Computational research in the post-API age

Deen Freelon

University of North Carolina at Chapel Hill

Forthcoming in *Political Communication*

Keywords: API, computational, Facebook, Twitter, social media

2018-08-20

On April 4, 2018, the post-API age reached a milestone. On that day, Facebook closed access to its Pages API, which had allowed researchers to extract all posts, comments, and associated metadata from public Facebook pages (Schroepfer, 2018). This decision followed the company's April 2015 closure of its public search API, which provided searchable access to all public posts within a rolling two-week window (Facebook, n.d.). The closure of the Pages API eliminated all terms of service (TOS)-compliant access to Facebook content. Let me underscore the magnitude of this shift: there is currently no way to independently extract content from Facebook without violating its TOS.

At the flip of a metaphorical switch, Facebook instantly invalidated all methods that depended on the Pages API. For example, I gave a Facebook data collection workshop in January 2018 at the University of Michigan whose lessons are now mostly unusable. A Python module I wrote to extract data from the Pages API is similarly obsolete. The specific implications for Facebook research are immense, but larger still are those for API-based research more generally. When companies can restrict or eliminate API access at any time, for any reason, and without any recourse, computational researchers and students need to seriously consider how to proceed. We find ourselves in a situation where heavy investment in teaching and learning platform-specific methods can be rendered useless overnight: this is what I mean by "the post-API age."

In this brief essay I want to provide two guiding lights for graduate education in computational methods going forward. APIs will continue to be important sources of digital communication data, but the closure of the Pages API demonstrates the dangers of relying on them exclusively. Researchers of social and other online media content should start by doing two things as they brace themselves for the uncertainty ahead. First, they should learn how to scrape the web; and second, they should understand the potential consequences of violating platforms' TOS by doing so.

Web scraping

Web scraping refers to the practice of automatically extracting content from web pages and other digital files. It predates API extraction by well over a decade, yet it has fallen somewhat out of favor as APIs have become easier to access. It is more flexible than API extraction because it can be used on most webpages, not just those that offer APIs. As a skill, therefore, it is more robust to the kind of obsolescence that befell my Facebook module. Instructors have many scraping tools they might choose to teach, from browser extensions like Web Scraper and Grepsr to standalone programs like Teleport Pro to cloud-based solutions like Import.io and Webhose.io. The main advantage of such tools is their easy-to-use graphical interfaces that allow users to get started immediately. However, most either scale poorly for research applications (browser extensions) or are prohibitively priced (most cloud solutions). Code-based scraping tools like BeautifulSoup and Selenium offer maximum flexibility at minimum cost (both are free) and thus would fit well into existing computational methods syllabi.

The major practical disadvantage to code-based scraping frameworks is that they are more difficult to learn than API tools. Whereas one can begin collecting API data with just a few lines of code (see e.g. Jürgens & Jungherr, 2016), scraping with code involves perusing and understanding the target page's unique DOM (Document Object Model) structure. The DOM elements containing the desired data must then be inserted into the chosen tool's functions and the output captured in a file. Not only is this

a more complex workflow than APIs typically require, it also differs for every web site. A researcher wishing to collect campaign messages from all members of the US Congress could easily do so using one of the many Twitter API interfaces and a list of member screen names. But scraping similar data from campaign web sites would entail developing independent scraping protocols for every major candidate site—a considerably more onerous undertaking, even for experienced computational researchers.

Terms of service

In addition to the technical challenges it poses, web scraping also raises legal and ethical dimensions that API users seldom face. Automated, large-scale content extraction exerts a substantial bandwidth toll on target sites, which is why many of the web's top platforms (including Facebook and Google) explicitly forbid it in their TOS. But this fact does not always prevent researchers from scraping data from such sites: while they typically attempt to automatically detect and block scraping, carefully-crafted software can circumvent such restrictions. With TOS-violating tools becoming ever more attractive in the post-API era, students and instructors need guidance concerning the risks of using them. The following recommendations are intended to balance researcher safety, user privacy, and corporate prerogatives, but are neither exhaustive nor universally applicable.

Use authorized methods whenever possible. Other things being equal, it is best to avoid violating a platform's TOS if possible. Before scraping a site that prohibits it, exhaust all possible means of procuring your data via officially-sanctioned means. This may entail thinking creatively about how to use these methods: for example, Twitter's API allows generous prospective collection of tweets but sharply limits retrospective collection. Therefore, researchers might set up a server-based data collection workflow that can quickly be initialized as events of theoretical interest occur. Of course, TOS-compliant methods to acquire the data of interest may not necessarily exist. In such cases, researchers should carefully consider the potential benefits and harms of using methods that violate a system's TOS.

Do not confuse TOS compliance with human subjects compliance or privacy protection. One of the most important purposes of research ethics is to protect research participants. The purpose of TOS is to protect the companies that create them. By employing TOS-compliant methods, you are respecting the business prerogatives of the company that created the platform you are studying, but you may or may not be respecting the dignity and privacy of the platform's users. For example, Twitter does not allow the public distribution of complete tweet datasets, ostensibly to prevent user content from persisting should it later be deleted. But this rule does not completely protect users from harm, since Twitter's API allows researchers to collect potentially sensitive information from vulnerable populations like minors. There are important reasons one might want to respect a company's express wishes about what should be done with the content it hosts, but that is not the same as protecting those responsible for creating the content.

Understand the risks of violating TOS. Researchers who decide to use unauthorized research methods should be aware of the risks. Twitter has been known to ask researchers not to share its data online (see Domanski, 2011)—the only data it allows to be shared are unique object IDs from which the (undeleted) original data can be reconstituted. To my knowledge, Twitter has not penalized anyone for violating its TOS by sharing forbidden data, but it may not have to. Given the relative power positions of university-based researchers and international corporations, it is little surprise that none of the former has emerged to directly oppose the company's wishes. Those who serially overuse APIs risk having their access credentials blacklisted ("Rate Limiting," n.d.), and while this does not permanently eliminate one's ability to use the API in question, it can result in delays and missing data. Google temporarily bans IP addresses that query its servers too quickly—this has even been known to befall overly enthusiastic humans.

Recent events have raised the specter of more dire consequences for violating TOS. In early 2013, computational activist Aaron Swartz committed suicide after being prosecuted by the Justice

Department for violating JSTOR's terms of service by downloading too many scholarly journal articles at once. He had been charged with violating the Computer Fraud and Abuse Act (CFAA) and was facing a maximum of 50 years in prison and $1 million in fines. While Swartz's case thankfully remains an extreme outlier, even the remote prospect of criminal prosecution for violating TOS creates a chilling effect strong enough to deter most researchers. A recent lawsuit brought by four computational researchers challenges the government's practice of prosecuting TOS violations but is still pending at the time of writing (Sandvig, 2017). Until it is resolved, researchers should bear in mind the potential (if unlikely) consequences of even small-scale TOS violations.

Conclusion

The post-API age is only beginning. As we prepare to train the next generations of computational researchers, we need to keep the tenuousness of our access to digital data firmly in mind. On the one hand, APIs are easy to use and TOS-compliant but may vanish without warning. On the other, web scraping is much more flexible but also more work-intensive and possibly illegal in some cases. I suggest that scholars interested in these issues prioritize them in future publications and conference submissions..

References

Domanski, R. J. (2011, May 5). Twitter Prohibits Research on Osama Bin Laden Tweets... Retrieved May 1, 2018, from http://thenerfherder.blogspot.com/2011/05/twitter-prohibits-research-on-osama-bin.html

Facebook. (n.d.). Public Feed API - Graph API - Documentation. Retrieved May 9, 2018, from https://developers.facebook.com/docs/public_feed/

Jürgens, P., & Jungherr, A. (2016). *A Tutorial for Using Twitter Data in the Social Sciences: Data Collection, Preparation, and Analysis* (SSRN Scholarly Paper No. ID 2710146). Rochester, NY: Social Science Research Network. Retrieved from https://papers.ssrn.com/abstract=2710146

Rate Limiting. (n.d.). Retrieved from https://developer.twitter.com/en/docs/basics/rate-limiting

Sandvig, C. (2017, October 19). Heading to the Courthouse for Sandvig v. Sessions. Retrieved May 1, 2018, from https://socialmediacollective.org/2017/10/19/heading-to-the-courthouse-for-sandvig-v-sessions/

Schroepfer, M. (2018, April 4). An Update on Our Plans to Restrict Data Access on Facebook | Facebook Newsroom. Retrieved May 9, 2018, from https://newsroom.fb.com/news/2018/04/restricting-data-access/