Auto.Tabs: A Group-Based Machine-Learning Tab Manager

Victor Goncalves
Harvard University
Cambridge, Massachusetts, USA
victorgoncalves@college.harvard.edu

ABSTRACT

With an increasing usage of the internet for a variety of tasks and projects, internet browser users might open a numerous of tabs in order to access information. This can lead to having too many tabs open in a web browser, commonly known as "tab overload", which can lead to decreased productivity and negatively impact cognitive load. To address this issue, we created the Auto.Tabs Chrome extension, a group-based tab manager that automatically creates tab groups in a user's browser window. Auto.Tabs uses machine learning to compare tab titles based on sentence similarity, sorts tabs by alphabetical order, and encourages an iterative grouping process by leaving previously made groups untouched.

Author Keywords

tab management; tab overload; machine learning organization

INTRODUCTION

Tabs are a fundamental aspect to everyday web browsing, and they have been adopted by most browsers. Tabbed-browsing allows users to open and view multiple pages in a single web browser window in order to complete a variety of tasks in both their professional and personal lives.

Since tabs were introduced into the browser in the early 2000s, the usage of the internet has exploded–including both the amount of users and content currently on the internet [7]. Currently, there are over 4 billion web pages on the internet with about 60% of the world as users of the internet [9, 15]. Moreover, during the coronavirus pandemic in 2020-2021, people reported an increased usage of the internet as many shifted to remote work and online schooling [4]. Although internet usage has increased, little has changed with browser interfaces since tabs were introduced. Evidence suggests that current tabbed-browsing paradigms may not be sufficient for today's web browsing [7].

Previous works have explored the issues of tabbed-browsing, particularly revolving "tab overload", and have proposed systems to address these issues. In a 2019 study, about half (50.7%, N=75) of the participants "complained about tab clutter" [12]. There has also been evidence that tab overload can causes users to experience decreased task performance and increased cognitive load (cite here). Switching between many tabs can be difficult and could cause users to potentially lose important information. Furthermore, [7] explored the pressures, ranging from emotional to interaction costs, for why users might keep or close tabs. Using a two-week study (N=103), they created a user model for why tab management might be difficult for users today, which was later used by their task-centric tab manager: Tabs.do [8].

Some browsers, such as Chrome and Firefox, have introduced a tab group feature as a way to organize tabs within the browser by grouping them together. This feature allows users to create or close tab groups, name and color-code their groups, and also allows for easy switching between groups. This allows users to more easily manage and access multiple tabs at once, and can be useful for keeping related tabs together or organizing tabs by project or topic. However, these features require users to manually create groups. This can be time consuming, especially when users have a large (>25) amount of tabs.

Therefore, we created the Chrome extension Auto.Tabs, a group-based machine learning tab manager that automatically groups tabs together directly within the user's browser. Using Tensorflow.js, we compare the tab titles based on sentence similarity in order to create tab groups. Furthermore, our extension sorts tabs by alphabetical order upon grouping, and leaves previously made groups untouched. Lastly, the extension allows users to modify the 'similarity threshold' in order to make more accurate groupings depending on the types of tabs open in the browsers.

Using a 5-day user study (N=6), we show that participants generally liked the automatic grouping features Auto.Tabs. The extension served as a way to "start" their organization of tabs. However, results on whether Auto.Tabs alleviated tab overload was inconclusive. Generally, participants that use Chrome groups as their main method of browser organization found Auto.Tabs more useful than others.

RELATED WORK

Auto.Tabs builds on prior work related to tab organization and issues addressing tab overflow. Furthermore, we explore work on organization using machine learning, particularly with its connections to automated tab management.

CS279r, December 16, 2022

ISBN XXX-X-XXX-XXXX-X/XX/XX.

DOI: https://doi.org/XX.XXX/XXXXX.XXXXXXX

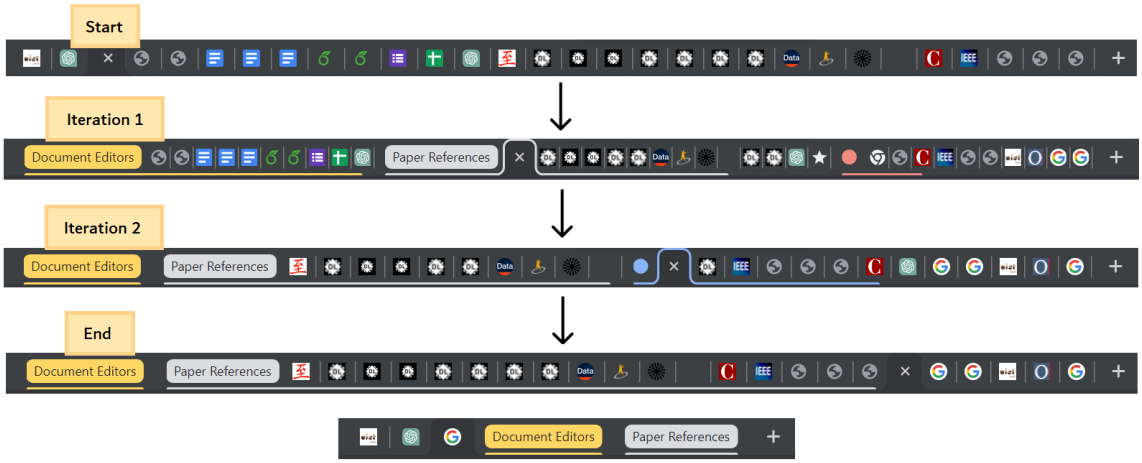

Figure 1. In this image, we demonstrate the iterative grouping process with Auto.Tabs. First, the user starts with a large number (>25) of disorganized tabs. The user can run our extension, providing a specific

Tabbed Web Browsing

Many tools have been established in order to improve current tabbed web browsing. Chrome existing built-in features such as creating tab groups, tab search by keyword or URL, tab previews, pinned tabs to prevent users from closing important tabs, and tab mute to mute noisy and unwanted tabs [11]. Many extensions on the Google Chrome also address tab organization. Workona is a tab manager that creates a page dedicated for a "work space" that allows users to switch between the tab groups they choose to display within the bar [2]. Snoozz snoozes tabs and windows until a preset time set by the user [5]. Group Tabs allows users to group tabs by URL and domain, sort alphabetically, and remove duplicated tabs [13]. One Tab takes all the tabs open in a user's browser and displays it on a single page. Currently, the extension has over 2,000,000 users [1].

In a previous work Sailboat, by Kulkarni et al., they create tab groups manually by task. Sailboat compartmentalizes history, bookmarks, downloads, and provides features to track time spent on various tasks in order for users to better fit their own groups [12].

Closest to our work is the Chrome extension, Tabs.do, that has addressed the issue of tab overload. Chang et al. use a task-centric approach to tab management to provide grouping suggestions [8]. However, Auto.Tabs differs from their approach as it focuses on managing tabs within the center of tab overflow, the tab bar, instead of on a separate web page.

Machine Learning Organization

Previous works have used machine learning in order to promote organization within tab management and the user's file systems. Again, closest to our work is Tabs.do, which uses machine learning in order to provide tab group suggestions. The extension provides suggestions based on tab tasks determined by behavioral and semantic features [8]. However, Auto.Tabs differs as it groups tabs automatically using strictly sentence similarity within tab titles.

The Chrome extension KondoCloud creates a file-browser interface for personal cloud storage. Using the data from their observation study, KondoCloud creates file interaction suggestions using machine learning based on the way users interacted with their file system [6]. For example, the system tracked different create, move, or delete actions in order to provide accurate recommendations. Auto.Tabs differs as it does not track different management actions, and automatically creates groups rather than providing suggestions.

DESIGN GOALS

When addressing tab overload, we wanted to provide a system that revolved around the usage of machine learning to group tabs. [3] provides several design guidelines on human-AI systems. In their work, they mention the following AI guidelines that we utilize in our design for Auto.Tabs:

- [G8] Support efficient dismissal.
- [G9] Support efficient correction.

Using these guidelines, we propose two following design goals.

Automatic Grouping

With Auto.Tabs, we wanted to address tab overload directly within the user's tab bar–the center of the overload itself. In order to provide groupings within the tab bar, Auto.Tabs automatically forces the groupings there, as opposed to providing suggestions in a separate web page. However, this inherently goes against the guideline in [3] on supporting efficient dismissal. We wanted to experiment with this feature and see how it affected the user study. With Chrome's built-in group features, they allow for easy switching of tabs between groups. We hypothesis that automated groupings will not negatively influence the user's experience as users can correct groups manually if a non-ideal paring was created.

Iterative Grouping Process

Within Auto.Tabs, we plan to emphasize an "iterative grouping process" when it comes to organizing a large set of tabs (see

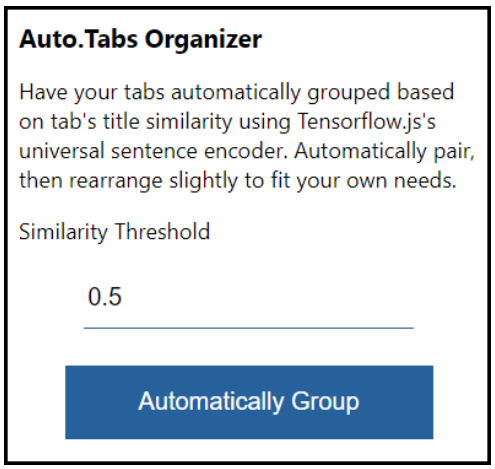

Figure 2. Auto.Tabs popup. Allows user to select a similarity threshold to group their tabs.

Figure 1). When grouping tabs by similarity, it is possible that some tabs might not necessarily fit a particular task or group, or that the machine learning might provide incorrect/non-ideal groups. Instead, we plan for Auto.Tabs to group larger, more obvious groups first, to start the organization process, then continue grouping until all tabs are organized. We plan on having an option to allow users to adjust "sentence similarity" of the groupings made by the tab titles. This adheres to the support efficient correction guideline as users can choose to make more specific or more general groupings depending on their set of tabs.

Furthermore, to emphasize the iterative group process, we plan to preserve previously made groups–allowing for users to build groups as they go along with their tasks. This is necessary to enforce some of the guidelines by [3] in terms of efficient correction/dismissal. Sometimes, the model might produce non-ideal groupings, and it would be inconvenient to accidentally break up a group up that was created by the user.

IMPLEMENTATION

Taking some of the guidelines discussed in the design goals, we implement the interface of Auto.Tabs as follows (see Figure 2). The user can click to group their tabs, and input a "similarity threshold" that they want to group their tabs by.

Sentence Similarity

Auto.Tabs uses machine learning to compare tabs based on sentence similarity. First, we retrieve the information about all the tabs within the user's browser using the Chrome API. Then, using Jinglescode's online example[1] for reference, we use Tensorflow.js to calculate sentence similarity. Using this model, we take the titles of these tabs and input it into the Universal Sentence Encoder. We receive a similarity matrix back, and group tabs based on a similarity threshold.

Within Tab Bar Organization

To produce within tab bar organization, we use the Chrome API to utilize the built-in grouping features. Once we have optimal groups of tabs found with the Universal Sentence Encoder model, we automatically group them within the user's tab bar. Also, we reorder and sort the tabs within the bar by URL upon grouping. This way, within the tab bar, websites with the same domain and icon will be adjacent to each other.

[1]https://jinglescode.github.io/textual-similarity-universal-sentence-encoder/

USER STUDY

To study the effect of automated tab groupings and the overall effect of Auto.Tabs, we conducted a user study.

For the recruitment process, we recruited 6 Harvard student participants using university mailing lists and by word-of-mouth (age range: 19-21; 2 male; 4 female; fields of study: 4 computer science, 1 neuroscience, 1 government). The criteria for the study was that all participants used either Chrome or Edge for their default browser–Auto.Tabs requires a Chromium-based browser, and did not want to force them to use an unfamiliar browser.

We sent out a pre-study survey with some demographic questions and some questions about how participants felt about their tabs. We asked whether participants whether they felt overwhelmed with their tabs, how they might organize their tabs currently, if at all, and some common tabs they might have open in their browser. Furthermore, we asked participants for their consent on collecting usage data during the study.

Following this survey, we conducted a quick demonstration of Auto.Tabs, including a test of the types of tabs it would group together and what happened to groups when the "sentence similarity" threshold was modified. We then asked participants to complete the two following within-subject tasks.

Task 1 Design

In the first task, we asked participants to organize the same set of 21 tabs in a browser using two sub-tasks: (control) their own personal method of organizing the browser, and (experiment) Auto.Tabs. Each participant was assigned to both conditions, and we randomized the order that these sub-tasks were assigned.

As all of the participants were Harvard college students, we created a generic set of tabs specific to the university that were recognizable by all of the students. The tabs that we selected included: Gmail, Google Calendar, Google Docs, my.Harvard (a portal for course selection, classes, and grades), the Q guide (review listings for classes), Canvas (a portal with page listings for all of the students' classes), a page listing for a Math class, Desmos (graphing calculator), YouTube, and Spotify. We allowed participants to explore the tabs before organizing in order to feel more comfortable with them.

For each sub-task we timed how long it took for the participant to feel content with the organization. We also took note on the types of organization methods used by each of the participants.

Task 2 Design

For the second task, we installed Auto.Tabs on the participants' browsers, and we asked them to interact with the extension on their own pace for 5 days. Using log data we tracked the participants' usage of Auto.Tabs. Every time the sentence similarity model was run, we logged the number of tabs and groups the

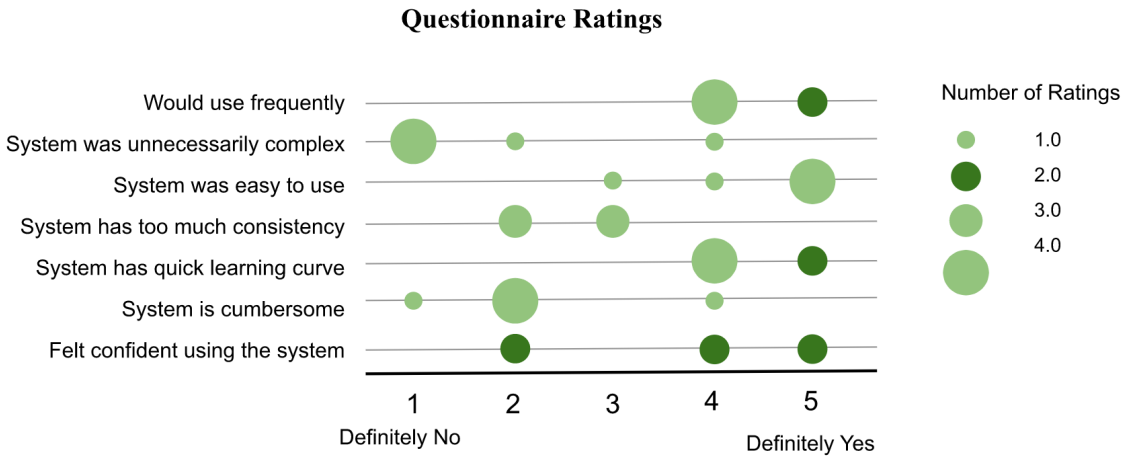

Figure 3. Post-Survey Auto.Tabs usability scale ratings

	Test Conditions (minutes)		
Participant	*Control*	*Experiment*	*Organization Method*
P1	1:42	1:07	Chrome groups
P2	1:44	1:39	Split windows
P3	1:37	1:32	Split windows
P4	3:07	1:40	Chrome groups
P5	1:12	2:05	Little organization
P6	2:11	1:10	Chrome groups

Table 1. Task 1 time results. Compared the time it takes for the participant to organize a browser with the same set of tabs using (control) personal method of organization and (experiment) Auto.Tabs.

Participant	*Days Used*	*Models Ran*	*Avg. Tabs Open*	*Groups Made*
P1	3	7	7	3
P2	4	5	13.4	10
P3	0	N/A	N/A	N/A
P4	4	9	31	14
P5	0	N/A	N/A	N/A
P6	0	N/A	N/A	N/A

Table 2. Task 2 usage study results. Tracked Auto.Tabs usage over a 5-day period. Results do not include initial run for extension installation.

participant had open in their browser. Also, we tracked the similarity threshold they used and the amount of groups they created. For privacy reasons, we did not collect any data on the types of tabs the participants had open, including URLs and tab titles.

At the end of the 5 days, we collected the data from the participants and asked them to complete a post-study survey. We designed our survey similar to the evaluation in [14]. On a 5-point Likert scale (1 for definitely no - 5 for definitely yes), participants answered the following prompts:

- Would use frequently
- System was unnecessarily complex
- System was easy to use
- System had too much inconsistency
- System has quick learning curve
- System is cumbersome
- Felt confident using the system

Lastly, we asked participants general questions about their thoughts on Auto.Tabs.

RESULTS

In the pre-survey, we studied the way that participants currently organize their browser and asked if they experience tab overload. Most participants felt overwhelmed by their tabs (5/6). (4/6) of the participants usually organize by class or topic of the task. Of these, (3/6) of the participants reported they use Chrome groups to organize their browser, while the other (1/6) splits their screen using windows organized by topic. The last (2/6) participants reported that they do not typically organize their browser, and both of these participants felt overwhelmed by their tabs. Furthermore, most of these organization techniques were reflected in each participant's approach to Task 1.

In the first task, we analyzed browser organization methods and compared the time it takes for participants to organize their browsers filled with the same set of tabs (see Table 1). Completion times varied greatly with the control task, which is reasonable as participants each had their own different method for organization. Experiment completion times also varied, but generally the faster times were from participants that used Chrome groups prior to the study (P1, P4, P6). For these participants, Auto.Tabs also sped up their organization time to create the groups. In terms of organization techniques, those that organized using Chrome groups created similar groups as Auto.Tabs. Across all participants, they created similar groups using Auto.Tabs.

In the second task, we tracked the usage of Auto.Tabs amongst participants (see Table 2). During this 5-day period, half (3/6) of the participants interacted with Auto.Tabs, and they did so on a near-daily basis. Amongst these participants, two of originally reported that they use Chrome groups as their main method of organization. Furthermore, Auto.Tabs usage seemed positively correlated with the number of average tabs opened in the browser increased, specifically with (P4).

Upon completion of the second task, we asked participants to fill out the post-survey (see Figure 3). All participants said that they use the system at least frequently. Most responded that the system was not unnecessarily complex and that the system was easy to use. Furthermore, participants were neutral about how inconsistent the system performed. This is mostly likely due to the inconsistency of the grouping with the sentence similarity model. This could have also attributed to the varied responses on how confident that participants were with the system.

Overall, the Auto.Tab extension received mostly positive feedback from participants in this study. Many participants appreciated its ability to assist with organization and decision-making, however some had concerns about the time it took for the extension to run and the difficulty in understanding the "sentence similarity" threshold, which is used to group similar tabs together. There was mixed feedback on the extension's handling of existing groups, with some participants appreciating that it did not re-sort already sorted tabs and others preferring the ability to add new tabs to existing groups. Some participants also suggested that the similarity threshold should be made clearer or abstracted away for easier understanding.

DISCUSSION

In this work, we introduced Auto.Tabs, a chrome extension that automatically organizes and groups tabs together using tab titles, sentence similarity, and machine learning. Through a pre-and-post survey and two tasks, we observed participants' current browser organization techniques and measured the effectiveness of Auto.Tabs in improving organization and decision-making.

Our results showed that Auto.Tabs was generally well-received by participants, with most reporting that the extension was easy to use and helped with organization. However, some participants had concerns about the time it took for the extension to run and the difficulty in understanding the "sentence similarity" threshold. These concerns could be addressed in future work by optimizing the extension's performance and providing clearer explanations of the sentence similarity threshold.

Auto.Tab's automatic and forced grouping features contradicts one of the Human-AI guidelines discussed in [3] due to potential non-ideal group pairings. However, this did not seem to be a concern with many of the participants. Although participants mentioned weird or non-ideal groupings, they were easily able to fix them, rearranging into the ideal pairing.

In terms of organization techniques, Auto.Tabs did not appear to perform faster when compared to participants' personal organization methods. However, participants in our study found the extension enjoyable to use as a starting point for organizing their tabs. Additionally, we found that Auto.Tabs was more suitable for participants who were already using Chrome groups for personal organization. In fact, participants who had previously used Chrome groups before the study were able to complete the organization process faster with Auto.Tabs than when using their own manual organization techniques.

In the second task, we observed that among the participants who interacted with Auto.Tabs, there seemed to be a positive correlation between their usage and the average number of tabs opened in the browser. This raises concerns about whether Auto.Tabs is actually effective at addressing tab overload. In fact, our findings suggest that it may actually encourage tab overload, as its usage seemed to be positively correlated with the number of average tabs opened. This is reasonable as we found that Auto.Tabs was able to produce more reasonable groupings with a larger number of tabs, based on our personal testing. While our extension creates sorted tab groups, we cannot guarantee that the built-in Chrome group feature assists in addressing tab overload.

In terms of future work, with the improvement of technology and machine learning, there is a clear way to improve Auto.Tabs with smarter groupings. Even when creating this work, a novel intelligent machine system was released to the public, GPT-3.5 and ChatGPT [10]. The system has intelligent natural-language processing capabilities, and when given a list of tab sentences, made identical or better groupings than Auto.Tabs–each with one or two word summaries–that could be used to create a Chrome group. We could recreate Auto.Tabs using this software if we were to approach the project again.

With the limitations of our work, we had a small sample size, where all participants were recruited from the same university and the same resident building mailing list. This could potentially introduce response biases–we live in the same undergraduate house. For example, they might not want to rate our system poorly if they know who created it. Furthermore, given more time to work on the project, we could have improved the functionality of Auto.Tabs to make it more usable than it is currently. This was also our first introduction with human-computer interaction, so there are definitely ways to improve the system and study designs.

Overall, our results suggest that Auto.Tabs could a useful tool for improving browser organization and decision-making in the tabbed web browsing interface. We cannot guarantee that Auto.Tabs lessens tab overload with our study. Further studies with a larger sample size and longer usage periods would be needed to fully evaluate its effectiveness and identify potential areas for improvement.

Conclusion

Our work presents a group-based approach to browser tab management through the Auto.Tabs Chrome extension. By using machine learning to compare tab titles based on sentence similarity and encouraging an iterative grouping process, Auto.Tabs aims to improve the tab management experience for internet users and reduce tab overload. The results of our

user study suggest that the extension is well-received by participants and serves as a useful starting point for organizing tabs. However, some concerns were raised in terms of the time it took the model to run, difficulty in understanding the "sentence similarity" threshold, and how well the model could create groupings. These concerns could be addressed in future work as improvements are made in our design and machine learning and natural-language processing. The release of the novel GPT-3.5 system offers more accurate groupings and improved machine intuition, enabling users to explain how they would like their browser organized [10]. Further research is needed to fully understand the long-term effects of using the Auto.Tabs extension on productivity and cognitive load, but it represents a promising step towards addressing the challenges of tabbed-browsing and supporting modern online tasks in the browser.

ACKNOWLEDGMENTS

I am grateful for the unwavering support and guidance for professor, Elena Glassman, and teaching fellow, Priyan Vaithilingam. The opportunity to investigate different aspects of human-computer interaction and to produce a system of my own was remarkably fulfilling. It would have been impossible without them.

REFERENCES

[1] 2017. OneTab extension for google chrome and firefox - save up to 95 memory and reduce tab clutter. (2017). https://www.one-tab.com/

[2] 2022. The work organizer for the browser. (2022). https://workona.com/

[3] Saleema Amershi, Kori Inkpen, Jaime Teevan, Ruth Kikin-Gil, Eric Horvitz, Dan Weld, Mihaela Vorvoreanu, Adam Fourney, Besmira Nushi, Penny Collisson, Jina Suh, Shamsi Iqbal, and Paul N. Bennett. 2019. Guidelines for Human-AI Interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19*. ACM Press, 1–13. DOI: http://dx.doi.org/10.1145/3290605.3300233

[4] Sara Atske. 2022. The internet and the pandemic. (Apr 2022). https://www.pewresearch.org/internet/2021/09/01/the-internet-and-the-pandemic/

[5] Rohan Bhansali. 2021. Tab amp; Window snoozing. (Dec 2021). https://snoozz.me/

[6] Will Brackenbury, Andrew McNutt, Kyle Chard, Aaron Elmore, and Blase Ur. 2021. KondoCloud: Improving Information Management in Cloud Storage via Recommendations Based on File Similarity. In *The 34th Annual ACM Symposium on User Interface Software and Technology (UIST '21)*. Association for Computing Machinery, New York, NY, USA, 69–83. DOI: http://dx.doi.org/10.1145/3472749.3474736

[7] Joseph Chee Chang, Nathan Hahn, Yongsung Kim, Julina Coupland, Bradley Breneisen, Hannah S Kim, John Hwong, and Aniket Kittur. 2021a. When the Tab Comes Due:Challenges in the Cost Structure of Browser Tab Usage. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 148, 15 pages. DOI: http://dx.doi.org/10.1145/3411764.3445585

[8] Joseph Chee Chang, Yongsung Kim, Victor Miller, Michael Xieyang Liu, Brad Myers, and Aniket Kittur. 2021b. Tabs.do: Task-Centric Browser Tab Management. (2021).

[9] Maurice de Kunder. 2016. The size of the World Wide Web (the internet). (Feb 2016). https://www.worldwidewebsize.com/

[10] Thilo Hagendorff, Sarah Fabi, and Michal Kosinski. 2022. Machine intuition: Uncovering human-like intuitive decision-making in GPT-3.5. (2022). DOI: http://dx.doi.org/10.48550/ARXIV.2212.05206

[11] Edward Jung. 2020. Keep tabs on your tabs in Google Chrome. (May 2020). https://blog.google/products/chrome/manage-tabs-with-google-chrome/

[12] Minchu Kulkarni, Kshitij Kapoor, Deva Surya Vivek Madala, Sanchit Bansal, and Sudheendra Hangal. 2019. Compartmentalizing Web Browsing with Sailboat. In *Proceedings of the 10th Indian Conference on Human-Computer Interaction (IndiaHCI '19)*. Association for Computing Machinery, New York, NY, USA, Article 9, 8 pages. DOI: http://dx.doi.org/10.1145/3364183.3364187

[13] michimani. 2021. Group tabs. (2021). https://chrome.google.com/webstore/detail/group-tabs/cnmcnafaccboidemenkpfnlgfcejijgm

[14] Luke S. Murray, Divya Gopinath, Monica Agrawal, Steven Horng, David A. Sontag, and David R. Karger. 2021. MedKnowts: Unified Documentation and Information Retrieval for Electronic Health Records. *CoRR* abs/2109.11451 (2021). https://arxiv.org/abs/2109.11451

[15] Max Roser, Hannah Ritchie, and Esteban Ortiz-Ospina. 2015. Internet. (Jul 2015). https://ourworldindata.org/internet