Web Scraping for a Database of Court Decision Related Documents

Alec Schürmann

January 2022

Bachelor Thesis

at the

Research Center for Digital Sustainability
Faculty of Science
University of Bern

supervised by

PD. Dr. Matthias Stürmer

Joel Niklaus

for the award of the title of "Bachelor of Science in Computer Science"

Field of Study:	Computer Science
Matriculation no.:	16-115-701
Postal address:	Neuhausweg 49
	3097 Liebefeld
E-Mail:	alec.schuermann@students.unibe.ch

Abstract

Reports of Swiss court rulings are anonymous to protect the privacy of involved subjects. The Swiss national research project "Open Justice vs. Privacy" strives to automate re-identification of anonymous reports of court rulings using natural language processing. The achievement of this goal requires a database of court decision related documents. So far, despite the increasing amount and availability of data, no data sets of external documents related to Swiss federal court rulings have been collected and processed. This research project aims to provide a database of court decision related documents from five promising online data sources. In a second step, the sources were analyzed and the documents along with the metadata provided were scraped. After being structured in a JavaScript Object Notation (JSON) format, the results were analyzed in the context of the overarching research project and quantitatively evaluated by examining the text similarity of the resulting data and the court rulings.

Acknowledgements

I would like to thank my supervisors, PD Dr. Matthias Stürmer and Joel Niklaus from the Research Center for Digital Sustainability, for their support during the elaboration process and the supervision of my bachelor thesis. Furthermore, I am very thankful for the helpful inputs of my family and friends.

Contents

List of Figures

List of Tables

1 Introduction

Digitization, which is advancing rapidly in industrialized countries, is not only changing everyday life and the entertainment industry, but is also leading to the adaptation, optimization and acceleration of processes in the public sector, such as in the area of justice. The large amount of available, but often not yet linked, data has the potential to not only improve, but also ensure privacy in the investigative work of the judiciary. To protect the privacy of involved people in Swiss court decisions, the publicly available documents are anonymized. Previous research from Vokinger & Mühlematter (2019) has shown, that it is possible to re-identify companies involved in court decisions by linking the rulings with external data. The overarching research project on "Open Justice vs. Privacy", which is supported by the Swiss National Science Foundation, aims to build an automated system based on natural language processing for re-identifying people from court rulings, for which external data is needed. The goal of this project is to create a structured database from Swiss court decision related documents by scraping related online documents and extracting their data. The data sources were selected from a list of promising sources provided by my supervisor, focusing on distinct sources that allow for experimentation with different approaches in the scraping process. The thesis starts with a review of related work, followed by an overview of the tools used over the course of this project. Subsequently, the data and method for each data sources are described in detail. These chapters are followed by the presentation and analysis of the resulting data sets. In a first step, the data was analyzed from a theoretical perspective to find out which parts of the data are particularly promising for the re-identification process. In a second step, the algorithms were evaluated in terms of the richness of their metadata, their overall consistency of the provided data and their robustness. In a last step, the data was analyzed quantitatively by using an algorithm, calculating the average Jaccard coefficient to measure the text similarity of the scraped data and the reports of the court decisions. The thesis concludes with a discussion of the findings and possible future research questions.

2 Related Work

Several studies, as for example the work from Gunawan et al. (2019), have examined different web scraping methods and their efficiency by comparing the REGEX, HTML DOM and XPATH methods in their process time, memory usage and data consumption when retrieving data from target websites. Table 1 shows the results of the study, according to which HTML DOM is the front runner with regard to average time and data usage, while regular expression (REGEX) seems to be the most efficient in terms of average memory usage.

Table 1: Comparison of three web scraping techniques

Parameter	REGEX	HTML DOM	XPATH
Time (Avg)	399,75	298,55	435,15
Memory Usage (Avg)	564.782,5	4.817,132	574.546,4
Data Usage	50.295,05	8.803,3	17.769,85

Sources: Gunawan et al. (2019)

Another study on web scraping from Sirisuriya et al. (2015) shows the background and many different techniques of web scraping in general. Additionally, it compares them by evaluating techniques and software and giving a final review for each of them. Compared to the study of Gunawan et al. (2019), this study includes more different techniques, while the other goes more in depth of the selected three methods. A third study from Myers & McGuffee (2015) was helpful at the start of the project when deciding which tools will be used, as it explores the viability of the Scrapy library for undergraduate projects. Finally, they also explain why it was preferred for their own student project. Lawson (2015) published a book about web scraping with Python. The book not only goes into detail of how web scraping works on a technical level, but also presents interesting methods to handle challenges such as solving CAPTCHA or interacting with forms. Another book from Chapagain (2019) covers the basics of web scraping and describes the web scraping techniques with LXML, XPATH and CSS selectors. It also presents the scraping libraries BeautifulSoup and pyquery as well as the Scrapy library. A very interesting contrast to this project is the article from Krotov & Silva (2018) about the legality and ethics of web scraping. It is meant to help researchers to decrease the

likelihood of controversies in their work. Another article going in a similar direction is from Gold & Latonero (2017). The article from ten Bosch et al. (2018) presents research about combining big data web scraping with survey technology and how to overcome sources that are more volatile, unstructured and badly defined. As the natural language toolkit (nltk) is used for the analysis of the results, the paper from Bird & Loper (2004) describes the toolkit and its features as well as reporting on its current state of development. The paper from Niwattanakul et al. (2013) uses the same method as this project to calculate the Jaccard coefficient to compare similarities between sets of data. Furthermore, the test results show advantages and disadvantages of the measurement.

3 Tools

The scraping algorithms in this project are generally based on **Python**. In this context, **BeautifulSoup** and **Scrapy** were considered to be useful as scraping tools. While Scrapy is a complete web scraping framework and supports crawling in a box, BeautifulSoup is only a parsing library and has to be combined with other libraries such as **Requests** to support crawling. After having evaluated the different approaches, BeautifulSoup and Requests have been chosen for this project, mostly because of their simplicity and transparency of what exactly the written code does, while also being very beginner friendly. Furthermore, **Tika-Python** was used to parse PDF files. This was necessary because a lot of documents are only provided in a PDF format. Later on in the project, **Selenium** was required for its ability to open a Firefox web browser. The web browser allowed to execute a scrolling script to trigger "lazy-loading", which is a design pattern in web design to improve performance and defer the initialization of objects to the point at which they are needed[1]. In our case this means that we have to iteratively scroll to the bottom of the page to access the objects that are not loaded yet. To identify the language in the cases where it was not available from the metadata alone, **TextBlob** ended up being a very simple, but yet effective package to fulfill this task. Next, **Puppeteer** for **JavaScript** was used to access documents in a HTML format that required a JavaScript-compatible browser. To evaluate the results in a quantitative way, the natural language toolkit (**nltk**) package was used together with **spaCy** to create an algorithm that allowed to calculate the text similarity using the Jaccard coefficient.

[1] https://en.wikipedia.org/wiki/Lazy_loading

4 Data and Method

The data from the five selected data sources were not equally structured and available, thus different methods were developed and used to collect and process the data. The table below shows the number of entries and the size of all five resulting data sets as of January 6, 2022. The source with the most data was LexFind with 34 844 entries and a size of 2.15 GB. This data source was available in German, French, Italian as well as partly in Romansh and English.

Table 2: Overview of data sources

	BGER	**BAG**	**SUST**	**Twitter**	**LexFind**
Amount	1689	407	2699	2407	34844
Languages	de,fr,it	de	de,fr,it,en	de	de,fr,it,en,rm
Size	8.25 MB	19.76 MB	18.44 MB	0.85 MB	2.15 GB

Sources: Own table based on the scraping results

4.1 Media Messages of the Swiss Federal Court

4.1.1 Data

First of all, I consulted the media messages of the Swiss Federal Court (ger. Bundesgericht, "BGER"), which are published on the official website[2]. Although the content of the messages is only available in a PDF-Format and not HTML, which would provide more details, the website itself already contains some useful information. This includes the date and the title of the media message. The title consists of the case number, the date of the judgment as well as the title of the judgment, which already provide initial information about the content of the media release. Besides that, there is a URL linked to the PDF file with the message content. The media releases summarize the facts of the case, the considerations and the decision of the federal court. This allows to gain important insights whether the case is relevant for a specific research question.

[2]https://www.bger.ch/index/press/press-inherit-template/press-mitteilungen.htm?histo=true

4.1.2 Method

The first approach to scrape the PDF files with the media releases was to use the Python Requests package to get the HTML content. In a second step, all 'a' elements and their 'href' attributes had to be found using the Python package BeautifulSoup. In a third step, the PDF file was fetched again using Requests and parsed with the Python Tika package. Overall, this method to get the PDF content worked satisfactorily. Unfortunately, the metadata of the PDF mostly did not contain the correct title, nor the correct date. As it was not possible to match the data from the website with the PDF file when filtering by the 'a' element, a different approach was necessary. On the second attempt, instead of just filtering the URLs, finding all 'div' elements inside the 'article' container enabled to directly access all necessary metadata in the output string. Following, the date, title and PDF URL were extracted from that string, and the PDF was parsed analogically to the first attempt. Finally, the output was formatted and saved as a JSON file.

4.2 Federal Office of Public Health

4.2.1 Data

Next, I analyzed the website of the Federal Office of Public Health (ger. Bundesamt für Gesundheit, "BAG")[3]. It releases weekly official journals for medical and media professionals. The journals contain information about medications, therapies, health-related developments and recommendations in the areas of prevention and screening.

4.2.2 Method

Similar to the media messages of the Swiss Federal Court, the journals are only published as PDF files. The website itself contains less information about the journals, such that most of the data had to be extracted from the PDF content. This also proved difficult as the content is mostly made of illustrations and tables. Therefore,

[3]https://www.bag.admin.ch/bag/de/home/das-bag/publikationen/periodika/bag-bulletin.html

there existed no consistent format that allowed for automatic extraction of the data. Consequently, the PDF files were downloaded and parsed analogically to the media messages of the Swiss Federal Court, with just the PDF URL, title and date being scraped from the website by using BeautifulSoup and Regex.

4.3 Swiss Transportation Safety Investigation Board

4.3.1 Data

The third website investigated was the internet presence of the Swiss Transportation Safety Investigation Board (ger. Schweizerische Sicherheitsuntersuchungsstelle, "SUST")[4]. The website contains, among other things, reports of accidents and serious major incidents. The data we are interested in is split into reports on aviation events and reports on rail/navigation events.

4.3.2 Method

After investigating the two result pages, it occurred that they do not behave in the same way when filtering for all events. The reports on rail/navigation events load all at once such that they are easier to scrape, while the reports on aviation events only show a small number of events and use lazy-loading to improve the websites performance. As all the events had to be scraped, a JavaScript-compatible web driver had to be installed to bypass this issue. In this project, "geckodriver" for Firefox was used with Selenium to open a Firefox browser and to execute a script to keep scrolling to the bottom of the page until the end was reached and all data was loaded. Besides the PDF files with the content of the reports, there was a lot of information provided within the table. It contained two dates, one from the event itself and another from the publication. Furthermore, it reported the location and some details about the involved aircraft or type of accident in the case of rail/navigation events respectively. All this metadata was scraped using BeautifulSoup to find the table with the data and to extract its content using the regular expression operations library from Python. While this has been all the information collected

[4]https://www.sust.admin.ch/de/berichte/

in the first iteration of this data source, after revisiting and analyzing the content of the PDF reports, it was possible to extract more data from the notification type of reports, as they had a somewhat consistent structure of listing detailed information such as involved people, companies, vehicles/aircrafts, further flight information and damage. The procedure was once again accomplished by using regular expression operations to acquire the start- and end-indices of the keyword-matches and to extract the characters in between. This process was far more successful with the rail/navigation rather than the aviation events as its structure was more consistent.

4.4 Twitter

4.4.1 Data

Another approach was to use social media, since it plays a huge role in our society to exchange opinions and information. While it was expected to be difficult to find and filter for promising Tweets related to court decisions, it was an interesting experiment of how useful it can be. Other platforms, such as for example Instagram or TikTok, consist primarily of pictures/videos and their descriptions, which are expected to be rather rare in the context of court decisions.

4.4.2 Method

Collecting data from Twitter worked differently than in the case of the other data sources. Instead of scraping a website, Twitter offers a public application programming interface (API) to search for posts. For academic purposes, Twitter grants access to the full-archive search and Tweet counts as well as advanced search operators. The challenge was to assemble purposeful keywords to retrieve a collection of court decision related Tweets. In a first approach, a data set containing file numbers of court decisions, provided by my supervisor, was used as an attempt to find the intended Tweets. Although it could be expected from the outset that not many people post about a specific case with the file number, it was surprising that even after searching the archive for hundreds of file numbers, no matches were found. In a second approach, Tweets were filtered by keywords related to court decisions. While

this time a lot of matches had been received, the results were difficult to clean up as many keywords not only resulted in nonrelated Tweets, but furthermore included Tweets from other countries. Even though it seemed that it would be an easy task to filter the tweets by their country to only end up with Swiss Tweets, the academic research Twitter API offers no viable option to fulfill these requirements. On the one hand, most of the relevant Tweets are not tagged with a location and on the other hand, it cannot be ensured that the possible location information in the profile also corresponds to the place of residence. Another approach would be to evaluate the language of the Tweet and then to narrow down the Tweets to the languages in Switzerland. But as its virtually impossible to filter by all the spoken languages in Switzerland due to the variety of dialects and regions, the attempt to collect relevant data turned out to be out of scope for this project and would therefore be a field for future research.

4.5 LexFind

4.5.1 Data

LexFind provides access to all Swiss federal, intercantonal and cantonal laws as well as their historic versions from 2006 onwards[5]. In relation to the research project, these are relevant to track any legislative changes and link them to court decisions.

4.5.2 Method

Compared to the other four data sources, scraping all the different federal, intercantonal and cantonal laws was far more extensive and complex. First of all, the metadata attached to all URLs from the LexFind API seemed incomplete after the initial examination. Because the cantonal websites do not follow a universal format in their page structure, every entity including all the metadata had to be analyzed separately and as if they were completely different data sources. Starting with the federal law, there was an issue with the HTML response, as the website required a JavaScript-compatible browser to see its content. This was solved by writing a script

[5]https://www.lexfind.ch/fe/de/info

using Puppeteer to launch a Chrome browser and navigating to each base URL, followed by its history, changes and quotes pages. The HTML contents fetched by the script were stored as JSON in a separate folder to be later used for the scraping process with Python. After the script has been running as intended, the HTML content oftentimes still was partly missing. As the bug did not always affect the same files, it was clear that the issue was not directly in the code and a timeout of two seconds made the process far more consistent, even if this led to a reduction in speed. After having received the HTML content, every entity needed its own scraping algorithm in a first version. In a second version, as some of the cantons appeared to have certain similarities, a "bucket-system" has been integrated. It follows that a lot of code could be reduced by applying one algorithm for each canton in the same bucket, with only small adaptations and exceptions for certain cases. When taking a step back and further analyzing the original metadata received from the LexFind API, it became clear that it is not as incomplete as it seemed at first. This resulted in many steps being unnecessary, such that they could be removed and replaced in a third version, forming an easier and generally more consistent method utilizing more of the already provided metadata. Finally, the output file was generated a little differently than usual, as its total size surpassed 2 GB and was written by an iterative process of attaching data to the file, making use of the JSON-lines data format. Due to its size, DVC[6] was used to remotely store the data. To make the results consistent, this was later also applied to the other data sets. The laws were scraped in German, French, Italian, English and Romansh, although most of the cantons only had them published in their main language. The biggest exception were the federal laws, which were available in German, French and Italian. A few of them also had an English and a Romansh version, which were only accessible by manipulating the URL, as the API only returned it in German, French and Italian.

[6]https://dvc.org/

5 Results

In this section the resulting data sets will be presented and analyzed in the context of the overarching national research project about automatically re-identifying involved people in court rulings by using natural language processing. The following four criteria were used to evaluate whether the data sources are "useful" or not: **Relevance**, **Metadata**, **Consistency** and **Robustness**. Relevance is the main criterion determining how closely the content of the documents relates to court rulings and how relevant they are for the re-identification process. For every source except LexFind, as it only contains Swiss laws, this criterion is also backed up with a quantitative evaluation of the text similarity. This analysis was made by splitting each word of the documents into tokens (= tokenization) and transforming the words into their base form (= lemmatization). The resulting tokens were then compared to each other to calculate the Jaccard coefficient. This coefficient is calculated by the size of the intersection divided by the size of the union, which results in a floating point arithmetic that indicates the text similarity. This process was iterated with a sample size of 100 documents per language from each data set, each matching with 1 000 court decisions in the corresponding language provided by my supervisor. This sample size was chosen because it is enough to represent a very accurate and significant result, while bigger sample sizes would only affect the result by a small margin. In the end, the factors were added up to form an average Jaccard coefficient for each language in a data set. The second criterion metadata is about how much metadata is available and its meaningfulness. Third, the consistency focuses on the completeness and structure across the data provided from the source. At last, robustness is evaluated by how error-prone the scraping process is.

5.1 Media Messages of the Swiss Federal Court

Figure 1 shows an example from its resulting data set. Besides the full PDF-content, the most important metadata are the title, date and references. The title is mostly useful as a short description to identify the content, while the references guarantee a direct link to its related court decisions. They may contain duplicates, depending

Figure 1: Example from Media Messages of the Swiss Federal Court


```
"id": 0,
"title": "Urteile (5A_927/2020, 5A_656/2019, 5A_701/2020) Schutz vor ungerechtfertigter Betreibung",
"date": "28.09.2021",
"url": "https://www.bger.ch/files/live/sites/bger/files/pdf/de/5a_0927_2020_2021_09_28_T_d_15_48_12.pdf",
"references": ["5A_927/2020", "5A_656/2019", "5A_701/2020", "5A_927-2020", "5A_656-2019", "5A_701-2020"],
"content": "Lausanne, 28. September 2021  Medienmitteilung des Bundesgerichts  Urteile (5A_927/2020,
5A_656/2019, 5A_701/2020)  Schutz vor ungerechtfertigter Betreibung […]"
```


Sources: Own figure with data from Media Messages of the Swiss Federal Court

on how many times they are mentioned in the article. The date provides additional information that can be used to filter for media messages in a specific period of time. The Jaccard coefficient of this data set is **0.03974** for German, **0.02438** for French and **0.02261** for Italian. The German coefficient of 0.03974 is the highest across all data sources, thus showing the highest average text similarity. Overall, this source contains interesting data for the re-identification process as the content is closely related to court decisions and the metadata provides useful identifiers. The consistency is very high as the data is well structured on the website and is always complete. The media messages of the Swiss Federal Court also have a high robustness as there have been no issues arising during the scraping process. In conclusion, this data source has a lot of potential to help re-identifying people in court decisions.

5.2 Federal Office of Public Health

Figure 2: Example from Federal Office of Public Health


```
"id": 0,
"title": "BAG-Bulletin 50/21",
"date": "13.12.2021",
"url": "https://www.bag.admin.ch/dam/bag/de/dokumente/cc/Kampagnen/Bulletin/2021/bu-50-
21.pdf.download.pdf/BU_50_21_DE.pdf",
"content": "BAG-Bulletin 50/2021 (Deutsch)  SO SCHÜTZEN  WIR UNS.  www.bag-coronavirus.ch
Informationsmagazin für medizinische Fachpersonen und Medienschaffende  BAG-Bulletin 50/2021 […]"
```


Sources: Own figure with data from Federal Office of Public Health

Figure 2 shows an entry from the resulting data set of the Federal Office of Public Health. It is structured very similarly to the media messages of the Swiss Federal Court. As the journals are not directly linked to court decisions, there are no

reference numbers. Although there is also a title, it is not as useful as the title from the media messages, as it only contains "BAG-Bulletin" in addition to the week and year of publication. Most journals do not contain a lot of information directly related to court decision, but some reports might have interesting details that could indirectly be linked to them. With a Jaccard coefficient of **0.01982** for German, the Federal Office of Public Health is higher than expected in terms of text similarity. While still not one of the highest, as it only contains medical information, it ends up being a considerable data source for the national research project. The metadata available is very limited, as the only helpful information besides the content is the date. In terms of consistency, its few data provided has always been complete and structured. The website itself is quite error-prone as its often unreachable with a 504-Error code.

5.3 Swiss Transportation Safety Investigation Board

Figure 3: Example from Swiss Transportation Safety Investigation Board


```
"id": 0,
"event_date": "08.10.2021",
"publish_date": "19.10.2021",
"location": "Emmenbrücke, LU",
"type": "Arbeitsunfall",
"url": ["https://www.sust.admin.ch/inhalte/BS/2021100801_VB_Emmenbruecke_D.pdf"],
"content": ["…"]
"infos": [{"event": "Arbeitsunfall", "event_type": "Arbeitsunfall", "location_date": "Emmenbr\u00fccke (LU), 8. Oktober 2021, 06:40 Uhr",
"regnr": "2021100801", "transport_type": "Eisenbahn", "companies": "Eisenbahnverkehrsunter- nehmen Steeltec AG, Emmenbr\u00fccke
Infrastrukturbetreiberin Steeltec AG, Emmenbr\u00fccke (Anschlussgleis) Weitere Unternehmen Baskarad AG, W\u00fcrenlos", "people":
"Rangiermitarbeiter, Jahrgang 1965, Baskarad AG", "vehicles": "Lokomotive vom Typ DER 100 (ferngesteuert) 1 vierachsiger Flachwagen der
Bauart Rs 4 vierachsige offene G\u00fcterwagen der Bauart Eaos", "damage": "Personen Der Rangiermitarbeiter wird t\u00f6dlich verletzt
Verkehrsmittel Keine Infrastruktur Keine", "description": "W\u00e4hrend Rangierarbeiten geriet ein Rangiermitarbeiter zwischen die Puffer
eines zuvor abgekuppelten Wagens und der \u00fcbrigen Rangierkomposition. Bern, 12. Oktober 2021"}]
```


Sources: Own figure with data from Swiss Transportation Safety Investigation Board

The data set from the Swiss Transportation Safety Investigation Board consists of two parts. Figure 3 shows one example from the rail/navigation data set. Aviation and rail/navigation only differ in a few values. In place of "type" in rail/navigation, aviation has a "details" value with information about the aircraft directly from the website. Furthermore, the "infos" section contains "location_date", "aircraft", "pilots", "passengers", "flight", "damage", "description" and "remarks", which stands in contrast to the rail/navigation information presented in Figure 3. In case of rail/-

navigation, the most important values are the event date and exact time, location, involved companies and people, vehicles and damage. Date and time are useful to delimit the search to events in a specific period of time. The location is also an important indicator for re-identifying people involved in court rulings. Companies and people are exactly what the research project is looking for, therefore being a crucial part of the metadata. Vehicles can also be a valuable identifier to match vehicles, as long as they are mentioned in court decisions. At last, the damage is the main fact of an accident and therefore another useful value to match events in court rulings. The metadata from the aviation part follows the same reasoning, with aircraft instead of vehicles and pilots/passengers instead of people. Overall, the data source is very relevant for court decisions regarding rail, navigation or aviation accidents and incidents. The Jaccard coefficient for the aviation part is **0.01742** for the German, **0.01491** for the French and **0.01994** for the Italian documents. The English reports could not be included in this evaluation, as the court rulings are only available in German, French and Italian. On the other side, the Jaccard coefficient for the rail and navigation part is **0.02134** for German, **0.01963** for French and **0.02908** for Italian. Not surprisingly, the text similarity from rail and navigation is higher than from aviation, as there tend to be overall more incidents in rail and navigation than in aviation. Out of all the five data sources, this is the one with the largest amount of metadata. A lot of the details are also very relevant for the re-identification of people in court decisions. But the data is rather inconsistent, as the documents are usually only available in a specific language and the structure, even only for a specific type of report, may vary. In terms of robustness, while not being very error-prone, the website is very slow and needs a timeout of at least two seconds for each scrolling action in the script in order to work properly and to load all documents. Overall, it is a very useful data source when the court rulings are about rail, navigation or aviation accidents/incidents.

5.4 Twitter

Figure 4 shows an example of a court decision related Tweet in the database. The Twitter API offers interesting metadata, some of them being the date of the Tweet

Figure 4: Example from Twitter


```
"id": "1455875458277511171",
"date": "2021-11-03T12:32:12.000Z",
"language": "de",
"author_id": "749585891946033152",
"conversation_id": "1455875458277511171",
"content": "RT @datt_thomas: Polizist soll bei Lehrgang in Gedenkstätte Buchenwald Thor-Steinar-Shirt
getragen haben. Er zeigt 2 Zeugen wg falscher Ver\u2026"
```


Sources: Own figure with data from Twitter

and the author-id. While the date is very important when it comes to an event or a court decision, it is less valuable in the context of this data source because the date of a Tweet may not correspond to the date from events or official reports, therefore possibly even providing misleading information. This also holds true for information about the authors, as their location or other details might also not be interrelated. Even the content itself might contain false information, as it is not an official source and users are free to write whatever they like. As a result, there is a high variance in terms of relevance of the Tweets. The Jaccard coefficient for this data set is **0.00660** for German, forming the lowest across all data sources. Not only is it expected to have the smallest relation out of the five selected sources, but the numerous typing errors the users make in their Tweets as well as the different dialects spoken in Switzerland negatively impact the calculation of text similarity. In terms of metadata, the API has a lot to offer, but most of it is not providing consistent and helpful data for the re-identification of court decisions. Finally, the robustness of the API is very high, although the academic research product track is limited to 300 requests every 15 minutes. Overall, it is found to be too unreliable and not the best data source.

5.5 LexFind

Figure 5 shows one example from its resulting data set across all 28 LexFind data sources, including the federal, intercantonal and the 26 cantonal laws. While not directly enabling re-identification of involved people in court decisions, LexFind is relevant for the research project to link the mentioned laws in the court decisions with the content of their versions in force. It is not only the data source with the

Figure 5: Example from LexFind (from the canton of Aargau)


```
"canton":"ag",
"language":"de",
"uuid":"3dd10853-05f8-421a-af35-5d8dabb11518",
"title":"Gesetz über die Geoinformation im Kanton Aargau",
"short":"Kantonales Geoinformationsgesetz",
"abbreviation":" KGeoIG",
"sr_number":"740.100",
"is_active":true,
"version_active_since":"2021-09-01",
"family_active_since":"2011-05-24",
"version_inactive_since":null,
"version_found_at":"2021-09-01",
"pdf_url":"https:\/\/www.lexfind.ch\/tol\/1557\/de",
"html_url":"https:\/\/gesetzessammlungen.ag.ch\/data\/740.100\/de",
"pdf_content":"…",
"html_content":"…"
```


Sources: Own figure with data from LexFind

most entries and highest complexity, but it also offers a lot of metadata. The main values for the research project, besides the content, are the title and short title, abbreviation and serial number, as those will be the values used to link them to the court decisions. Furthermore, it is important to know whether or not this version is currently in force, which is shown by the boolean "is_active". As previously mentioned in the part about the method, the occasional issue when scraping the HTML content with Puppeteer impacts the consistency. Due to the content also being available in a PDF format, this does not lead to a loss of data. But the fact that the 28 data sources do not have a universal structure, and sometimes even not provide the content in the HTML format, makes some cantonal websites very inconsistent. Fortunately, this does not affect the resulting data set too much, as the metadata is also available from the LexFind API. The only exceptions, where the LexFind API does not provide metadata, are the English and Romansh version of the federal laws. The robustness is generally high, except for the aforementioned issue with Puppeteer requests. Overall, this will be a useful data source for the research project.

6 Discussion

6.1 Conclusion

In the framework of this research project, I created and applied algorithms to scrape court decision related data from five different data sources and analyzed their usability for the overarching national research project about "Open Justice vs. Privacy", which aims to automate re-identification of people involved in court rulings. When selecting the sources, the main goal was not to only to focus on the most promising sources, but also to explore differently structured sources, such that different methods had to be developed. After scraping the websites, extracting the data and structuring it in a JSON format, the resulting data was evaluated in two steps. Table 3 shows the results of the theoretical analysis.

Table 3: Theoretical evaluation of data sources

	BGER	**BAG**	**SUST**	**Twitter**	**LexFind**
Relevance	High	Medium	High	Low	High
Value of Metadata	Medium	Low	High	Medium	High
Consistency	High	High	Low	Low	Medium
Robustness	High	Medium	Medium	High	Medium

Sources: Own table based on evaluating resulting data sets.

As a result, three of them, namely the Media Messages of the Swiss Federal Court, the Swiss Transportation Safety Investigation Board and LexFind seem promising for the future work within the overarching research project. The Federal Office of Public Health turned out to be useful in niche scenarios and Twitter is rather ineffective. Table 4 presents an overview of the quantitative analysis.

Table 4: Quantitative evaluation of data sources

Ø Jaccard coefficient	**BGER**	**BAG**	**SUST Aviation**	**SUST Rail & Navigation**	**Twitter**
German	0.03974	0.01982	0.01742	0.02134	0.00660
French	0.02438	n/a	0.01491	0.01963	n/a
Italian	0.02261	n/a	0.01994	0.02908	n/a

Sources: Own table based on evaluating resulting data sets.

The analysis is based on the average Jaccard coefficients of each data set to calculate

the text similarity of the data with court rulings. I chose this as my evaluation method, as it reflects well the core research question on how useful the chosen data sources can be as a database for automating re-identification of involved people in court rulings. Another possible method to evaluate the results would be to only consider the documents with the highest Jaccard coefficient. While that would more accurately return which documents might be usable for the re-identification, it would not represent the overall value across all documents of the chosen data sources and therefore is less suited for the initially formulated research question of this project. Yet, it might be an interesting evaluation method to consider in the future work of the overarching national research project itself, when analyzing the data provided by this project and from other data sources. Furthermore, unfortunately some of the work, as for example the initial iterations of the LexFind algorithm, which were completely functional as well, turned out to be unnecessary, as there was finally found a more efficient and consistent method.

6.2 Future Research

The project focused on five different data sources with documents related to court decisions. It could be therefore expanded in a first step by adding more data sources. Moreover, the data collected could be used to manually try to re-identify people in selected court decisions to consolidate the results of the evaluation. The analysis concerning LexFind could be expanded by going more into detail with the first iteration of the implementation, where every page that had an HTML version available was scraped with a separate algorithm. Doing this might result in getting additional information that is not available via the LexFind API, especially in languages that are not supported by it. While Twitter will probably never be an ideal data source for such a project, there might be some usable cases when refining the search queries and finding a way for more geolocation data to further delimit the amount of unrelated and foreign Tweets. Mostly, I hope that this database will be useful and expandable in the future research of the national research project about "Open Justice vs. Privacy".

7 Bibliography

Bird, S., & Loper, E. (2004). Nltk: the natural language toolkit. Association for Computational Linguistics.

Chapagain, A. (2019). *Hands-On Web Scraping with Python: Perform advanced scraping operations using various Python libraries and tools such as Selenium, Regex, and others*. Packt Publishing Ltd.

Gold, Z., & Latonero, M. (2017). Robots welcome: Ethical and legal considerations for web crawling and scraping. *Wash. JL Tech. & Arts*, *13*, 275.

Gunawan, R., Rahmatulloh, A., Darmawan, I., & Firdaus, F. (2019). Comparison of web scraping techniques: regular expression, html dom and xpath. In *2018 International Conference on Industrial Enterprise and System Engineering (ICoIESE 2018)*, (pp. 283–287). Atlantis Press.

Krotov, V., & Silva, L. (2018). Legality and ethics of web scraping.

Lawson, R. (2015). *Web scraping with Python*. Packt Publishing Ltd.

Myers, D., & McGuffee, J. W. (2015). Choosing scrapy. *Journal of Computing Sciences in Colleges*, *31*(1), 83–89.

Niwattanakul, S., Singthongchai, J., Naenudorn, E., & Wanapu, S. (2013). Using of jaccard coefficient for keywords similarity. In *Proceedings of the international multiconference of engineers and computer scientists*, vol. 1, (pp. 380–384).

Sirisuriya, D. S., et al. (2015). A comparative study on web scraping. *Department of Computer Science, Faculty of Computing, General Sir John Kotelawala Defence University, Ratmalana, Sri Lanka*.

ten Bosch, O., Windmeijer, D., van Delden, A., & van den Heuvel, G. (2018). Web scraping meets survey design: combining forces. In *Big Data Meets Survey Science Conference, Barcelona, Spain*.

Vokinger, K. N., & Mühlematter, U. J. (2019). *Re-Identifikation von Gerichtsurteilen durch" Linkage" von Daten (banken): eine empirische Analyse anhand von Bundesgerichtsbeschwerden gegen (Preisfestsetzungs-) Verfügungen von Arzneimitteln.*

Erklärung

gemäss Art. 30 RSL Phil.-nat.18

Name/Vorname: Schürmann Alec

Matrikelnummer: 16-115-701

Studiengang: Informatik

Bachelor ☑ Master ☐ Dissertation ☐

Titel der Arbeit: Web Scraping for a Database of Court Decision Related Documents

LeiterIn der Arbeit: PD Dr. Matthias Stürmer

Ich erkläre hiermit, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen benutzt habe. Alle Stellen, die wörtlich oder sinngemäss aus Quellen entnommen wurden, habe ich als solche gekennzeichnet. Mir ist bekannt, dass andernfalls der Senat gemäss Artikel 36 Absatz 1 Buchstabe r des Gesetzes vom 5. September 1996 über die Universität zum Entzug des auf Grund dieser Arbeit verliehenen Titels berechtigt ist.
Für die Zwecke der Begutachtung und der Überprüfung der Einhaltung der Selbständigkeitserklärung bzw. der Reglemente betreffend Plagiate erteile ich der Universität Bern das Recht, die dazu erforderlichen Personendaten zu bearbeiten und Nutzungshandlungen vorzunehmen, insbesondere die schriftliche Arbeit zu vervielfältigen und dauerhaft in einer Datenbank zu speichern sowie diese zur Überprüfung von Arbeiten Dritter zu verwenden oder hierzu zur Verfügung zu stellen.

Ort/Datum

Bern, 10.1.2022

A. Schürmann

Unterschrift