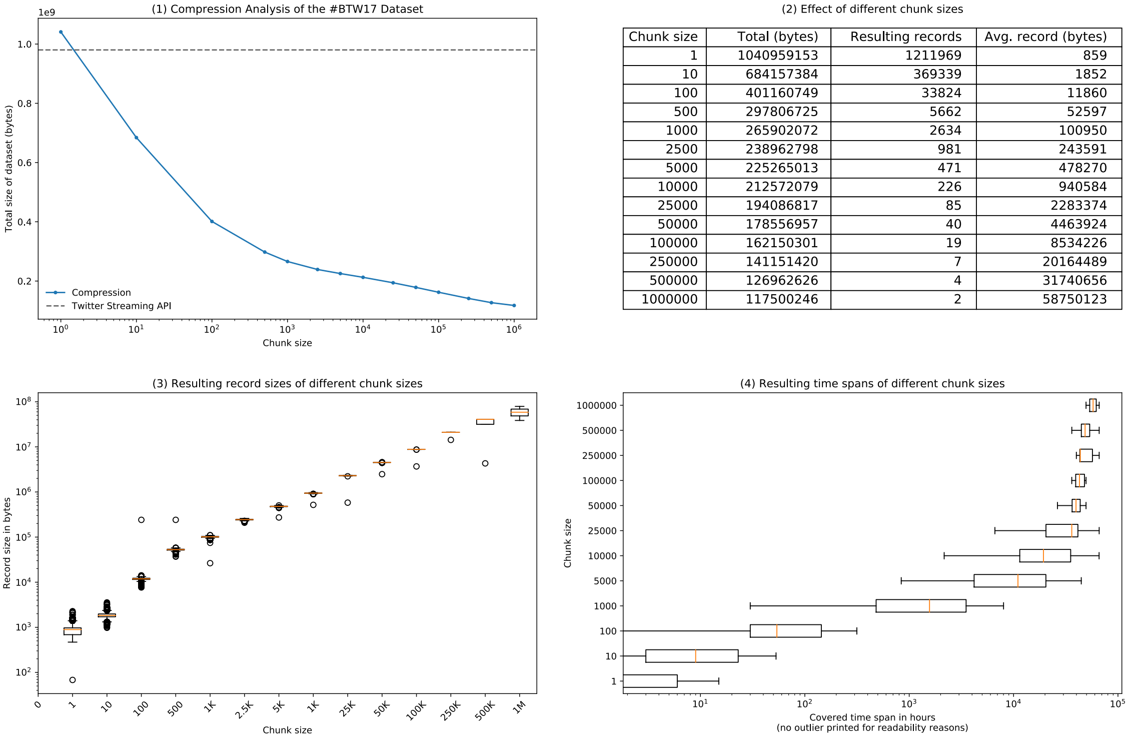


Figure 5. Long-term evaluation of Twista (recording the complete public sample of the German Twitter stream [26])

Figure 6 shows the evaluation results processing the #BTW17 dataset. The compression analysis presented in Figure 6(1+2) shows that increasing the chunk size decreases the overall size of the recorded dataset. The effect is for smaller chunk sizes much more significant than for larger chunks. So enlarging the chunk size results obviously in more massive records but the redundancy reduction effect is hardly measurable on the right end. Chunk sizes larger than 250.000 entities per file make merely sense and only increasing the record size (but hardly minimize the redundancy any more).



(2) Effect of different chunk sizes

Chunk size	Total (bytes)	Resulting records	Avg. record (bytes)
1	1040959153	1211969	859
10	684157384	369339	1852
100	401160749	33824	11860
500	297806725	5662	52597
1000	265902072	2634	100950
2500	238962798	981	243591
5000	225265013	471	478270
10000	212572079	226	940584
25000	194086817	85	2283374
50000	178556957	40	4463924
100000	162150301	19	8534226
250000	141151420	7	20164489
500000	126962626	4	31740656
1000000	117500246	2	58750123

Figure 6. Replaying of #BTW17 data, results of evaluation

Increasing the chunk size also increases the time-span that is covered by a record. According to Figure 5(4), records with 100 recorded events include about 100 hours (4 days) between the youngest and oldest event. This period can increase to more than 50.000 hours (six years) for a chunk size of 500.000 events. This astonishing long period has to do with the fact that in some rare cases, really "old" tweets are referenced. Especially the time-spans are deeply dependent on the recorded dataset and may show completely different characteristics.

All these effects shall be considered when selecting a chunk size for a large-scale recording. According to made experiences chunk sizes larger than 250.000 make little sense (even for massive streams like the complete English Twitter stream). For less frequent streams – like the German Twitter stream – chunk sizes of 100.000 per record seem to be a reasonable balance between redundancy reduction and record size.

In general, more massive Twitter streams need larger chunk-sizes, less frequent Twitter streams should prefer smaller chunk-sizes (otherwise the generation of a record takes unpractical long). However, archive sizes can be easily reduced to 20% compared with raw data, but some initial explorative experiments might be necessary to figure out an appropriate chunk size.

5. Critical Discussion

The Twitter streaming API returns only a sample of all tweets flowing through the Twitter social network. Data analysis must consider this and should take corresponding studies into consideration [2]. It is not assured by Twitter how big this sample size is. However, Twitter states a range of 1% and 10% for tweets. Studies that measured this sample size reported a sample size between 0.95% and 9.6% for tweets and between 10% and 45% for users [2,3]. Wang et al. concluded that *"the sample datasets truthfully reflect the daily and hourly activity patterns of the Twitter users. (...) Even with a very small sampling ratio (i.e., 0.95%), the sample datasets (...) preserve the relative importance (i.e., frequency of appearance) of the content terms"* [3].

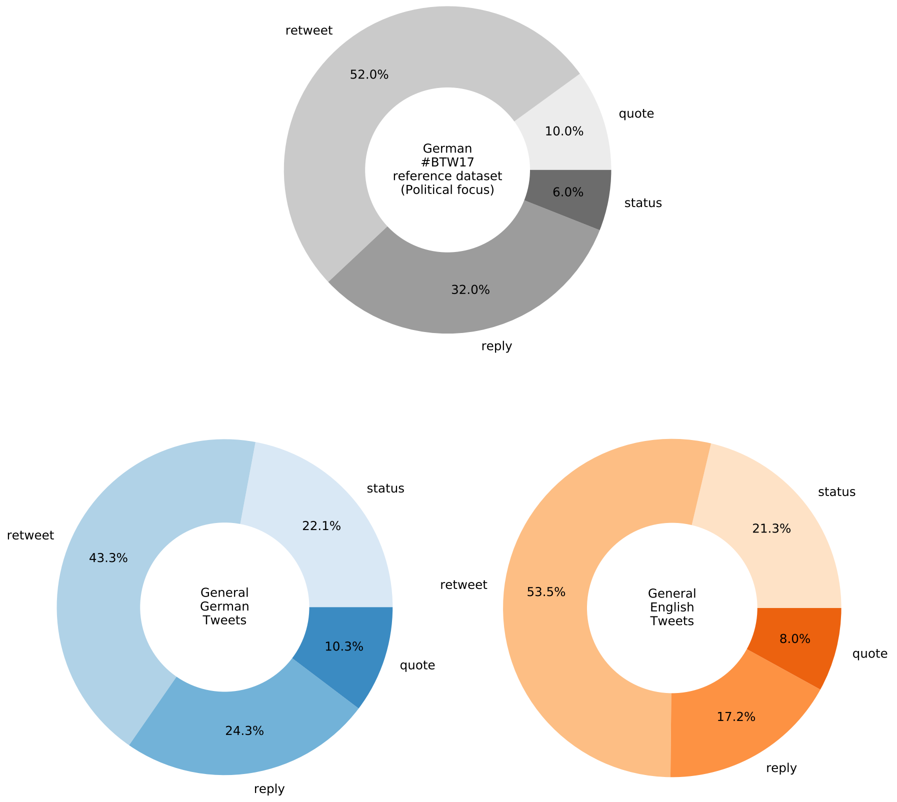


Figure 7. Example of observable differences across languages and datasets, the general German and English Tweets have been recorded from 9th to 12th Feb. 2020 for cross-validational purposes, the data for the #BTW17 dataset is taken from [8].

The evaluation of the compression rate in Section 4 has been done using the #BTW17 dataset [9]. This dataset has not been recorded for that purpose. It was taken merely because it is one of the rare Twitter raw datasets that exist publicly. However, according to its general characteristics (see Figure 4 and [9]), it should be big enough and typically shaped to derive a realistic picture of the compression capabilities of Twista. Nevertheless, if datasets are collected that differ significantly from the percentages of tweet types shown in Figure 7, different compression rates are expectable. For instance, an unrealistic dataset only with status posts would result in almost no compression rate at all. Figure 7 shows that slightly different percentages of tweet types are expectable across different languages. According to Figure 7, German tweeting users tend to reply more often than English tweeting users. On the other hand, English tweeting users seem to retweet more often than German tweeting users. Because retweets contain the retweeted tweet, and replies not (due to API historical backward compatibility reasons of the Twitter API) English tweets should be slightly better compressed than German tweets. However, this study does not investigate such aspects more deeply. Nevertheless, effects like that should be expected to some degree. Figure 7 shows furthermore another interesting aspect. Even within the same language (here German), the focus of the recording can influence the percentages of retweets, replies, status, and quotes. In political contexts, retweets and replies seem to occur more often than in other not explicitly specified contexts (at least in Germany). So, political recordings should be slightly better compressed than the general Twitter "basic noise".

Twitter interactions happen in the open space, and every Twitter user is aware of that by accepting the Twitter terms and conditions. However, recording with Twista enables to curate large-scale and long-term datasets of Twitter social network events that might deduce more in-depth insights of individuals that are hardly detectable by only analyzing the short-living real-time stream of social network interactions. Therefore, it is emphasized that the Twitter User Protection terms of use and general ethical considerations must be respected under all circumstances and compromises explicitly the following aspects:

- The data may not be used to conduct surveillance or gather intelligence with the primary purpose to isolate a group of individuals or any single individual for any discriminatory purpose.
- The data may not be used to target, segment or profile individuals due to their political affiliation or any other category of personal information.

6. Conclusions

Twitter provides a free sample of all events flowing through its streaming APIs and is, therefore, a valuable source for research. However, this data must be captured and archived. Plenty of studies made use of this data, but the recording, scraping, and archiving of this data seems to be more kind of an "art" than systematic application of standardized tools. Almost every social network-related study seems to develop its specific set of recording and data processing tool-suite. This situation leads to datasets in varying formats captured with hardly documented tool-sets and recording methodologies. In other words, the datasets are hardly comparable.

One problem is the amount of data that 145 million active users a day are producing. It was shown that this data is full of redundancy resulting quickly in Terrabyte of data and datasets that are hardly processable and shareable for research.

Gladly, it is possible to minimize the redundancy to made effective use of this valuable data source for research. Social network datasets can be reduced to about 20% of the original raw data without losing the valuable relationships between social network events.

This paper presented and evaluated a solution proposal. Twista can be used as a standardized means to record and generate Twitter datasets for research. Twista is available as open-source software [30]. In a long-term evaluation since April 2019, Twista recorded the complete public sample of the German Twitter stream [26]. Although the dataset for each month is about 1GB of data, this is astonishingly small for a social network stream covering all German tweets. That such large-scale datasets are effectively recordable is only possible because of the systematic exploitation of the inherent redundancy of such social network event streams.

Ongoing work will focus on better integration with dataset platforms like Zenodo and graph databases like Neo4j to simplify curation, sharing, and updating of large-scale social network datasets as well as their comparable and reproducible analysis.

Funding: This research received no external funding. Especially not from Twitter, or Governmental Agencies with technological needs or wishes for mass surveillance.

Conflicts of Interest: The author declares no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

API	Application Programming Interface
CSV	Comma Separated Values (data format)
JSON	JavaScript Object Notation (data format)
UML	Unified Modeling Language
TXT	Text file (data format)
XML	Extensible Markup Language (data format)
ZIP	compressed data format

References

1. Barberá, P.; Rivero, G. Understanding the Political Representativeness of Twitter Users. *Social Science Computer Review* **2015**, *33*, 712–729. doi:10.1177/0894439314558836.
2. Morstatter, F.; Pfeffer, J.; Liu, H.; Carley, K.M. Is the sample good enough? comparing data from twitter's streaming api with twitter's firehose. Seventh international AAAI conference on weblogs and social media, 2013.
3. Wang, Y.; Callan, J.; Zheng, B. Should We Use the Sample? Analyzing Datasets Sampled from Twitter's Stream API. *ACM Trans. Web* **2015**, *9*. doi:10.1145/2746366.
4. Wang, H.; Can, D.; Kazemzadeh, A.; Bar, F.; Narayanan, S. A system for real-time twitter sentiment analysis of 2012 us presidential election cycle. Proceedings of the ACL 2012 system demonstrations. Association for Computational Linguistics, 2012, pp. 115–120.
5. Crooks, A.; Croitoru, A.; Stefanidis, A.; Radzikowski, J. #Earthquake: Twitter as a Distributed Sensor System. *Transactions in GIS* **2013**, *17*, 124–147. doi:10.1111/j.1467-9671.2012.01359.x.
6. Oliveira, R.; Almeida, P.; de Abreu, J.F. From Live TV Events to Twitter Status Updates-a Study on Delays. Iberoamerican Conference on Applications and Usability of Interactive TV. Springer, 2016, pp. 117–128.
7. Gayo-Avello, D.; Metaxas, P.T.; Mustafaraj, E. Limits of electoral predictions using twitter. Fifth International AAAI Conference on Weblogs and Social Media, 2011.
8. Kratzke, N. The BTW17 Twitter Dataset - Recorded Tweets of the Federal Election Campaigns of 2017 for the 19th German Bundestag. *Data* **2017**, *2*. doi:10.3390/data2040034.
9. Kratzke, N. The #BTW17 Twitter Dataset - Recorded Tweets of the Federal Election Campaigns of 2017 for the 19th German Bundestag, 2017. doi:10.5281/zenodo.835735.
10. Cook, J.M. Twitter adoption and activity in US legislatures: A 50-state study. *American Behavioral Scientist* **2017**, *61*, 724–740.
11. Fraisier, O.; Cabanac, G.; Pitarch, Y.; Besancon, R.; Boughanem, M. # Élysée2017fr: The 2017 French Presidential Campaign on Twitter. Twelfth International AAAI Conference on Web and Social Media, 2018.
12. Stier, S.; Bleier, A.; Bonart, M.; Mörsheim, F.; Bohlouli, M.; Nizhegorodov, M.; Posch, L.; Maier, J.; Rothmund, T.; Staab, S. Systematically Monitoring Social Media: the case of the German federal election 2017. *arXiv preprint arXiv:1804.02888* **2018**.
13. Baviera, T.; Calvo, D.; Llorca-Abad, G. Mediatisation in Twitter: an exploratory analysis of the 2015 Spanish general election. *The Journal of International Communication* **2019**, *25*, 275–300.
14. Waisbord, S.; Amado, A. Populist communication by digital means: presidential Twitter in Latin America. *Information, Communication & Society* **2017**, *20*, 1330–1346.

15. Shao, C.; Hui, P.M.; Wang, L.; Jiang, X.; Flammini, A.; Menczer, F.; Ciampaglia, G.L. Anatomy of an online misinformation network. *PLOS ONE* **2018**, *13*, 1–23. doi:10.1371/journal.pone.0196087.
16. Moriano, P.; Finke, J.; Ahn, Y.Y. Community-based event detection in temporal networks. *Scientific reports* **2019**, *9*, 1–9.
17. Mazza, M.; Cresci, S.; Avvenuti, M.; Quattrociocchi, W.; Tesconi, M. RTbust: Exploiting Temporal Patterns for Botnet Detection on Twitter. Proceedings of the 10th ACM Conference on Web Science; Association for Computing Machinery: New York, NY, USA, 2019; WebSci '19, p. 183–192. doi:10.1145/3292522.3326015.
18. Basak, R.; Sural, S.; Ganguly, N.; Ghosh, S.K. Online Public Shaming on Twitter: Detection, Analysis, and Mitigation. *IEEE Transactions on Computational Social Systems* **2019**, *6*, 208–220. doi:10.1109/TCSS.2019.2895734.
19. Kratzke, N. Twitter Datasets, 2017 - 2020. https://zenodo.org/communities/twitter-datasets.
20. Nielsen, L.H.; Smith, T. Introducing ZENODO, 2013. doi:10.5281/zenodo.7111.
21. Shao, C.; Hui, P.M.; Wang, L.; Jiang, X.; Flammini, A.; Menczer, F.; Ciampaglia, G.L. Anatomy of an online misinformation network, 2018. doi:10.5281/zenodo.1154840.
22. Moriano, P.; Finke, J.; Ahn, Y.Y. Community-Based Event Detection in Temporal Networks, 2018. doi:10.5281/zenodo.1321085.
23. Basak, R. Online Public Shaming on Twitter- Dataset, 2019. doi:10.5281/zenodo.2587843.
24. Halasz, P. Twitter pre-trained word vectors, 2019. doi:10.5281/zenodo.3237458.
25. Mazza, M.; Cresci, S.; Avvenuti, M.; Quattrociocchi, W.; Tesconi, M. Italian Retweets Timeseries, 2019. doi:10.5281/zenodo.2653138.
26. Kratzke, N. Monthly Samples of German Tweets, 2019-2020. doi:10.5281/zenodo.2783954.
27. Roesslein, J. Tweepy, 2009 - 2020. https://tweepy.org.
28. Hui, P.M.; Yang, K.C.; Torres-Lugo, C.; Monroe, Z.; McCarty, M.; Serrette, B.; Pentchev, V.; Menczer, F. BotSlayer: real-time detection of bot amplification on Twitter. *Journal of Open Source Software* **2019**. doi:10.21105/joss.01706.
29. Zacharias, C.; Poldi, F. TWINT - Twitter Intelligence Tool, 2017 - 2020. https://github.com/twintproject/twint.
30. Kratzke, N. Twista - A Twitter streaming and analysis command line tool suite, 2017-2020. doi:10.5281/zenodo.845856.