

Solutions Journalism in Google News

A semester project

Christopher Voit

christopher.voit@campus.lmu.de

LMU Munich

Munich, Germany

1 INTRODUCTION

News feeds have become ubiquitous in our daily lives, as we use them to publish content, connect with social peers and inform ourselves about current world affairs. While our lives have certainly been enriched through being able to access all kinds of information, whenever we please, this development has also led to significant economic and societal issues. Especially news outlets are struggling to retain customers, as more and more content can be directly accessed on platforms, such as Facebook or Google News [2]. In an effort to regain readers, outlets are increasingly publishing polarizing articles knowing, that users are more likely to interact with them [11]. However, this highlights only one of the issues arising from the current way of publishing and consuming news. Szabo and Hopkinson, for example, show that watching TV news increases stress levels in individuals, which stay high even 15 minutes after having finished watching [21]. Hilbig, as well as Newhagen and Reeves, have found that, compared to neutral or positive information, individuals deem negative information more true and remember it better [6, 16]. Furthermore, an increased coverage of societal and environmental issues can give rise to compassion fatigue. Affected individuals disengage from the news and report feelings of helplessness towards the issue at hand [7].

2 BACKGROUND

One approach to deal with the above-mentioned issues is the proposal of a new journalistic format, called solutions journalism (SJ). According to the non-profit organization "Solutions Journalism Network", this form of journalism "investigates and explains, in a critical and clear-eyed way, how people try to solve widely shared problems"[15]. The focus hereby lies on accurately portraying the problem, while, at the same time, discussing the benefits and shortcomings of potential solutions. With this journalistic practice still in its development, the effects of SJ news articles on the readers have already been studied by the academic community. Overgaard, for example, found that readers of SJ-articles report higher levels of self-efficacy and trust towards the news organization [18]. McIntyre, as well as Baden et al., showed that, compared to the control groups, individuals confronted with SJ news sources had less negative feelings and higher intentions to take positive action [1, 10]. Complimenting SJ news postings with relevant advertisements even led to an increased interest to seek out more information, share the article and to a more positive attitude towards the advertisement

and the associated brand [8]. However, the presented research focuses solely on the effects of substituting conventional news articles with SJ ones. No academic work has yet looked into complimenting conventional with relevant SJ news and investigating the effects on optimism, as well as positive and negative affect of users.

3 CONCEPT

To fill this gap in the literature, I developed a browser extension, which augments a website's news feed with SJ-articles. The extension scans all news articles on the website and looks at their titles. It then extracts the keywords of the title and searches for matching SJ-articles in a designated database. The article with the best keyword match is then injected into the original website right underneath the original article. This process happens once for all visible articles of the given news feed during the initial load and is repeated each time users scroll or reload the page. Injected SJ-articles can be interacted with by clicking on the article's title, which opens the original source of the article in a new browser tab. Additionally, users can click a button to expand the short form of the article (figure 2) to also show the publishing outlet, as well as a short description of the article (figure 3).

One of the benefits of this approach is the high accessibility of the solution. Since all functionality is encapsulated inside a browser extension, users only have to install it once to be able to interact with it on any given news feed. Furthermore, it increases convenience for users as they do not need to switch websites or media types to find designated SJ-article news outlets, such as Perspective Daily or the New York Times "Fixes" Series [3, 22]. Additionally, showing relevant SJ-articles right next to a regular article might contextualize the seen information and prompt a deeper discussion with the presented content. This could not only benefit the users' knowledge retention and understanding, but also increase the interaction rate and the time spent on site, thus generating more income for website providers and news outlets. Lastly, this approach might limit users' so-called "doom-scrolling" behavior by intentionally breaking with the often times negative content of regular news feeds through showing elements, which are visually and content-wise different from the surrounding user interface [24].

3.1 Prototype Iterations

To stay within the scope of the project, I chose to limit the functionality of the solution to the browser version of the Google News feed. Based on this decision, I developed a first Figma prototype concept, which can be seen in figure 1. It consists of a mockup of an article element from the Google News feed on the left hand side and a preliminary SJ-article element on the right hand side. The SJ-article, in this version, only showed the title and a "story match",

which aimed to indicate how relevant the SJ-article was compared the original one. By using the expand-button on the right hand side of the SJ-article, the users were able to unfold the article to show, both, a short description, as well as a link to original source of the article.

Following an informal feedback session with 3 potential users, I adapted the Figma prototype to the version seen in figure 2 and figure 3. One of the changes in this version was the layout of the SJ-article, which was now below instead of next to the original article. This had the benefit of better fitting into the DOM-structure of the Google News feed while also enabling a mobile-friendly layout for smartphone users. Furthermore, I chose to outline the article element with red to increase the affordance of the element and to show the author, as well the story match, in the same element as the title. I also changed the story match from a percentage to a categorical scale of "low", "medium" and "high" as this was better understood by users. Lastly, I adapted the expand-button to increase visibility and chose to include the date of the SJ-article in the unfolded state of the element so users could better classify the actuality of the content.

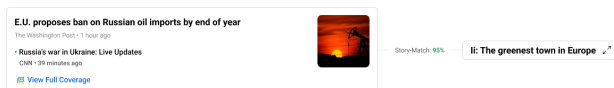


Figure 1: Screenshot of the first design version

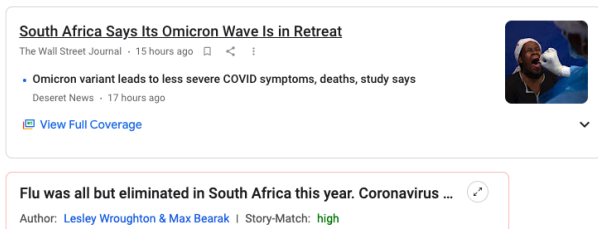


Figure 2: Screenshot of the final version in the folded state

4 TECHNICAL IMPLEMENTATION

For implementing the solution, I chose to rely on the Google Chrome extension infrastructure. This allowed me to quickly iterate as code changes are automatically reflected on the live page once the initial developer setup for the extension package is finished. Furthermore, I could rely on my existing web development experience as Chrome extensions build upon basic HTML, CSS and Javascript concepts. Lastly, Google Chrome is by far the most used browser by market share worldwide (circa 65%) with Safari in second place (circa 18%) and Firefox accounting for around 4% only making the solution accessible to a large audience [20].

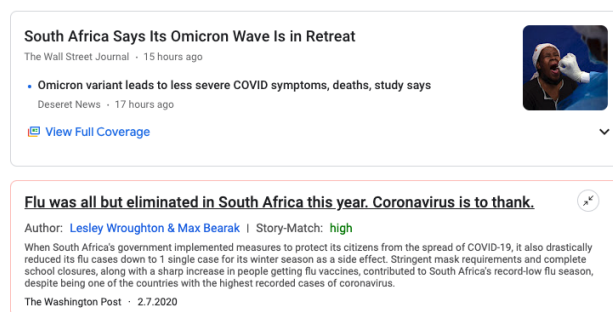


Figure 3: Screenshot of the final version in the unfolded state

4.1 Frontend

The frontend of the extension is mainly controlled by two (groups of) files:

- (1) The JS-file for loading the relevant SJ-article placeholders into the Google News feed (content script)
- (2) The JS-, HTML- and CSS-file filling in the content, defining the structure and styling the SJ-article placeholders

4.1.1 Content Script. The content script is loaded as a regular Javascript-file into the chosen website and can interact with all elements on the site. In the case of this project, the content script first checked the URL of the currently loaded page to assess whether to inject any SJ-articles. This was needed as not all Google News feeds follow the same design and HTML structure and, therefore, require different algorithmic rules for injecting the SJ-HTML at the correct place. It also adds a so-called MutationObserver, which listens to URL changes and adapts the scripts behavior accordingly. Once the script validated that the correct URL is loaded, it reads the page's DOM-tree and injects iFrames below each original article of the Google News feed. Each iFrame is given a unique ID, which is later used by the iFrames JS-file to only load the content reserved for it. To fill the content of each iFrame, the content script then communicates with the background script by making use of the messages API provided by the Chrome browser (see backend section). Once all needed content is assembled by the background script, the content script sends a message to all iFrames with a list of the loaded SJ-articles.

4.1.2 SJ-iFrame. Upon receiving the list of SJ-articles from the content script, each iFrame checks whether any of the included SJ-articles match their given ID. If this is the case, the article data, including the title, URL of the source, author, story match, news outlet, description and publishing date are loaded into the HTML template provided by the SJ-article HTML file. The HTML-file, in turn, loads the relevant styles from the associated CSS-file to adjust the design to the one of the latest Figma prototype. Lastly, the functionality for the expand-button is loaded and the iFrame is ready to be interacted with.

In addition to the elements loaded into the website itself, a small UI-element is provided to the users by clicking on the extension's icon in the Chrome browser interface. In this case, it gives additional

information about the concept of Solutions Journalism and in the future could be used to provide different settings, such as on which news feeds to load SJ-articles (see figure 4).

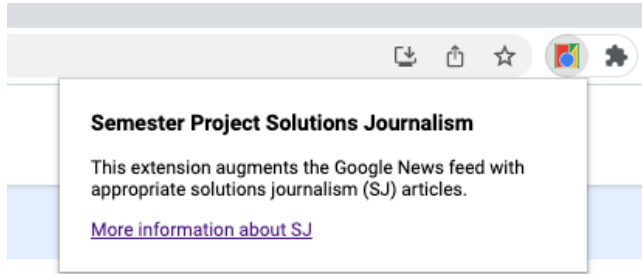


Figure 4: Screenshot of the settings UI in the Chrome browser

4.2 Backend

The backend of the extension is responsible for generating and loading all relevant information for the injected articles from different API-endpoints. It consists of the so-called background script and runs as a single instance across all loaded pages in the Chrome browser. In this project, it is the first script to load and inject the above mentioned content script into all active tabs. It then waits for the content script to request the data to fill the injected iFrames with SJ-articles. Following this request, the background script first extracts relevant keywords using the MonkeyLearn KeywordExtractor before loading a relevant article from the News-API based on the extracted keywords [13, 17]. It does so by accessing the provided API endpoints of the respective services and building a new JSON-object of the relevant information units included in the API responses. The properties making up the final SJ-article can be seen in table 1. Finally, it generates the so-called story match by looking at the title and description of the matched article and counting how often each keyword appears in either of them. The counted occurrences are weighted with the relevance score of the keyword and normalized to be between 0 and 1. The "low" story match is given for scores lower than 0.5, the "medium" story match for scores lower than 0.7 and the "high" story match for all higher scores.

Table 1: SJ-article JSON object.

Attribute	Type
Keywords	String
Keyword relevancy	Number
Title	String
Description	String
Author	String
URL	String
News Outlet	String
Publishing Date	Date
Storymatch	String

An example of this interaction between the content and the background script exchanging information using the messages API provided by the Chrome browser can be seen in figure 5 and figure 6. The content script extracts the title of the original article from the Google News feed and includes it in a posted message. The background script then listens to this message, reads the title and connects to the MonkeyLearn API. It receives a list of keywords including their respective relevancy scores and filters out all which have a lower score than 0.6. Finally, it sends this list of filtered keywords back to the content script which listens to incoming messages.

```

async function extractKeywords (element) {
  return new Promise( (executor: (resolve) => {
    const title = element.find('article').eq(0).children( elem: 'h3').children( elem: 'a').html()
    const port = chrome.runtime.connect( connectInfo: { name: 'extractKeywords' } )
    port.postMessage({
      data: {
        title: title
      }
    })
    port.onMessage.addListener( callback: (keywords) => {
      resolve(keywords)
    })
  })
)
}

```

Figure 5: Screenshot of the code requesting the keywords from the background script

```

async function extractKeywords (port, title) {
  const URL = 'https://api.monkeylearn.com/v3/extractors/ex_YCy9nrrn/extract/'
  const body = { data: [title] }
  return fetch(URL, {
    method: 'POST',
    headers: {
      'Authorization': 'Token ' + '433784597bcff13325dgd2fbc725f741e428aa1',
      'Content-Type': 'application/json'
    },
    body: JSON.stringify(body),
  }).then((response: Response) => response.json())
  .then(json => {
    let relevantKeywords = []
    const extractions = json[0]['extractions']
    extractions.forEach((extraction) => {
      let relevance = extraction['relevance']
      if (relevance >= 0.6) {
        relevantKeywords.push({ keyword: extraction['parsed_value'], relevance: extraction['relevance'] })
      }
    })
    return relevantKeywords
  })
}

```

Figure 6: Screenshot of the code loading the keywords for the content script

4.3 Miscellaneous

In addition to the above described files, the Chrome extension also contains an assets folder including all used images and icons, a minified version of JQuery for coding convenience, as well as the manifest file. This file holds all configuration for the extension, such as when to load which assets, permissions needed by the extension and on which pages to load the extension. Lastly, the extension also includes a "Test Mode", which was used for building a reproducible version of an augmented Google News feed for the user test. It detects if an archived version (using the WayBackMachine [9]) of the news feed was used and loads a predefined set of articles from a static JSON file into the website. This ensured that all participants

of the user test interacted with the same version of the Google News feed.

5 USER TEST

The user study followed a pre- and post-test design with a treatment and control group. In both groups participants were first asked to provide information about their demographics and news consumption behavior. This was followed by the State Optimism questionnaire by Millstein et al. to assess the participants' optimism as a changing state rather than a trait [12]. Additionally, users filled out the PANAS questionnaire to understand how positive and negative affect, such attentiveness or nervousness, changed through the treatment [23]. Once participants were done filling out the surveys, they were asked to interact with the Google News feed for 5 minutes in an exploratory manner. While both groups interacted with the same, archived version of the news feed, the treatment group was presented with a version modified by the SJ-extension [5]. During this interaction, the users were asked to verbalize their thoughts and their screens, as well as their voice, was recorded. Following the interaction, participants were again presented with the State Optimism and the PANAS questionnaire. Lastly, the treatment group was asked to fill out the scales of "Trustworthiness" and "Quality of Content" of the UEQ+ questionnaire [19]. These scales were used to understand how users perceive the actuality, quality and reliability of the presented information. A visualized version of the study design can be seen in table 2.

Table 2: Visualized study design.

Steps	Treatment Group	Control Group
Demographics	yes	yes
Pre-Test State Optimism	yes	yes
Pre-Test PANAS	yes	yes
Treatment	with SJ-extension	only Google News
Post-Test State Optimism	yes	yes
Post-Test PANAS	yes	yes
UEQ+	yes	no

5.1 Sample

The sample consisted of 5 female and 7 male participants with a mean age of 31.75 and a standard deviation of 14.06. Participants were split into a treatment and a control group of six members each. The treatment group consisted of 3 female and 3 male participants (Age: $M = 26.33$, $SD = 2.34$), while the control group included 2 female and 4 male participants (Age: $M = 37.16$, $SD = 18.96$). Information about the level of education (LoE) and news consumption (NC) can be found in table 3. The most used news sources as reported by the participants were "Tagesschau" (5 times), "Zeit" (4 times) and Instagram (3 times). The complete list can be found in the appendix in table 7.

Table 3: Demographics between groups.

Measure	SJ-extension Group	No extension Group
LoE - High school	0%	33.33%
LoE - Bachelor	50%	50%
LoE - Master	50%	0%
LoE - PHD	0%	16.66%
NC - Multiple times per day	33.33%	33.33%
NC - Once a day	66.66%	50%
NC - Once every few days	0%	0%
NC - Once per week	0%	0%
NC - Rarely	0%	16.66%

5.2 Quantitative Results

Because of the small sample and, therefore, group size it was important to choose an appropriate statistical method to analyze the scores. For both, the Optimism, as well as the PANAS scores, tests for variance homogeneity and normal distribution were conducted.

5.2.1 Optimism. The Optimism scores did not follow a normal distribution, as seen in the third row of table 4, and, therefore, ruled out using a mixed ANOVA for analysis. Instead, within and between subject scores were looked at independently using non-parametric tests. For within differences a Friedman-Test was used, which showed no significant results ($\chi^2(1) = .22$, $p = .64$). The differences between groups also showed no significant results, both for the pre-test ($U(N_{\text{SJ-extension}} = 6, N_{\text{No-extension}} = 6) = 22.5$, $p = .51$) and post-test Optimism scores ($U(N_{\text{SJ-extension}} = 6, N_{\text{No-extension}} = 6) = 17$, $p = .94$). A visualization of all test scores can be seen in the graph in figure 7. The descriptive statistics, as seen in table 4, however, show a possible negative impact on participant's Optimism levels, which is bigger in the control than in the treatment group.

Table 4: Optimism - Descriptives & Shapiro Wilk

	Pre-Test		Post-Test	
	No extension	SJ-extension	No extension	SJ-extension
Mean	4.214	4.167	3.905	4.071
SD	0.637	0.545	0.806	0.716
p-value	0.032	0.061	0.383	0.033

5.2.2 Positive PANAS. According to Levene's test, variance homogeneity for the positive PANAS scores pre- ($F(1, 10) = 3.97$, $p = .07$) and post-treatment ($F(1, 10) = 4.33$, $p = .06$) could be assumed. Additionally, all p-values of the Shapiro-Wilk-Test were not significant (see table 5) and, therefore, indicated the use of a mixed ANOVA for analyzing main and interaction effects. The mixed ANOVA showed no significant main effect within ($F(1, 10) = 3.55$, $p = .08$, $\eta_p^2 = .26$)

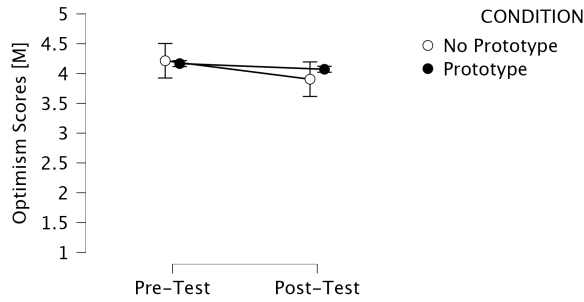


Figure 7: Mean Optimism scores

and between subjects ($F(1, 10) = 3.55, p = .08, \eta_p^2 = .26$). Furthermore, no significant interaction effect could be found ($F(1, 10) = .39, p = .54, \eta_p^2 = .03$). A visualization of the effects can be seen in figure 8.

Table 5: Shapiro Wilk - Positive PANAS

	Pre-Test		Post-Test	
	No exten- sion	SJ- extension	No exten- sion	SJ- extension
p-value	0.612	0.534	0.621	0.902

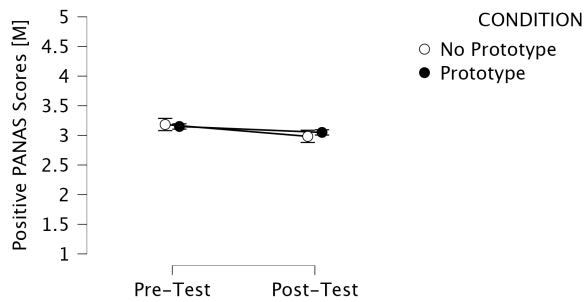


Figure 8: Mean positive PANAS scores

5.2.3 Negative PANAS. For the negative PANAS scores, no normally distributed data could be assumed in all groups and conditions (see table 6). As with the Optimism scores, non-parametric tests were used instead. A Friedman-Test showed no significant results for within group differences ($\chi^2(1) = .04, p = .83$). Additionally, between groups no significant differences were found pre- ($U(N_{SJ-extension} = 6, N_{No-extension} = 6) = 14, p = .57$) and post-treatment ($U(N_{SJ-extension} = 6, N_{No-extension} = 6) = 15.5, p = .74$). A visualized version of the test statistics can be found in figure 9. The descriptive statistics, as seen in 6, give a slight indication towards a trend of reducing negative affect scores in the treatment group while increasing them in the control group.

5.2.4 UEQ+. Finally, participants of the treatment group reported a mean of 0.21 for the subscale of "Trustworthiness" ($SD = 1.47$) and a mean of 0.63 for the subscale of "Quality of Content" ($SD = 1.41$) of the UEQ+. As the UEQ+ scale ranges from -3 (very negative

Table 6: Negative PANAS - Descriptives & Shapiro Wilk

	Pre-Test		Post-Test	
	No exten- sion	SJ- extension	No exten- sion	SJ- extension
Mean	1.517	1.617	1.617	1.533
SD	0.668	0.467	0.749	0.250
p-value	0.036	0.817	0.133	0.443

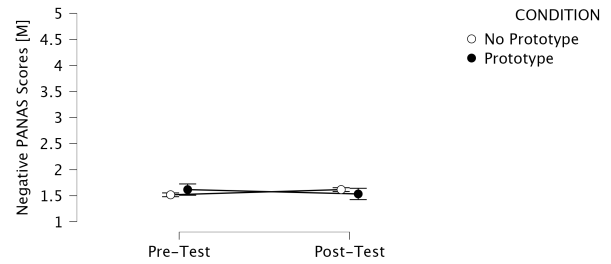


Figure 9: Mean negative PANAS scores

experience with the product) to +3 (very positive experience with the product), the reported values center around a neutral evaluation of the presented prototype. Subscales can also be used to calculate a KPI describing the overall user experience with the product, which resulted in a mean of 0.43 and a standard deviation of 0.72 for the SJ-extension.

6 CONCLUSION

This semester project focused on the ideation, implementation and evaluation of a novel approach to augment news feeds with SJ-articles. Following the indications of the positive influence of SJ on readers optimism and affect levels, I developed a Chrome browser extension that injects SJ-articles next to relevant Google News articles [1, 10, 18]. This approach aimed to provide higher accessibility of SJ content, deepen understanding of the presented information and increase time spent on the news website. It works by scanning the titles of the website's news feed articles, extracting keywords and finding a matching SJ-article in a designated database. The matched article is then injected at the adequate position in the website's DOM tree and can be clicked on by the user to show a short description or redirect to the original source of the SJ-article. A user test of 12 participants, split into a treatment and control group, showed no significant results for the measures of State Optimism, as well as positive and negative PANAS scores [12, 23]. The evaluation of the UEQ+ subscales of "Trustworthiness" and "Quality of Content" in the treatment group produced mean values corresponding to a neutral user experience [19].

6.1 Limitations & Future Work

One of the limitations of this work stems from the concept of solutions journalism itself. Since none of the users, either from the informal or the official user test, heard of SJ before, interacting with an extension that makes use of that concept created uncertainty around the presented content. Some users expected the injected

articles to be a highlighted article of the original website, others thought of it as a fact checking feature, which lead to misunderstandings about the purpose of the extension. However, as users usually actively seek out a browse extension that they install, most of such complications would be negligible in real-world scenarios. Furthermore, users were unsure about the content of story match property of the injected articles, which could be addressed by either showing a short tutorial upon first installing the extension of providing an information element when hovering over the story match property. Another limitation is posed by the extension currently only working for selected feeds on the desktop version of Google News. To significantly increase the accessibility of SJ content across the users' browsing journey, the extension would also need to work for major news feeds, such as Facebook, Instagram or Apple News, in the future. Furthermore, as users consume most news content using mobile devices, focusing on the mobile versions of such news feeds would make sense [4]. In addition, the current solution fetches SJ-articles not from a designated database, such as the one of the SJN, but from a general news API providing regular, non-SJ articles [14]. This is due to the SJN, unfortunately, not responding to our access request to their database, but could be fixed by substituting the current API endpoint with one that provides SJ-articles instead. Finally, the user test of the presented solution is limited by its small sample size of 12 participants. While all of the statistical tests showed no significant results, the descriptive statistics point towards a possible trend of decreased Optimism and positive PANAS scores in the control group, as well as decreased negative PANAS scores in the treatment group. Assuming a statistical power of 0.9 and an alpha level of 0.05, the required sample sizes needed to register significant effects can be found in table 8 in the appendix and show that at least 32 participants would be needed to produce a significant within effect for positive PANAS scores.

7 APPENDIX

Table 7: List of news sources.

News Source	Count
Tagesschau	5
Zeit	4
General Apps	3
Instagram	3
SZ	3
TV	3
FAZ	2
Financial Times	2
LinkedIn	2
Podcasts	2
Radio	2
Other	12

REFERENCES

- [1] Denise Baden, Karen McIntyre, and Fabian Homberg. 2019. The impact of constructive news on affective and behavioural responses. *Journalism Studies* 20, 13 (2019), 1940–1959.

Table 8: Required sample sizes

	Optimism	Pos. PANAS	Neg. PANAS
Within Effect	114	32	10500+
Between Effect	3494	10500+	10500+
Interaction Effect	396	268	68

- [2] Josh Constine. 2018. *How facebook stole the news business*. <https://techcrunch.com/2018/02/03/facebook-siren-call/>
- [3] Perspective Daily. [n.d.]. *Perspective Daily - Constructive Journalism*. <https://perspective-daily.de/>
- [4] World Economic Forum. [n.d.]. *WEF - News consumption per device*. <https://www.weforum.org/agenda/2021/05/rise-of-media-on-mobile-phone-chart/>
- [5] Google. [n.d.]. *Google News Feed - 01. January 2021*. <https://web.archive.org/web/20220101121742/https://news.google.com/topics/CAAqJggKliBDQkFTRWdvSUwyMHZNRGx1Y1Y4U0FtVnVhZD0pWVXlnQVAB?hl=en-US&gl=US&ceid=US%3Aen>
- [6] Benjamin F Hilbig. 2009. Sad, thus true: Negativity bias in judgments of truth. *Journal of Experimental Social Psychology* 45, 4 (2009), 983–986.
- [7] Katherine N Kinnick, Dean M Krugman, and Glen T Cameron. 1996. Compassion fatigue: Communication and burnout toward social problems. *Journalism & Mass Communication Quarterly* 73, 3 (1996), 687–707.
- [8] Minjie Li. 2021. The synergistic effects of solutions journalism and corporate social responsibility advertising. *Digital Journalism* 9, 3 (2021), 336–363.
- [9] Wayback Machine. [n.d.]. *Wayback Machine - The internet archive*. <https://newsapi.org/>
- [10] Karen McIntyre. 2019. Solutions journalism: The effects of including solution information in news stories about social problems. *Journalism Practice* 13, 1 (2019), 16–34.
- [11] Michael F Meffert, Sungeun Chung, Amber J Joiner, Leah Waks, and Jennifer Garst. 2006. The effects of negativity and motivated information processing during a political campaign. *Journal of Communication* 56, 1 (2006), 27–51.
- [12] Rachel A Millstein, Wei-Jean Chung, Bettina B Hoeppner, Julia K Boehm, Sean R Legler, Carol A Mastromauro, and Jeff C Huffman. 2019. Development of the state optimism measure. *General Hospital Psychiatry* 58 (2019), 83–93.
- [13] MonkeyLearn. [n.d.]. *MonkeyLearn Keyword Extractor API*. <https://monkeylearn.com/keyword-extractor-online/>
- [14] Solutions Journalism Network. [n.d.]. *Story Tracker*. <https://www.solutionsjournalism.org/storytracker>
- [15] Solutions Journalism Network. [n.d.]. *What is Solutions Journalism?* <https://www.solutionsjournalism.org/about/solutionsjournalism>
- [16] John E Newhagen and Byron Reeves. 1992. The evening's bad news: Effects of compelling negative television news images on memory. *Journal of Communication* 42, 2 (1992), 25–41.
- [17] NewsApi. [n.d.]. *NewsAPI for the latest News*. <https://newsapi.org/>
- [18] Christian Staal Bruun Overgaard. 2021. Mitigating the consequences of negative news: How constructive journalism enhances self-efficacy and news credibility. *Journalism* (2021), 14648849211062738.
- [19] Martin Schrepp and Jörg Thomaschewski. 2019. Construction and first Validation of Extension Scales for the User Experience Questionnaire (UEQ). <https://doi.org/10.13140/RG.2.2.19260.08325>
- [20] GS Statistics. [n.d.]. *Statcounter Global Stats Browser Market Share Worldwide 2022*. <https://gs.statcounter.com/>
- [21] Attila Szabo and Katey L Hopkinson. 2007. Negative psychological effects of watching the news in the television: Relaxation or another intervention may be needed to buffer them! *International journal of behavioral medicine* 14, 2 (2007), 57–62.
- [22] New York Times. [n.d.]. *New York Times - Fixes*. <https://www.nytimes.com/column/fixes>
- [23] David Watson, Lee Anna Clark, and Auke Tellegen. 1988. Development and validation of brief measures of positive and negative affect: the PANAS scales. *Journal of personality and social psychology* 54, 6 (1988), 1063.
- [24] Brita Ytre-Arne and Hallvard Moe. 2021. Doomscrolling, monitoring and avoiding: news use in COVID-19 pandemic Lockdown. *Journalism Studies* 22, 13 (2021), 1739–1755.