Measuring selective exposure in mock website experiments: A simple, free, and open-source

solution

Julian Unkel

LMU Munich

POST-PRINT

2019-03-20

Author Note

Julian Unkel, Department of Media and Communication, LMU Munich, Oettingenstr. 67, 80538 Munich, Germany. E-mail: unkel@ifkw.lmu.de

Abstract

The mock website paradigm has been established as a methodological standard for experimental selective exposure research. To date, despite the amount of empirical research utilizing mock website experiments, no openly accessible practical implementations tailored to the needs of communication scholars exist. In this paper, we first review extant implementations of mock website experiments in selective exposure research and outline the methodological shortcomings. We then outline a procedure to conduct mock website experiments that addresses these shortcomings using only freely available software and that does not restrict participants to specific devices nor require additional software to be installed. Additionally, it can be employed and easily adapted without further programming knowledge as all scripts are published open-source on GitHub. We also demonstrate an application of the procedure by outlining a study on selective exposure to search engine content ($N = 1,187$). Further applications and limitations of the procedure are discussed.

Measuring selective exposure in mock website experiments: A simple, free, and open-source

solution

The question of how information selectivity affects the occurrence or absence of (media)

effects is one of the most prolific and long-standing debates in the field of communication and its

adjacent disciplines (Knobloch-Westerwick, 2015; Smith, Fabrigar, & Norris, 2008). For

example, a recent systematic review of selective exposure research found more than 7,000

articles that have been published in more than 3,000 academic journals since the 1960s,

highlighting the heterogeneity, long tradition, and ongoing expansion of research into

information selectivity (Domahidi & Günther, 2017). The emergence of new communication

technologies not only invigorated the theoretical debates about selective exposure (e.g., Bennett

& Iyengar, 2008; Garrett, 2017; Holbert, Garrett, & Gleason, 2010; Katz & Fialkoff, 2017), but

has also lead to methodological innovations that seek to improve the validity and reliability of

empirical findings about recipients' selectivity (Garrett, 2013).

While the long-term productivity of the research field has led to a number of

methodological approaches (for a review of the different techniques, see Clay, Barber, & Shook,

2013), one type of research design may be considered as the "current state of the art" (Leiner,

Scherr, & Bartsch, 2016, p. 203): the "mock website paradigm" (Clay et al., 2013, p. 159), most

prominently refined by Hastall and Knobloch-Westerwick (2013). This research paradigm

involves participants selecting links from an experimentally manipulated hub page and has been

employed in a variety of research contexts and studies in the last two decades (Clay et al., 2013).

However, conducting studies incorporating the mock website paradigm may result in substantial

technical challenges to researchers, as both the setup and implementation usually require

advanced programming skills and knowledge about web technologies (Hastall & Knobloch-

Westerwick, 2013). To date, few flexible and openly accessible software solutions have been

provided (Leiner et al., 2016).

In response to calls for "the development of open-source software tools for convenient

implementation in different research contexts or the development of easy-to-use templates

implemented in existing software tools" (Hastall & Knobloch-Westerwick, 2013, p. 104), we aim

to provide a practical solution for measuring selective exposure in mock website experiments

that is tailored to the needs of communication scholars working in experimental selective

exposure research and that can be easily adapted to a multitude of other research contexts. After

outlining and discussing extant implementations of the mock website paradigm, we show how

such research may be conducted using only widespread web programing languages and freely

available research tools. We provide easy-to-adapt, open-source scripts for our measurement

tool, detail the necessary steps to set up a selection experiment, and illustrate our approach with a

study on selecting search engine content. Thus, comparable studies may be conducted with little

to no further programming skills required by researchers as well as no specific hardware or the

installation of additional software being required by participants.

Choice experiments and the mock website paradigm in selective exposure research

While the term "selective exposure" has been—and, especially in media psychology

contexts, still is—traditionally associated with a confirmation bias mechanism focusing on

attitude-congruent media exposure (Smith et al., 2008), recent communication scholarship

follows a broader understanding of the concept that includes "any systematic bias in selected

messages that diverges from the composition of accessible messages" (Knobloch-Westerwick,

2015, p. 6; see also Brosius & Peter, 2011; Bryant & Davies, 2008). As such, selective exposure

has been empirically studied in a wide variety of media contexts and conceptualized via several

theoretical perspectives. For example, Hartmann (2009) attempts to classify extant theoretical

perspectives into four meta-theoretical approaches, including neo-behavioral theories, decision

theory-like approaches, action-theory approaches, and integrative models. Other recent reviews

of the concept (e.g., Knobloch-Westerwick, 2015) further highlight the variety of theoretical

perspectives used to study selective exposure as well as the integration of the concept into other

theoretical frameworks such as the reinforcing spirals model (Slater, 2007, 2015).

This plethora of theoretical perspectives is mirrored in a similar diversity of

methodological approaches. Key methodological decisions in the study of selective exposure

concern the investigation of 1) reported versus observed behavior, 2) the consideration of

individual versus aggregate data, and 3) the use of "real" versus experimentally manipulated

stimuli. While these decisions should depend on the research interest at hand and are discussed at

length in the selective exposure methodology literature (e.g., Clay et al., 2013; Feldman, Stroud,

Bimber, & Wojcieszak, 2013; Knobloch-Westerwick, 2015; Leiner et al., 2016), the first point is

of particular importance to the proposed methodological procedure. Knobloch-Westerwick and

colleagues (2015) criticize how a large percentage of the empirical results stem from survey-

based self-reports, either relying on retrospective accounts of selection behavior or addressing

behavioral intentions. However, the validity of self-reports is limited by distortions due to

misjudgments, ex-post rationalization, and socially desirable response behaviors (Clay et al.,

2013; Scharkow, 2016). Observations may therefore provide more reliable and valid measures

than self-reports can—especially in the online context, where digital trace data allow for the

automated, unobtrusive, and granular measurement of selection behavior (Keyling & Jünger,

2016; Wieland et al., 2018). At the same time, observations do not usually provide us with

information on how the participants actually perceived the media content. Thus, the recent

methodological literature favors observations of actual selection behavior over self-reported

behavior, combined with introspective measures obtained through additional questionnaires that

provide a more in-depth understanding of participants' perceptions of selection situations and the

reasons behind their selection decisions (Hastall & Knobloch-Westerwick, 2013; Leiner et al.,

2016).

The mock website paradigm

In the case of experimental study designs, most studies opt for a sub-type of experimental

design called "choice experiments." In contrast to traditional, between-subjects experiments

where participants are assigned randomly to experimental conditions (e.g., different variations of

manipulated media stimuli), choice experiments let the participants choose freely between the

available options (Feldman et al., 2013). This creates a more realistic, externally valid selection

situation, as participants are usually able to select as many or as few options as they like

(however, researchers may limit the maximum number of options that can be selected, introduce

a time limit, or allow participants to abandon the selection task of their own accord).

The "mock website paradigm" (a term coined by Clay et al., 2013) combines this type of

experimental logic with manipulated web stimuli that unobtrusively track selection decisions.

Stimuli are built with a "hub-and-spoke" navigation logic (Hastall & Knobloch-Westerwick,

2013), consisting of a hub page that contains links (i.e., the options to select from) to other pages.

This navigation logic only allows for one-at-a-time, sequential selection, as spoke pages are

linked to from the hub page but are not linked to each other. The selection task may be

accompanied by pre- and post-task questionnaires assessing other individual-level variables of

interest.

The mock website paradigm has been established as a standard in experimental selective exposure research and is called the "current state of the art" (Leiner et al., 2016, p. 203), not least because it closely resembles the actual selection in several online contexts, for example, when selecting results from a search engine results page, choosing posts from a social network site's (SNS's) news feed, or picking out articles from a news site. As such, mock website experiments have been employed in a variety of online media contexts such as news magazines (Knobloch-Westerwick & Meng, 2011), blogs (Winter & Krämer, 2012), search engines (Knobloch-Westerwick, Mothes, Johnson, Westerwick, & Donsbach, 2015; Unkel & Haas, 2017), news aggregators (Sundar, Knobloch-Westerwick, & Hastall, 2007; Winter & Krämer, 2014), and social media (Messing & Westwood, 2014; Sülflow, Schäfer, & Winter, 2018).

Apart from the high external validity and the unobtrusive measurement of selection decisions, further key strengths of the mock website paradigm and choice experiments lie in their ability to allow for several different facets of selective exposure (e.g., selection, the order of selection, exposure time, moderators and mediators of selection decisions) to be obtained at the same time (Clay et al., 2013). The disadvantages include the disputed generalizability of online selection behavior in other mediated contexts and the necessity of restricting selection options and the factors that are experimentally varied in line with the experimental design. Furthermore, the investigation of dependent variables that are measured or assessed after the initial selection decision may be complicated by participant self-selection that negates the crucial randomization that takes place in many experiments (Feldman et al., 2013). Finally, mock website experiments generally build on the researchers' interpretation of the stimulus (e.g., the credibility or attitude-consistency of the selectable options) and not on the participants' perceptions of the selection

decision (i.e., whether the selectable options were actually perceived as intended; Clay et al., 2013).

Extant implementations of mock website experiments

The multitude of studies from different research contexts that employ mock website experiments corresponds to a large variety of methodological implementations. Practical problems may arise from maintaining the integrity of the experimental situation, the merging of observed data with additional survey data, and the (technical) requirements the study setup places on both researchers and participants. One distinct problem is that even with carefully created website stimuli, it is difficult to control whether the participants remain in the experimental setting or not. For example, links in Hypertext Markup Language (HTML) pages can be opened in new browser tabs or windows, thus negating the sequential hub-and-spoke selection logic.[1] Additionally, researchers may want to mandate (e.g., full-screen view) or prohibit (e.g., tabbed browsing) the use of specific browser features or deactivate the display of certain browser elements (e.g., the browser's info bar) to enforce the boundaries of the experimental setting and disguise the manipulated nature of the stimulus. Thus, several study setups require the prior configuration of hardware and software by researchers, for example, to record website interactions with video cameras or screen-capture software (e.g., *Microsoft Expression Encoder Screen Capture*; Unkel & Haas, 2017; Winter et al., 2016), to mandate the use of a specific, pre-configured browser (e.g., Tremel, 2010), or to facilitate the use of certain questionnaire and analysis software (e.g., *MediaLab/DirectRT*; Knobloch-Westerwick & Meng, 2011). The majority of mock website experiments are therefore carried out in university computer labs or comparable centralized settings (e.g., in Internet cafés or computer rooms in schools; Knobloch, Patzig, & Hastall, 2002; Tremel, 2010). Such centralized settings allow for

the extensive control of the study setup and situation, but also involve the usual caveats of lab

studies such as an increased awareness of participants regarding the study situation, or the much

more difficult and resource-intensive recruitment of participants. Accordingly, most laboratory

studies are conducted with student samples.

If the selection experiment is to be carried out in the "field"—which in most cases means

that subjects can participate in the study at home, on the move, or at work on their own

devices—, the aforementioned problems may again be addressed by using special software or

non-HTML stimuli. For example, participants may be required to install software on their own

devices from storage media that is sent to them (e.g., Iyengar, Hahn, Krosnick, & Walker, 2008)

or to download a browser plug-in (e.g., Knobloch-Westerwick & Johnson, 2014; Wieland et al.,

2018). Other study setups enforce the hub-and-spoke logic by creating interactive stimuli not as

HTML websites, but via specific web applications (e.g., *Microsoft Silverlight*; Knobloch-

Westerwick, Mothes, Johnson, Westerwick, & Donsbach, 2015) or as survey questions styled to

look like website navigation implemented directly into survey software (e.g., Engelmann &

Wendelin, 2017). In addition to the fact that most of these solutions are proprietary, unpublished

in-house developments, they also diminish the advantages gained from leaving the lab. The

installation of software might be perceived as intrusive and undesirable by potential participants

and the switch from HTML pages to other web technologies reduces the desired authenticity and

real-word feel of the situation. Furthermore, specialized third-party software such as *Microsoft

Silverlight* in 2017 may be discontinued.

Recently, the utilization of web analytics software such as *Google Analytics*, *Woopra,* or

Matomo (formely *Piwik*) that, by default, unobtrusively tracks user behavior on websites, has

been discussed for use in social science research. For the latter software, Leiner and colleagues

(2016) have developed a procedure and accompanying tool that enables the matching of

behavioral data tracked by Matomo with survey data from various types of online questionnaire

software for individual participants and prepares the obtained data for social science purposes.

The greatest strength of this solution is its flexibility: Even complex selection behavior outside

the hub-and-spoke logic can be recorded and its use is not limited to mock websites. For

selection experiments, however, there are also distinct disadvantages. If the experimental design

requires such a hub-and-spoke logic, this cannot be enforced due the above-mentioned

navigation capabilities of HTML links (opening links in new tabs, etc.). Furthermore, the use of

Matomo (and the HTML stimulus) requires the installation and configuration of a web server,

which may represent a technical hurdle. Most importantly, observation and survey data are stored

separately (in Matomo's database and the questionnaire software's database, respectively) and

can only be merged after data collection has ended, thus limiting the ability to obtain detailed

interpretations of participants' selection decisions.

Conducting mock website experiments with free-to-use survey software

Currently there is no solution to measure selective exposure in mock website experiments

that is both adapted to the needs and demands of communication research and is openly

available. In the following, we therefore discuss a procedure we have developed to conduct mock

website experiments that seeks to preserve the inherent strengths of such experimental designs

while at the same time addressing several of the weaknesses and disadvantages reviewed above.

Building on the technique outlined by Hastall and Knobloch-Westerwick (2013), the proposed

procedure enables researchers to 1) unobtrusively measure selection behavior in mock website

experiments with a hub-and-spoke logic, and 2) obtain additional self-reports through

questionnaires placed before and/or after the selection task. Furthermore, 3) observation data and

questionnaire data are stored in the same database and are thus already accessible while the

respective participant is still filling out the questionnaire. Applicable to a wide variety of

research contexts, 4) the proposed procedure does not restrict participants to certain devices or

certain browsers and does not obligate them to install additional software. Finally, to increase

accessibility and promote scientific openness, the procedure 5) can be employed by researchers

without further programming knowledge, 6) uses only software and technologies that are free of

charge for academic research, and 7) may be easily adapted for specific research interests, as all

included scripts are readily available from an open-source repository on *GitHub*:

https://github.com/joon-e/OSEM.

Overview

      The procedure is implemented by integrating the selection task directly into the

questionnaire with scripts in PHP and JavaScript. Questionnaire pages in most survey software

are HTML forms whose contents are stored in a database when the forms are sent (i.e., the

"submit" button is clicked), whereupon the next form (i.e., the next questionnaire page) is

displayed. For the selection task, two questionnaire pages are inserted into the questionnaire

between the two "regular" parts of the survey (pre- and post-task questionnaire). The first of

these pages contains the "hub" page; the second page handles the display of the "spoke" pages,

conditional on the participants' selection decisions on the hub page. To ensure a realistic look for

the hub and spoke pages, all regular questionnaire elements (e.g., questionnaire headers and

footers, "submit" buttons) are hidden on these pages. When the hub page is loaded for the first

time, a loop is initialized that interrupts the regular, linear questionnaire order, loads the spoke

page when the hub page is submitted, and vice versa. Clicking on predefined HTML elements on

the hub and spoke pages (i.e., spoke links on the hub page or a hub link on the spoke pages)

leads to the scripts to submit the current questionnaire page (and thus display the next page in the selection task loop) and store all behavioral variables associated with this selection in the questionnaire database. Therefore, participants can browse the stimulus in a hub-and-spoke logic just like on any other webpage. After an end condition is met—either a time or a selection limit is reached, or the participant manually aborts the selection task through an "abort" button, or any combination of these—the loop is broken and the questionnaire returns to its regular, linear flow, starting with the first page of the post-task questionnaire.

Client-side behavior tracking with JavaScript

  Participants' behavior during the selection task is tracked via JavaScript. JavaScript is a programming language that is mainly used to add interactive features to web pages and is by far the most common programming language on the Internet—about 95% of all websites make use of its features (W3Techs, 2018). As primarily a client-side language, JavaScript files are executed by web browsers and as such they are able to record all user actions on web pages (e.g., clicks, cursor movements) as well as obtain additional information (e.g., browser window size, coordinates and dimensions of HTML elements). All recent versions of major web browsers have JavaScript functionality activated by default, which makes it highly probable that potential participants will be able to run JavaScript files on their devices without further actions being required (approximately 99% of web users have JavaScript enabled; Herlihy, 2013). Our script records the following user actions and additional information:

1) On the hub page, we record all user clicks on spoke links with a numeric, distinct *ID* for the selected spoke link, a *timestamp* that is accurate to the millisecond, the *duration* between the loading of the hub page and the click in milliseconds (i.e., the dwell time on the hub page), and the *number of the selection decision* (i.e., first

selected spoke, second selected spoke, etc.). The conditional display of the spoke on

the spoke questionnaire page is then based on the recorded spoke link ID.

2) On the spoke page, in addition to the *ID* of the spoke (which corresponds to the ID of

the spoke link on the hub page), the *dwell time* on the respective spoke is also

recorded in milliseconds. If a single spoke is selected more than once during the

selection task, the dwell time is measured separately for each individual selection

decision.

3) Furthermore, we employ *cursor tracking* on the hub page. In lab studies, eye-tracking

methods are most commonly used to investigate the viewing patterns of websites

during selection tasks (e.g., Sülflow et al., 2018; Vraga, Bode, & Troller-Renfree,

2016). However, studies have shown that cursor movements may act as a proxy for

gaze positions (Guo & Agichtein, 2010; Huang, White, & Dumais, 2011; Rodden, Fu,

Aula, & Spiro, 2008). At the very least, cursor positions allow for inferences to be

made about what parts of the website stimulus were visible in the participant's

browser window. Because cursor tracking can quickly generate very large amounts of

data, we opted for a compromise between accuracy and parsimony (see Huang et al.,

2011). Every 100 ms, a JavaScript function checks whether the cursor has moved

more than 5 px in a horizontal and/or vertical direction compared to the last time—if

this is the case, the new position is saved, otherwise it is discarded. A total of up to

300 cursor positions are recorded per hub page visit. If no cursor information is

available (e.g., when the participant uses a smartphone or tablet to take part in the

study), a missing value is stored instead.

4) Finally, to account for different screen and browser window sizes, the *size of the browser window* when starting the selection task (X and Y values in pixels) as well as the *position and measurements of all spoke links* (X and Y values of the dimensions of the corresponding HTML element in pixels) are recorded as well.

Implementation into free-to-use survey software

In the following, the main steps necessary to implement the procedure outlined above into the survey software *SoSci Survey* (www.soscisurvey.com) are described. SoSci Survey allows researchers to construct online questionnaires by setting up a variety of different question formats and placing them on questionnaire pages via a drag-and-drop system. We choose SoSci Survey for multiple reasons. Most importantly, the software supports custom scripts in JavaScript and PHP as well as custom questionnaire layouts using HTML, Cascading Style Sheets (CSS), and Extensible Markup Language (XML), and is available to use free of charge without advertisements for academic research. The user interface can be set to either English or German, while the questionnaires support any number of languages, including non-Roman alphabets. Last but not least, the data are stored in accordance with the European Union's General Data Protection Regulation (GDPR). All necessary script files for the implementation into SoSci Survey as well as a demo project including HTML stimulus files and detailed instructions on how to set up the project are provided on the GitHub repository.

Experiment planning and stimulus preparation. After deciding on the main parameters for the selection experiment (experimental factors and the resulting number of spokes, end conditions), the hub and spoke pages of the experiment can be constructed just like any other HTML web page, including images, custom CSS files, and interactive JavaScript features. Importantly, all HTML files must follow a strict hub-and-spoke logic; that is, all links

on the hub page must point to one of the spoke pages and all links on a given spoke page must lead back to the hub page.

To prepare the HTML stimulus files for implementation into SoSci Survey, some small modifications of the HTML source code are necessary. Instead of using actual HTML links (<a> HTML tags), certain HTML attributes have to be added to each HTML element on the hub page that should act as a spoke link: specific class attributes that lead the provided scripts to recognize a given HTML element as a spoke link, and a specific data attribute that points to the numeric ID of the spoke page that the HTML element should link to (see Figure 1). Similarly, <a> tags on the spoke pages have to be removed and specific class attributes have to be added to the corresponding HTML elements as well. To ensure that these new "links" have the look and feel of actual HTML links, additional styling information (e.g., underlined text and a "pointer" cursor when hovering) may be added to the HTML elements.

SoSci Survey project setup and script upload. The pre- and post-task questionnaires can be created in SoSci Survey without limitations using default software functionality. For the selection task, two blank questionnaire pages—which will later display the hub and spoke pages, respectively—have to be inserted at the desired place in the questionnaire. When conducting studies outside of a centralized setting like a computer lab, it is advisable to obtain further contextual information about the participants as control variables. SoSci Survey offers several of such control variables including information about the device (e.g., computer, smartphone, or tablet), operating system, and the browser used by the participants as well as whether they have JavaScript enabled. It is also advisable to place a questionnaire page containing instructions for the selection task before the actual task.

After downloading the script files from the GitHub repository, they can now be uploaded to SoSci Survey. These script files include:

- A JavaScript file responsible for tracking the selection behavior (see above). This file can be uploaded using SoSci Survey's media file upload form.

- An XML layout file that removes any default questionnaire styling, hides all questionnaire elements, and loads the JavaScript file on the hub and spoke pages. This file can be uploaded using SoSci Survey's layout upload form.

- Five XML question element files that are needed to store the selection behavior variables in the questionnaire database. These can be uploaded using SoSci Survey's question upload form.

- Two PHP files (for the hub and spoke questionnaire page, respectively) that are used to configure the selection task, initialize and break the loop of hub and spoke pages in the questionnaire flow, and display the hub and spoke pages. While SoSci Survey does not have an upload form for PHP files, the PHP code can simply be copied and pasted onto PHP elements provided by SoSci Survey that are placed on the questionnaire pages (see Figure 2).

Script configuration. While no further modifications of the scripts are necessary, the PHP code includes two variables that can be used to further configure the selection task. First, end conditions (selection limit, time limit, manual abort) can be defined by setting the corresponding values (see Figure 2). Second, by default, a bar is displayed on top of the stimulus pages that includes information about the defined end conditions (e.g., a timer when using a time limit, or an "abort" button when using manual abort) and, if on a spoke page, a button to return to

the hub page (see Figure 3). This bar can also be customized or disabled by setting the corresponding values.

     Stimulus upload. Finally, the prepared HTML stimulus files (i.e., the hub and spoke pages) have to be uploaded and integrated into the questionnaire. The HTML code of each stimulus file has to be copied into separate HTML blocks from SoSci Survey's text element feature, which can then be named with a unique identifier (e.g., "hub" for the hub page, and "spoke1," "spoke2," etc. for the spoke pages). A single line of PHP may then be used to place the HTML code on the corresponding questionnaire pages. For example, the PHP expression "text('hub');" can be used to display the hub page on the hub questionnaire page if using the naming scheme suggested above, while "text('spoke' . $spokeID);" will conditionally display the spoke pages on the spoke questionnaire page.[2] While these examples so far have assumed that all participants see the same static hub page, rotated or randomized spoke links or versions of the hub page are also possible, but may require some additional PHP code.

Data handling and analysis

     All observational variables are stored alongside the survey variables in the same dataset on a case (participant) level. For example, one variable per selection decision stores the numeric ID of the spoke selected, while another variable per selection decision stores the corresponding timestamp. Cursor-tracking data are stored per selection decision as three vectors of x-values, y-values, and timestamps, respectively. As such, the survey software can already access all the observational data during the post-task questionnaire stage. For example, researchers may want to verify whether a participant has spent a certain period of time with the selection task or has selected a certain number of spokes, or they may want to display specific questions based on the observed selection behavior (e.g., participants' perceptions of the first spoke selected or the

spoke with the longest dwell time). Furthermore, SoSci Survey offers a range of aggregate functions to be used in the questionnaire that aggregate or count values over all questionnaires completed so far, which may, for example, be used to identify participants with unusual selection behaviors.

Depending on researchers' strategies for data analysis, it may be necessary to restructure the data. On the GitHub repository, we provide scripts for R to restructure the data from a case/participant level (i.e., one row per participant) to the selection (i.e., one row per selected spoke per selection decision per participant), decision (i.e., one row per selectable spoke per selection decision per participant), or spoke level (i.e., one row per selectable spoke with aggregated selection information, for example, the total number of selections). In the selection- and decision-level cases, participant-level variables (e.g., participant variables obtained during the pre- and post-task questionnaires) will simply be copied into the corresponding rows, whereas these variables may be aggregated to be used on a spoke level (e.g., mean values of those participants who selected a given spoke). We additionally provide scripts to "unpack" the cursor-tracking vectors.

Practical application: Selective exposure to search engine content

We demonstrate a practical application of the procedure by outlining a study on selective exposure to search engine content conducted in March 2018 and completed by 1,187 participants. The participants were told to inform themselves about one of two political issues for up to five minutes through browsing a manipulated search engine stimulus. This stimulus consisted, per issue, of one search engine results page (SERP) displaying 12 search results, corresponding to 12 target pages. Accordingly, the SERP served as the hub page, while all target pages were spokes that could only be accessed by selecting a search result on the SERP. The

search results were experimentally varied in a 3 (stance on the issue: pro, contra, neutral) × 2 (recency: two weeks old, several months old) × 2 (source credibility: high, low) within-subject design, with all participants seeing the same 12 search results on the SERP but in a randomized order. The SERP was modeled after regular Google results pages (see Figure 4). During the selection task, which had a time limit of five minutes and an option to manually abort it, participants could select as many or as few search results as they liked. The pre-task questionnaire focused on issue-dependent variables (prior knowledge, stance on the issue, personal relevance) and included detailed instructions for the search task, while the post-task questionnaire included an evaluation task of all 12 search results as well as several questions about participants' personal characteristics (e.g., need for cognition, socio-demographics). The participants were recruited via a non-commercial online access panel and were able to fill out the questionnaire on their own devices.

We conducted a face-to-face pretest of the study with five colleagues who were unacquainted with the procedure, during which we also recorded their actions with screen-capture software. This pretest did not uncover any issues regarding the functionality and navigability of the stimulus. Two anonymous reviewers who inspected the questionnaire for the non-commercial online access panel also did not report any issues regarding the stimulus. The selection task also did not lead to an increased dropout rate for participants during the main study, with about 12.5% of all dropouts during the study occurring during the selection task. The descriptive statistics of participants' search behavior show that the stimulus was used as intended. On average, participants selected a little less than four search results ($M = 3.82$, $SD = 1.76$), browsed the SERP in total for about one minute ($M = 60.53$ s, $SD = 37.34$), and spent about three minutes on the target pages ($M = 178.08$ s, $SD = 80.63$), with more than half of

all participants (54.3%) utilizing the full five minutes for the selection task (and thus not

manually aborting it). During the selection task, participants mainly employed heuristic search

strategies and focused on the top-ranked search results, which was to be expected when

considering the results of "real" search engine use (e.g., Agichtein, Brill, Dumais, & Ragno,

2006; Pan et al., 2007): less than one-fifth of all participants went "below the fold" during their

SERP visit (i.e., scrolled past the search results that were initially visible in their browser

window); a heat map based on the cursor-tracking data shows a concentration on the five top-

ranked search results (see Figure 4); and the search result rank emerged as the overall strongest

predictor of participants' selection decisions. Thus, these results indicate that the participants'

behavior during the experimental procedure can be compared to actual usage situations.

Discussion

The inherent selectivity in online media use has invigorated selective exposure research.

Both theoretical and methodological improvements have brought forward reconsiderations and

reevaluations of extant findings, while new communication channels such as SNSs or search

engines, and new selection factors such as aggregated social information cues have generated

new research interests (Knobloch-Westerwick, Westerwick, et al., 2015). Mock website

experiments have become the method of choice when addressing online selective exposure in

experimental research designs, as they closely resemble the actual selection of content in many

online contexts and allow for the unobtrusive and valid observation of actual selection behavior

(Clay et al., 2013). However, while the general procedure to conduct mock website experiments

has been widely discussed and a "gold standard" theoretical implementation has been established

(Hastall & Knobloch-Westerwick, 2013; Leiner et al., 2016), practical solutions are still not

readily available. Extant implementations are oftentimes proprietary, study-specific solutions that

have not been further developed or published for widespread use, or they make use of tools that entail some critical disadvantages for experimental research. Therefore, we have developed a practical implementation of mock website experiments into survey software called "OSEM (Online Selective Exposure Measurement) for SoSci" that addresses these methodological shortcomings and is available as open-source scripts on GitHub: https://github.com/joon-e/OSEM.

To the best of our knowledge, the proposed procedure and accompanying measurement tool is the first openly accessible solution to conduct mock website experiments that is available free of charge for social science research. The described procedure and provided scripts enable the direct integration of selection tasks with interactive HTML stimuli into free-to-use survey software. Participants' selection behavior is measured unobtrusively and stored in the same dataset as the questionnaire data, making all observational variables accessible to the survey software already during the post-task questionnaire stage. The integration into browser-based survey software and the use of JavaScript to track the selection behavior mean that the vast majority of potential participants are neither restricted to a specific device nor required to install additional software.

As others have noted, mock website experiments require researchers to overcome "substantial technical hurdles" (Leiner et al., 2016, p. 202) and to have "specific programming skills" (Hastall & Knobloch-Westerwick, 2013, p. 103). Because all of the necessary script files for our procedure are openly available and easily adaptable to specific research designs, no further programming knowledge is required by the researchers apart from the construction of the HTML stimulus. However, we presume that sufficient HTML knowledge is much more common than an understanding of more in-depth web-programming languages. Furthermore, online

templates and what-you-see-is-what-you-get (WYSIWYG) editors such as *Adobe Dreamweaver* may assist researchers in creating good-looking stimuli even without substantial HTML knowledge. We thus argue that our procedure considerably lowers the technical barriers to conducting mock website experiments.

In the introduction, we listed three key methodological decisions researchers face when designing selective exposure studies. While the procedure described in this article has been developed specifically to address the decision between self-reports and observed behavior in that it allows for the straightforward combination of both types of measurements, it may provide some guidance for the other two decisions as well. First, selection behavior data are stored on an individual level, but can easily be analyzed on aggregated levels, for example, on the level of single spokes. Second, although developed with experimentally manipulated stimuli in mind, the procedure can be used to closely resemble "real" stimuli and usage situations: The provided scripts place few restrictions on the number (up to 99) and visual appearance of the spokes/spoke links, and include different end conditions, which allows researchers to approximate a variety of naturalistic use conditions (e.g., the presence of non-relevant information, selection tasks with unlimited time and with only a manual abort procedure; see Kessler & Zillich, 2019). Furthermore, as stimuli are implemented through a combination of PHP and HTML code, they can be updated or changed regularly during a study, for example, with real-world data scraped from websites.

As such, our proposed solution is applicable in a wide variety of research contexts, either conducted in the lab or in the field. Through integration into survey software, the selection task can easily be combined with questionnaires as well as with other tasks (e.g., reading and comprehension tasks, multimedia treatments, implicit association tests). In addition, the

measurement scripts may also be used in studies that do not focus on selective exposure *per se* but utilize an interactive stimulus and aim to include behavioral observations as mediators or moderators. Moreover, cursor-tracking data and the automatically recorded spoke link positions may assist researchers conducting eye-tracking experiments to define areas of interest and substantiate fixation patterns.

Limitations

The described procedure is not without limitations. First, while HTML stimuli provide a natural usage situation, some restrictions to the regular navigation features have to be made to maintain the hub-and-spoke logic and the functionality of the survey software. Because the form elements require the respective questionnaire page to be "submitted," participants are not able to use the browser's navigation (i.e., the back and forward buttons that in most survey software should also not be used during "regular" questionnaires), but instead have to use navigation buttons provided on the stimulus pages (e.g., a "back to hub" button in the top bar). Additionally, the regular functions of the middle (new tab) and right mouse buttons (context menu) are deactivated and are instead interpreted as clicks with the left mouse button. These restrictions should be addressed in the introduction to the selection task. In our sample study, we also encouraged participants to complete the questionnaire in their browser's full-screen mode, which in many browsers hides the navigation buttons and shields participants from other distractions on their screens. However, technical solutions that allow for more intuitive and natural navigation in the experimental stimulus should be considered in future refinements of the procedure. This includes the use of the browser's back and forward buttons, which is dependent on the survey software supporting this functionality.

Second, JavaScript has to be enabled in the participants' browsers. While the widespread use and availability of JavaScript (Herlihy, 2013) indicates that this should not be an issue for most potential participants, this may be of concern for those studies focusing on specific target populations who are prone to disabling JavaScript, for example, due to visual impairments or privacy concerns (see Leiner et al., 2016). In our sample study, we included JavaScript checks and, if disabled, instructions were offered on how to activate it. We encourage others to do so as well, not least because regular survey software functions may also be affected by disabled JavaScript.

Other procedure controls and checks should be employed as well, especially when conducting studies outside of pre-configured, controlled settings (e.g., a computer lab). SoSci Survey includes measures for several control variables, including the time spent on each questionnaire page, JavaScript functionality, and the used device, browser, screen size, and operating system. Our scripts additionally provide measures for the browser window size and the position and measurements in pixels of all spoke links to control for differences in stimulus rendering. Nevertheless, stimuli should be extensively pre-tested on a variety of different devices and browsers to ensure the consistent display of the stimulus for all participants.

Finally, the procedure is currently limited to a single survey software: SoSci Survey. We chose this software as, to the best of our knowledge, no other professional survey software allows for custom scripts and layouts and is available free of charge for academic research. However, all script files are available in an open-source format and we encourage adaptations of the procedure for other survey software.

Conclusion

Despite recent advantages in selective exposure research methodology, several methodological concerns remain, including the specificity of both behavioral and attitudinal measures, the *a priori* classification of information sources by the researchers, and the measurement of post-exposure interpretations of the selection situation (Clay et al., 2013; Feldman et al., 2013). While some of these concerns primarily pertain to study conceptualizations, they can and should also be addressed by software solutions tailored to the needs of selective exposure researchers. The described procedure may alleviate these concerns by providing a variety of behavioral measures and the possibility of more precisely and directly measuring perceptions of the selection situation.

Our procedure adds to current open-source, free, and openly available developments such as *ExposureResearchTools* for Matomo (Leiner et al., 2016) that provide communication researchers with tools to investigate selective exposure from a wide variety of theoretical and methodological perspectives, with established survey software and without major technical requirements. At the same time, the implementation of research projects with freely distributed and openly accessible software solutions fosters joint solutions to the problems and concerns that may arise from these research designs.

References

Agichtein, E., Brill, E., Dumais, S., & Ragno, R. (2006). Learning user interaction models for

predicting web search result preferences. *Proceedings of the 29th Annual International

ACM SIGIR Conference on Research and Development in Information Retrieval*, 3–10.

https://doi.org/10.1145/1148170.1148175

Bennett, W. L., & Iyengar, S. (2008). A new era of minimal effects? The changing foundations of

political communication. *Journal of Communication*, *58*(4), 707–731.

https://doi.org/10.1111/j.1460-2466.2008.00410.x

Brosius, H.-B., & Peter, C. (2011). Selective exposure. In P. Moy (Ed.), *Oxford bibliographies

online: Communication*. Retrieved from

http://www.oxfordbibliographies.com/display/id/obo-9780199756841-0023

Bryant, J., & Davies, J. (2008). Selective exposure. In W. Donsbach (Ed.), *The international

encyclopedia of communication*. Retrieved from

http://www.communicationencyclopedia.com/subscriber/tocnode.html?id=g97814051319

95_yr2013_chunk_g978140513199524_ss27-1

Clay, R., Barber, J. M., & Shook, N. J. (2013). Techniques for measuring selective exposure: A

critical review. *Communication Methods and Measures*, *7*(3–4), 147–171.

https://doi.org/10.1080/19312458.2013.813925

Domahidi, E., & Günther, E. (2017, May 29). *60 years of research on selective exposure. A

systematic review*. Presented at the 67th Annual Meeting of the International

Communication Association, San Diego, CA.

Engelmann, I., & Wendelin, M. (2017). Comment counts or news factors or both? Influences on news website users' news selection. *International Journal of Communication*, *11*, 2501–2519.

Feldman, L., Stroud, N. J., Bimber, B., & Wojcieszak, M. (2013). Assessing selective exposure in experiments: The implications of different methodological choices. *Communication Methods and Measures*, *7*(3–4), 172–194. https://doi.org/10.1080/19312458.2013.813923

Garrett, R. K. (2013). Selective exposure: New methods and new directions. *Communication Methods and Measures*, *7*(3–4), 247–256. https://doi.org/10.1080/19312458.2013.835796

Garrett, R. K. (2017). On retiring concepts. *Annals of the International Communication Association*, *41*(1), 105–110. https://doi.org/10.1080/23808985.2017.1288553

Guo, Q., & Agichtein, E. (2010). Towards predicting web searcher gaze position from mouse movements. *Extended Abstracts on Human Factors in Computing Systems*, 3601–3606. https://doi.org/10.1145/1753846.1754025

Hartmann, T. (2009). A brief introduction to media choice. In T. Hartmann (Ed.), *Media choice. A theoretical and empirical overview* (pp. 1–9). New York, NY: Routledge.

Hastall, M. R., & Knobloch-Westerwick, S. (2013). Caught in the act: Measuring selective exposure to experimental online stimuli. *Communication Methods and Measures*, *7*(2), 94–105. https://doi.org/10.1080/19312458.2012.761190

Herlihy, P. (2013, October 21). How many people are missing out on JavaScript enhancement? Retrieved from Government Digital Service: https://gds.blog.gov.uk/2013/10/21/how-many-people-are-missing-out-on-javascript-enhancement/

Holbert, R. L., Garrett, R. K., & Gleason, L. S. (2010). A new era of minimal effects? A response to Bennett and Iyengar. *Journal of Communication*, *60*(1), 15–34. https://doi.org/10.1111/j.1460-2466.2009.01470.x

Huang, J., White, R. W., & Dumais, S. (2011). No clicks, no problem: Using cursor movements to understand and improve search. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1225–1234. https://doi.org/10.1145/1978942.1979125

Iyengar, S., Hahn, K. S., Krosnick, J. A., & Walker, J. (2008). Selective exposure to campaign communication: The role of anticipated agreement and issue public membership. *The Journal of Politics*, *70*(1), 186–200. https://doi.org/10.1017/S0022381607080139

Katz, E., & Fialkoff, Y. (2017). Six concepts in search of retirement. *Annals of the International Communication Association*, *41*(1), 86–91. https://doi.org/10.1080/23808985.2017.1291280

Kessler, S. H., & Zillich, A. F. (2019). Searching online for information about vaccination: Assessing the influence of user-specific cognitive factors using eye-tracking. *Health Communication*, *34*(10), 1150–1158. https://doi.org/10.1080/10410236.2018.1465793

Keyling, T., & Jünger, J. (2016). Observing online content. In G. Vowe & P. Henn (Eds.), *Political communication in the online world: Theoretical approaches and research designs* (pp. 183–200). New York, NY: Routledge.

Knobloch, S., Patzig, G., & Hastall, M. (2002). "Informational Utility". Einfluss von Nützlichkeit auf selektive Zuwendung zu negativen und positiven Online-Nachrichten. ["Informational utility". Effects of utility on selective exposure to negative and positive online news]. *Medien & Kommunikationswissenschaft*, *50*(3), 359–375.

Knobloch-Westerwick, S. (2015). *Choice and preference in media use: Advances in selective exposure theory and research*. New York, NY: Routledge.

Knobloch-Westerwick, S., & Johnson, B. K. (2014). Selective exposure for better or worse: Its mediating role for online news' impact on political participation. *Journal of Computer-Mediated Communication*, *19*(2), 184–196. https://doi.org/10.1111/jcc4.12036

Knobloch-Westerwick, S., & Meng, J. (2011). Reinforcement of the political self through selective exposure to political messages. *Journal of Communication*, *61*(2). https://doi.org/10.1111/j.1460-2466.2011.01543.x

Knobloch-Westerwick, S., Mothes, C., Johnson, B. K., Westerwick, A., & Donsbach, W. (2015). Political online information searching in Germany and the United States: Confirmation bias, source credibility, and attitude impacts. *Journal of Communication*, *65*(3), 489–511. https://doi.org/10.1111/jcom.12154

Knobloch-Westerwick, S., Westerwick, A., & Johnson, B. K. (2015). Selective exposure in the communication technology context. In S. S. Sundar (Ed.), *The Handbook of the psychology of communication technology* (pp. 407–424). Malden, MA: Wiley Blackwell.

Leiner, D. J., Scherr, S., & Bartsch, A. (2016). Using open-source tools to measure online selective exposure in naturalistic settings. *Communication Methods and Measures*, *10*(4), 199–216. https://doi.org/10.1080/19312458.2016.1224825

Messing, S., & Westwood, S. J. (2014). Selective exposure in the age of social media: Endorsements trump partisan source affiliation when selecting news online. *Communication Research*, *41*(8), 1042–1063. https://doi.org/10.1177/0093650212466406

Pan, B., Hembrooke, H., Joachims, T., Lorigo, L., Gay, G., & Granka, L. (2007). In Google we trust: Users' decisions on rank, position, and relevance. *Journal of Computer-Mediated Communication*, *12*(3), 801–823. https://doi.org/10.1111/j.1083-6101.2007.00351.x

Rodden, K., Fu, X., Aula, A., & Spiro, I. (2008). Eye-mouse coordination patterns on web search results pages. *Extended Abstracts on Human Factors in Computing Systems*, 2997–3002. https://doi.org/10.1145/1358628.1358797

Scharkow, M. (2016). The accuracy of self-reported Internet use—A validation study using client log data. *Communication Methods and Measures*, *10*(1), 13–27. https://doi.org/10.1080/19312458.2015.1118446

Slater, M. D. (2007). Reinforcing spirals: The mutual influence of media selectivity and media effects and their impact on individual behavior and social identity. *Communication Theory*, *17*(3), 281–303. https://doi.org/10.1111/j.1468-2885.2007.00296.x

Slater, M. D. (2015). Reinforcing spirals model: Conceptualizing the relationship between media content exposure and the development and maintenance of attitudes. *Media Psychology*, *18*(3), 370–395. https://doi.org/10.1080/15213269.2014.897236

Smith, S. M., Fabrigar, L. R., & Norris, M. E. (2008). Reflecting on six decades of selective exposure research: Progress, challenges, and opportunities. *Social and Personality Psychology Compass*, *2*(1), 464–493. https://doi.org/10.1111/j.1751-9004.2007.00060.x

Sülflow, M., Schäfer, S., & Winter, S. (2018). Selective attention in the news feed: An eye-tracking study on the perception and selection of political news posts on Facebook. *New Media & Society*. Advance online publication. https://doi.org/10.1177/1461444818791520

Sundar, S. S., Knobloch-Westerwick, S., & Hastall, M. R. (2007). News cues: Information scent and cognitive heuristics. *Journal of the American Society for Information Science and Technology*, *58*(3), 366–378. https://doi.org/10.1002/asi.20511

Tremel, A. (2010). *Suchen, finden-glauben? Die Rolle der Glaubwürdigkeit von Suchergebnissen bei der Nutzung von Suchmaschinen* [Search, find – trust? The role of search result credibility in search engine use] (Doctoral dissertation, LMU Munich). Retrieved from http://d-nb.info/1009952234/

Unkel, J., & Haas, A. (2017). The effects of credibility cues on the selection of search engine results. *Journal of the Association for Information Science and Technology*, *68*(8), 1850–1862. https://doi.org/10.1002/asi.23820

Vraga, E., Bode, L., & Troller-Renfree, S. (2016). Beyond self-reports: Using eye tracking to measure topic and style differences in attention to social media content. *Communication Methods and Measures*, *10*(2–3), 149–164. https://doi.org/10.1080/19312458.2016.1150443

W3Techs. (2018). Usage of client-side programming languages for websites. Retrieved from W3Techs: https://w3techs.com/technologies/overview/client_side_language/all

Wieland, M., In der Au, A.-M., Keller, C., Brunk, S., Bettermann, T., Hagen, L. M., & Schlegel, T. (2018). Online behavior tracking in social sciences: Quality criteria and technical implementation. In C. Stützer, M. Welker, & M. Egger (Eds.), *Computational social science in the age of big data: Concepts, methodologies, tools, and applications* (pp. 131–160). Cologne, Germany: Herbert von Halem.

Winter, S., & Krämer, N. C. (2012). Selecting science information in Web 2.0: How source cues, message sidedness, and need for cognition influence users' exposure to blog posts.

Journal of Computer-Mediated Communication, *18*(1), 80–96.

https://doi.org/10.1111/j.1083-6101.2012.01596.x

Winter, S., & Krämer, N. C. (2014). A question of credibility—Effects of source cues and

recommendations on information selection on news sites and blogs. *Communications*,

39(4), 435–456. https://doi.org/10.1515/commun-2014-0020

Winter, S., Metzger, M. J., & Flanagin, A. J. (2016). Selective use of news cues: A multiple-

motive perspective on information selection in social media environments. *Journal of

Communication*, *66*(4), 669–693. https://doi.org/10.1111/jcom.12241

Footnotes

[1]This can be problematic even in experimental settings that do not require a strict hub-and-spoke logic. Server-side tracking (e.g., via log files or PHP scripts) only logs the time a web page is loaded; if a participant opens several pages one after the other in different tabs, it is no longer possible to see how much time was spent on which tab. Client-side tracking (e.g., via JavaScript) may be used to track such non-linear selection behaviors, but will require additional scripting and data preparation (Leiner et al., 2016).

[2]The SoSci Survey PHP function "text()" displays a named text element on a questionnaire, the PHP variable "$spokeID" is provided by the script and contains the numeric ID of the last selected spoke, and the PHP operator "." concatenates two strings. Thus, if, for example, a spoke with the numeric ID "2" was selected on the hub page, the PHP expression "'spoke' . $spokeID" results in the string "spoke2."

Acknowledgments

Figures

```
<div class="spoke-box">
    <img src="1.png">
    <div class="spoke-headline"><a href="spoke1.html">[Link headline]</a></div>
    <div class="spoke-text">[Link text]</div>
</div>


<div class="spoke-box osemSpoke" data-osem-spoke="1">
    <img src="1.png">
    <div class="spoke-headline osemClickable">[Link headline]</div>
    <div class="spoke-text">[Link text]</div>
</div>
```

Figure 1. Sample HTML hub page excerpt before (upper code block) and after (lower code

block) preparation. The original HTML code consists of a <div> container element that wraps

around an <img> image element, a <div> element containing a headline and another <div>

element containing additional text, mirroring, for example, an article teaser on a news website.

The headline links to a spoke page using an <a> link tag. To prepare this block of code for our

procedure, we have added the class "osemSpoke" and the data attribute "data-osem-spoke='1'"

to the parent <div> element. This tells our scripts that this element should act as a spoke link and

link to the spoke with the numeric ID "1". We have furthermore removed the <a> tag and added

the class "osemClickable" to the headline <div> element, which tells the scripts that the hub

page should be submitted and the spoke page loaded when the participants click on the headline.

Figure 2. Script configuration in SoSci Survey. The provided PHP code can be copied into PHP

elements and placed on the corresponding questionnaire pages. The code for the hub page

contains two variables which can be set to individually configure the end conditions (in the

example provided above: selection limit of 20 selections, time limit of 180 seconds, and manual

abort possible) as well as the display and contents of the information bar (see Figure 3).

End selection task | Back to hub | 115 seconds left | 19 selections left

Figure 3. Information bar displayed at the top of the stimulus pages. The bar can be configured by editing the PHP code provided with the script files (see Figure 2). The "Back to hub" button is only displayed on the spoke pages.
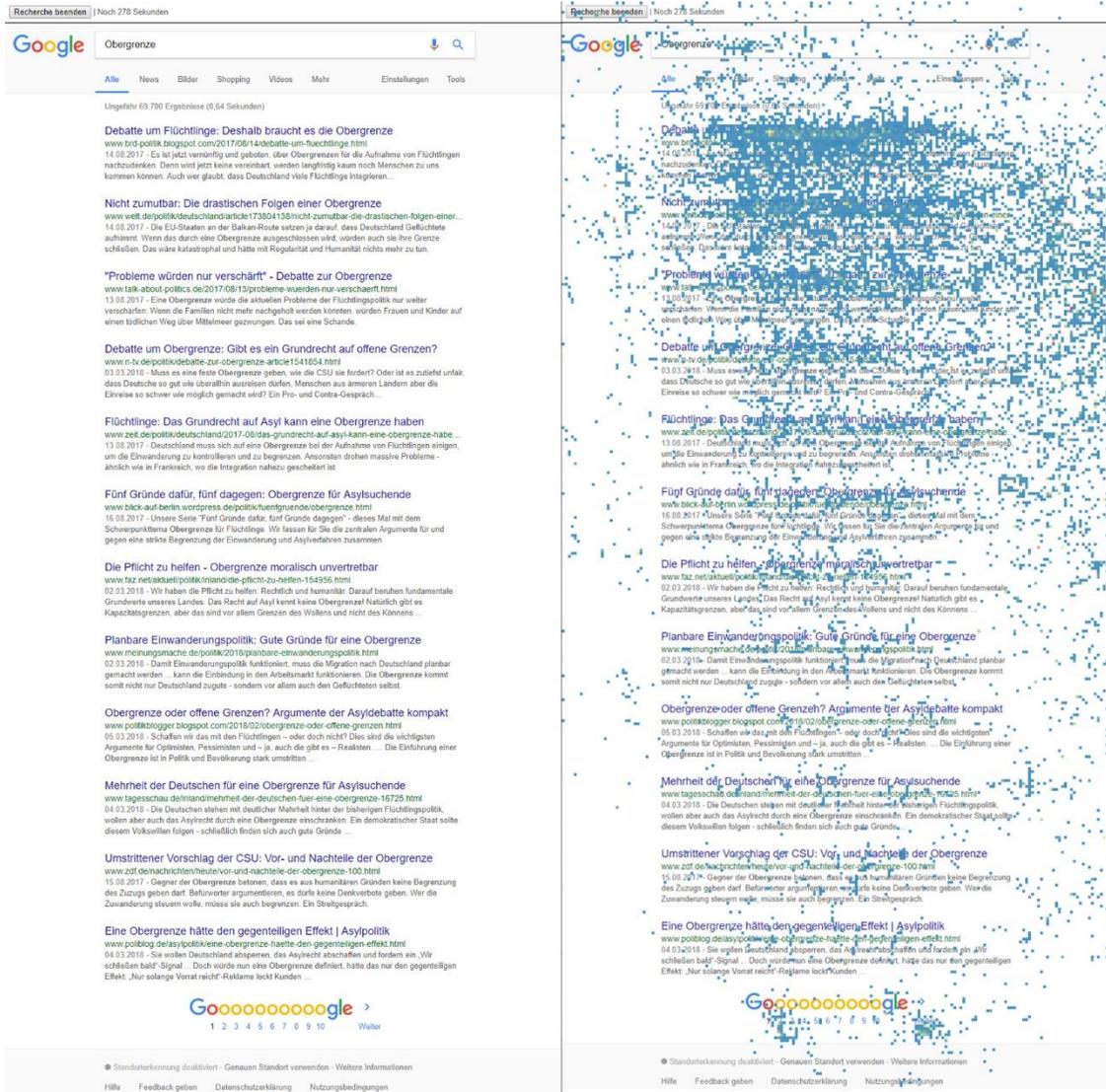
Figure 4. Stimulus (hub page) used in the study on selective exposure to search engine content.

The left panel displays the hub page as the participants saw it. In the right panel, the stimulus is

overlaid with cursor tracking data of all participants, showing clusters of cursor positions on the

top-ranked search results.