Proceedings of the 12th Web as Corpus Workshop, pages 1–4
Language Resources and Evaluation Conference (LREC 2020), Marseille, 11–16 May 2020
© European Language Resources Association (ELRA), licensed under CC-BY-NC

Current Challenges in Web Corpus Building

Miloš Jakubíček, Vojtěch Kovář, Pavel Rychlý, Vít Suchomel
Lexical Computing & Masaryk University

Abstract

In this paper we discuss some of the current challenges in web corpus building that we faced in the recent years when expanding the corpora in Sketch Engine. The purpose of the paper is to provide an overview and raise discussion on possible solutions, rather than bringing ready solutions to the readers. For every issue we try to assess its severity and briefly discuss possible mitigation options.

1. Introduction

Web corpus building has been the major way of obtaining large text collections for almost two decades now (see (Kilgarriff and Grefenstette, 2003) for a starting point and (Schäfer and Bildhauer, 2013) for a current overview) and there have been many web corpora built isolated (using methods such as WebBootCat (Baroni et al.,)) or as part of a bigger corpus family such as (Jakubíček et al., 2013), (Benko, 2014) or (Biemann et al., 2007).

Web corpora have been used as the primary source of linguistic evidence for many purposes. Besides linguistic research itself, the main areas of application included development and evaluation of natural language processing tools and methods, computer lexicography or practical analysis of large texts for varying tasks like trends or topics monitoring.

Building corpora from web has become popular for all the advantages it brings: small building costs, high speed of building and prospects on getting a very large dataset that would perform well in Zipfian distribution were reasons that are still very relevant, perhaps even more than before as NLP becomes more widespread and used in projects on a daily basis and many NLP methods (such as word embeddings) rely on large text corpora.

Sadly, most of the disadvantages of using web corpora have not been overcome in the 20 years: web corpora still provide only a very limited set of metadata, it is still difficult to clean the web content automatically and on the legal front there has not been any significant progress that would clarify the legal status of the datasets[1].

In this paper we are not going to discuss the advantages and disadvantages of web corpus building but take a very practical look at the biggest obstacles for web corpus building as of 2020. The starting point for all reasoning is that one aims at building a corpus from web which should be as big as possible and as clean as possible, where by clean we merely restrict ourselves to technical cleaning: yielding well-formed and well-encoded documents containing human-produced natural language texts, ideally (but not necessarily) split into paragraphs or sentences.

The issues that we mention are basically those that we have faced in the recent years when building corpora for the TenTen corpus family programme.(Jakubíček et al., 2013)

[1]In the European Union. In the US, the case law on related projects like Google Books (`https://en.wikipedia.org/wiki/Authors_Guild,_Inc._v._Google,_Inc.`) paved the way for more relaxed web corpus usage.

2. Current Issues

2.1. Machine Translation

2.1.1. The problem

Machine translation is ubiquitous on the web. Surprisingly, it is rather low-resourced language webs affected the most by machine translation, where the quality of machine translation is often very poor, but the market size simply does not make the case for human translation. Website owners are therefore confronted with a rather simple choice: either no content for that particular low-resourced language, or (poor, but) machine translated. Where reputation does not play a big role (and that means: hobbyists, fans, cheap sales websites, blogs platforms etc.), the choice is frequently to use machine translation, whatever its quality would be.

2.1.2. Mitigation strategies

Detecting machine translated content automatically is very difficult and there are no language-independent methods with reasonable precision-recall trade offs. Recall that this is in the first place a problem for low-resourced languages, which typically suffer from limited online content anyway. Thus applying any high-recall/low-precision strategies likely harms the size of the resulting dataset significantly and the most efficient way lies in using semi-automated methods: typically this involves hiring a native speaker for several days, checking the corpus wordlist and most represented web domains to discover "nests" of machine translated content and remove the whole domains. The general rule of thumb is: if a website offers many language versions, it is likely that most or all are machine translated. If there is an Esperanto version, it is always machine translated. In one of the most recent crawls of Estonian, which was carried out at the end of 2019 to create Estonian National Corpus 2019 (Kallas et al., 2015) in collaboration with the Institute for Estonian Language, we have generated a list of 600 most represented web domains which were manually inspected and 110 sites were removed from the corpus since their content was computer generated.

Another observation was made when cleaning a Lao Web corpus from 2019. 761 of 991 (77 %) domains with URI paths beginning with "/lo/" were identified as "bad language" by a Lao native speaker[2] based on samples of texts from particular domains. Since many of these bad language samples looked like machine translated, our hypothesis that

[2]Native speakers were asked to choose from three options: "good", "bad" or "I can't tell"

URI path can indicate machine translated content was confirmed. Together with a manual inspection of most represented domains in the corpus, approximately 9 % of tokens in the corpus were removed.

2.1.3. Severity

The severity of this issue is very high. The whole point about building corpora is to provide authentic evidence of language use and anything that hampers this idea represents a serious problem.

2.2. Spam

2.2.1. The problem

Spam represents a similar issue to the machine-generated content in terms that it also brings unnatural and thus unwanted content into the corpus. While it may not necessarily be automatically generated, it frequently is and spammers have been improving the text generation algorithms (including by means of applying NLP methods) during their long battle with search engines over the past years. There are, however, notable differences from the machine-translated content that have huge impact on how this should be dealt with. While machine translation is used permanently, intentionally (by website owners) and legally, spam typically occurs on someone's website as its temporary, illegal and random misuse. Such hacked websites are then used a honeypot to bring the user to some (less temporary, but also not very permanent) target site.

The illegality is also related to the topic of spamming: it tends to cover areas that are (in a particular country) prohibited or massively regulated, such as drugs, pharmacy, lottery, guns, loans and mortgages or prostitution. The topic heavily depends on the country and its regulations.

The temporal aspects of spam fighting may be crucial to fight it successfully. In our experience it was almost never possible to access a spam site several weeks after it has been crawled, because it was already cleaned and either shut down or previous content was restored. It is also likely the reason why search engines seem to fight spam rather well by analyzing its dynamic and temporary properties, but for web crawling by means of taking a static snapshot of a web, it is still a serious issue. During the past five years we have been regularly discovering spam sites where it took several minutes for a trained NLP engineers to conclude that this is a spam site. The spam site was mimicking a regular institutional website (such as of an U.S. university) including all its typical parts (courses, enrollment etc.), but starting with level 3 or 4 of nested links on the website, spam content was found which was completely unrelated to the institution. Notably, the institution was completely made up, so this was not a hacked institutional website, but a hacked domain with completely invented content.

2.2.2. Mitigation strategies

Automatic mitigation strategies may focus on the temporal aspects of spamming and involve:

- starting the crawl from a set of trustworthy seed domains obtained from web directories such as curlie.org, formerly dmoz.org, lists of newspapers (e.g. onlinenewspapers.com) which are less likely to get hacked
- measuring domain distance from seed domains and not deviating too deep from the seed domains
- using hostname heuristics (long hostnames consisting of multiple words are likely to be computer generated and containing spam)

Manual strategies are similar to the machine translation but thanks to the fact that spam is, unlike machine translated content, topical, one can use more analytic approaches than just looking up most frequent domains. Inspecting the usual suspects (like *viagra*, *loan*, *lottery*, ...) by means of collocations (in our case, word sketches) or other analytical tools can quickly reveal lot of spam content.

A complete solution to this problem would basically involve the same efforts that search engines put into this which is typically not feasible for a small company or NLP department. Out of all the aspects of spam, the temporality makes it most vulnerable: having most of the web indexed and permanently checking updates allows the crawler to temporarily suspend domains that suddenly completely or significantly change the content and this strategy could largely prevent getting spam into corpora without introducing any biases.

2.2.3. Severity

This is a very severe issue for the same reason like the ones given for machine translated texts.

2.3. Closed Content

2.3.1. The problem

Web crawling began as soon as Internet was sufficiently populated with texts. At that time, Internet consisted mostly of (plain) texts and as it became widespread in the developed world, everybody – institutions, companies, shops – went online, providing lots of natural language usage. Unfortunately, in many less developed countries where Internet became widespread later, going online meant creating a social network profile. As result, in these countries the Internet outside of social networks is simply much smaller and many companies and institutions have merely a Facebook page. Thus, while the Internet is now easily accessible, widespread and those countries are heavily populated, one only gets a fraction by crawling publicly accessible websites compared to similarly sized (in terms of native speakers) developed countries e.g. in Europe.

An example is e.g. Laos, a country with over 7 million citizens out of which over 25 % are online [3] where after extensive crawling for about half a year we were only able to obtain (after cleaning) a corpus of about 100 million words (whereas, in a country like Slovenia with 2 million citizens out of which almost 80 % are online, one can crawl a billion-word-sized corpus with no extra efforts).

We have also experienced more multimedia usage in these countries over textual content. But whether this is an unrelated issue or not would require more investigation.

[3] Data taken from `https://en.wikipedia.org/wiki/List_of_countries_by_number_of_Internet_users`.

2.3.2. Mitigation strategies

None. This paragraph might be a simple as that. Accessing social network content programmatically for the purposes of web crawling is typically not only illegal, but also technically very limited or impossible. Also, after more and more data privacy scandals around many social networks, their policies for data access and sharing have been tightened a lot and there are no prospects of this changing anytime soon. When people switch from open internet to closed platforms, it is over for linguistic web crawling.

2.3.3. Severity

This is a non-issue for "old" internet countries, big issue for "new" internet countries and generally a threat for the future if more and more online content is being shifted from open internet into closed (social media-like) platforms.

2.4. Dynamic Content

2.4.1. The problem

Modern websites rely more and more on dynamic content that is rendered in the client browser. While this brings better user experience and new functionalities, it also represents quite a technical challenge when crawling the texts from such websites. If yielding the texts requires rendering the content using a browser engine, it slows down the processing of a single website by several orders of magnitude.

2.4.2. Mitigation strategies

The only general solution really is to run a browser in headless mode and pass each found website to it, render its content as HTML and process it as usual. Some websites offer an HTML-only version to mobile browsers but it is not clear whether this could be applied generally (many other websites may still not be very mobile friendly).

2.4.3. Severity

The severity of this issue is so far rather low because websites still tend to provide textual fallback (e.g. for old mobile phones). As soon as they stop doing so, crawling will need to involve website rendering.

2.5. Paid Content

2.5.1. The problem

Early internet witnessed free news which, when the Internet population started to rise, were accompanied by ads. It is now clear that this was only a transition model from printed to online news and the revenues from online advertising (severely hindered by many users intentionally using tools for blocking adverts) are not sufficient to replace the fallen revenues on printed media subscriptions. Increasingly more media publishers therefore investigate new business models that incorporate online subscriptions (Fletcher and Nielsen, 2017) and a freemium model (a limited number of free articles per month, or limited set of articles, with other being paid) slowly becomes the new standard. Unfortunately the same news sources often represented valuable parts of the web corpus and if they become entirely missing, a whole genre of texts might become omitted.

2.5.2. Mitigation strategies

If at some point indeed most, or most quality, newspapers become completely unavailable without paying, web crawling such websites will either require paying (typically very modest) fee for a regular subscription or negotiating some access with the newspapers. The most problematic part is that this would require a per-website solution which significantly harms the current scalability of web crawling. Even if one manages to negotiate free access to the newspapers, it will still require developing customized solutions to incorporate data from that particular news.

2.5.3. Severity

Not very severe as long as reasonable amount of the newspaper text type remains freely accessible. But after that, this will represent an issue mainly for linguistic research focusing on this particular genre of texts.

3. Conclusion

In this paper we briefly discuss some issues of web crawling that we have stumbled upon most frequently in the recent years. The list is by no means complete and comprehensive and its whole purpose is to raise discussion at the workshop around the individual issues, possibly sharing further ideas on how to mitigate them.

Trying to predict the future of web crawling is tempting but of course hard. One may though imagine that the homogeneous Internet, as we know it now, slowly collapses into:

- content provided through some kind of web applications, possibly close or available only after payment
- the rest

The key question is how big the rest is going to be and whether it will be big enough and of sufficient quality to keep web crawling serving its current purpose. If not, it will require different approaches, which we may not even call crawling then.

4. Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731015. This work has been partly supported by the Ministry of Education of CR within the LINDAT-CLARIAH-CZ project LM2018101 and by the Grant Agency of CR within the project 18-23891S.

5. Bibliographical References

Baroni, M., Kilgarriff, A., Pomikálek, J., Rychlỳ, P., et al.). Webbootcat: instant domain-specific corpora to support human translators.

Benko, V. (2014). Aranea: Yet another family of (comparable) web corpora. In *International Conference on Text, Speech, and Dialogue*, pages 247–256. Springer.

Biemann, C., Heyer, G., Quasthoff, U., and Richter, M. (2007). The leipzig corpora collection-monolingual corpora of standard size. *Proceedings of Corpus Linguistic*, 2007.

Fletcher, R. and Nielsen, R. K. (2017). Paying for online news. *Digital Journalism*, 5(9):1173–1191.

Jakubíček, M., Kilgarriff, A., Kovář, V., Rychlỳ, P., and Suchomel, V. (2013). The tenten corpus family. *Corpus Linguistics 2013*, page 125.

Kallas, J., Kilgarriff, A., Koppel, K., Kudritski, E., Langemets, M., Michelfeit, J., Tuulik, M., and Viks, Ü. (2015). Automatic generation of the estonian collocations dictionary database. In *Electronic lexicography in the 21st century: linking lexical data in the digital age. Proceedings of the eLex 2015 conference*, pages 11–13.

Kilgarriff, A. and Grefenstette, G. (2003). Introduction to the special issue on the web as corpus. *Computational linguistics*, 29(3):333–347.

Schäfer, R. and Bildhauer, F. (2013). Web corpus construction. *Synthesis Lectures on Human Language Technologies*, 6(4):1–145.