Description of dataset "PolMedUse_polnewsDK" (Members of Danish Parliament in Danish news media)

CALDISS, Aalborg University

March 6, 2023

1 Introduction

This report describes how the dataset "PolMedUse_polnewsDK" was collected and handled.

The dataset contains all publicly available news articles from policy sections of Danish news media sites from September 2020 to December 2022.

The dataset contains news articles from the following Danish news outlets:

- DR Nyheder
- TV 2 Nyheder
- Berlingske
- Jyllands-Posten
- Ekstra Bladet
- Politiken

Articles were collected using web scraping with permission from the news outlets.

2 Collection of data

Data was collected using manually build scrapers in Python. All code is available at the following GitHub repository: https://github.com/CALDISS-AAU/project_pol-media-use.
Collection started on September 1st 2020 and was terminated December 12th 2022.
The scrapers were set up to monitor the policy sections of the news outlets every two hours, adding new articles to the dataset if new articles were encountered.
Based on manual inspection of news articles from the different news outlets, relevant source code (HTML) was identified to extract specific content of interest like the title, publish date

and text of the article. As several news outlets have articles that require a subscription to read (i.e. "paywalled"), texts for these articles were not collected. The dataset however contains an indicator as to whether the article is paywalled or not.

How the scrapers collected the data can be summarized in these overall steps:

- Go to policy section of news outlet
- Look for hyperlinks to news articles on policy section
- Filter hyperlinks for articles that have not already been collected
- For each new hyperlink, collect the article title and publish date
- If the article is not paywalled, collect the article text

As get request from client to website server can fail for various, arbitrary reasons, the scrapers always retried the collection three times both for accessing the policy section and for collecting a specific news article.

2.1 Download of raw HTML files post-collection

For validation purposes, the raw HTML files for all articles were downloaded after end of collection.

3 Contents of the dataset

The dataset contains 33.119 articles. Text was retrieved for 26.449 articles, corresponding to 79.86% of the articles collected. Data contains 6.596 paywalled articles for which text has not been retrieved and 74 articles for which text was not retrieved for other reasons (articles that no longer exists, live blog pages, articles with non-standard formatting or other).

Data contains 19.860 articles where one or several Members of the Danish Parliament are mentioned.

Data contains articles from September 7th 2020 to December 7th 2022.

3.1 Data filtering

The initial collected data contained 35.974 articles. The dataset was filtered to account for the scrapers collecting several irrelevant articles. Irrelevant articles were mainly collected due to the scrapers not being set up appropriately to only collect links from particular sections of the websites. Most of the news outlets' websites contain subsections or side panels on their policy sections with links to most read articles on the website as a whole or to entirely different sections of the website. The irrelevant articles were filtered out based on their URL path, meaning articles with a URL path indicating that it is from a irrelevant section of the website were excluded.

Filtering steps

- Removing duplicates based on article link: 35.873 articles
- Removing irrelevant articles based on URL paths: 33.200 articles
- Filtering to only include articles from start of collection (September 1st 2020): 33.119 articles.

3.2 Detecting mention of Members of Parliament

Members of Parliament were detected using a simple dictionary method. The dictionary contained all politicians who were at some point a member of the Danish Parliament during the collection of the data. The dictionary accounted for variations in whether middle name, last name or both were included but otherwise assumed that the name was correctly spelled at least once in the article.

Whether the article mentioned politicians who were Members of Parliament was determined with simple string matching using the dictionary against the article text in Python. After checking whether politicians were mentioned, it was determined whether they were Members of Parliament at the time of the article by checking whether the article was published while the politician was an active Member of Parliament.

The variables in the dataset pertaining to detecting mentions of Members of Parliament reflect both whether their name was mentioned and that the article was published while they were an active Member of Parliament.

3.3 Variables

The dataset contains the following variables:

Variable	**Description**
uuid	Unique UUID for the article.
article_accessed	Indicator for whether the article was accessed (1 if accessed, 0 otherwise).
newspaper_name	Name of the news outlet from which the article was collected from.
newspaper_frontpage_url	URL of the frontpage of the news outlet from which the article was collected from.
article_title	Title of article based on HTML title tag of article.
article_link	Hyperlink to article.
article_datetime_raw	Date and time of publishing of article based on HTML (NOTE: Datetime format varies).
article_date	Publish date of article in standardized format (format: YYYY-MM-DD).
encounter_datetime	Date and time of when article was collected (format: YYYY-MM-DD HH:MM).
article_paywall	Boolean for whether article was paywalled upon collection (True if paywalled, False otherwise).
article_text	Text of article based on HTML tags used for article text on the website.
mp_matches	Members of Danish Parliament mentioned in the article text (stored in Python list format, i.e. ['Name Nameson', 'Othername, Othernameson'])
mp_match	Indicator for whether article mentions Members of Danish Parliament (1 if Members of Danish Parliament are mentioned, 0 otherwise).
filename	Filename of locally stored HTML file of article.

4 Error sources in the dataset

As collection relies solely on web scraping methods, there are some sources of error in the dataset. The most relevant sources of error are described below in descending order of assumed relevance.

4.1 Missing information from articles

Web scrapers often have to rely on certain assumptions about how the webpage is structured. In order to pin-point specific contents of a webpage, web scrapers use source code like

HTML tags to specify specific parts of a page. While HTML follow certain conventions, the use of HTML tags and attributes may vary greatly across websites. This means that for specific content (like article title, date and text), one has to set up the scrapers to fit the websites being collected from. As manually inspecting every single article webpage would defeat the purpose of an automated data collection, the scraping of the specific contents instead relies on assumptions about what HTML tags and attributes the different news outlets use for specific content of interest (article title, date and text).

In cases where the article webpage does not conform to the assumptions (either due to formatting errors or being set up differently as part of a design choice), the scraper will miss the specific content of interest.

4.2 Irrelevant text included in article text

Similar to the source of error described in the previous section, the assumptions made for collecting the text of the article can in some cases lead to the inclusion of irrelevant text. This can occur for articles containing text boxes of links to other articles in the middle of the article text, depending on how the scraping for that particular news outlet is configured.

This can also lead to errors in the detection of mentions of Members of Parliament, if a name of a Member of Parliament is present in the text box or title of the article being linked to in the middle of the article.

4.3 Missing articles from the collection

The scrapers were set up so that the article information (title, link, date, text etc.) was collected from the article webpage itself. This means that the article was not collected if the scraper was not able to access the article webpage.

This source of error is assumed to be rare, as article collection was re-tried three times upon each collection, which occurred every two hours. In order for the article to have been completely missed, the scraper would have had to fail every time it encountered the article for as long as it was present on the policy section of the news outlet.

4.4 Inclusion of irrelevant articles

Most of the websites contain subsections or side panels on their policy sections with links to most read articles on the website as a whole or to entirely different sections. The scrapers were not set up properly to exclude these subsections upon collection but have been attempted to be filtered out afterwards. It is however still possible that some irrelevant articles were missed in the filtering process and are thus still part of the dataset.