Enlightened:
Can short-form news videos open minds?

by

Mike Hao Jiang

M.Sc. ETH Electrical Engineering and Information Technology,
Swiss Federal Institute of Technology Zurich

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
in partial fulfillment of the requirements for the degree of

Master of Science in Media Arts and Sciences

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2021

Author .....
Program in Media Arts and Sciences
May 21, 2021

Certified by .....
Andrew Lippman, Ph.D.
Senior Research Scientist and Associate Director of the MIT Media Lab
Thesis Supervisor

Accepted by .....
Tod Machover, Ph.D.
Muriel R. Cooper Professor of Music and Media
Academic Head, Program in Media Arts and Sciences

Enlightened:

Can short-form news videos open minds?

by

Mike Hao Jiang

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
on May 21, 2021, in partial fulfillment of the
requirements for the degree of
Master of Science in Media Arts and Sciences

Abstract

The United States of America has become severely polarized over the last twenty years, coincident with the increase in niche and fringe media. This contributes to the fragmentation of the shared assumptions, beliefs and trust in information that comprises ones perception of reality. Recently, the short-form video format has gained massive popularity in the world of social media and mobile applications (e.g. TikTok). To investigate whether this media format can be used to restore the shared reality among U.S. liberals and conservatives, I built a mobile-first progressive web application Enlightened that presents short, swipeable news videos in a manner similar to the popular dating app Tinder. The news clips were sourced from five major TV networks across the ideological spectrum (MSNBC, CNN, Bloomberg TV, ABC News and Fox News) and processed by a real-time news recording and processing system SuperGlue, to which I have also contributed to. The processed news videos were summarized using a variation of the TextRank algorithm on the closed captions and the news source was visually masked by removing the lower third of the video using FFmpeg. Although the current interface of Enlightened has limited features, the results of a user study consisting of two surveys and the daily usage of Enlightened suggest that masked short-form news videos show great promise in opening the minds of both conservative and liberal users. However, the biggest limitation of this thesis is the small size of the user study. Hence, a larger-scale test needs to be conducted to ascertain whether short-form news videos can open minds.

Thesis Supervisor: Andrew Lippman, Ph.D.
Title: Senior Research Scientist and Associate Director of the MIT Media Lab

This thesis has been reviewed and approved by the following committee member:

Andrew Lippman, Ph.D. . . . . .
Senior Research Scientist and Associate Director
MIT Media Lab

This thesis has been reviewed and approved by the following committee member:

Cynthia Breazeal, Ph.D. . . . . .
Professor of Media Arts and Sciences and Associate Director
MIT Media Lab

This thesis has been reviewed and approved by the following committee member:

David Rand, Ph.D. . . . . .
Associate Professor of Management Science and
Brain and Cognitive Sciences
MIT Sloan

Acknowledgments

I want to thank my advisor Andy for giving me opportunity to work on such an interesting topic and relevant problem, for expanding my horizon and curing me from my "engineering disease", for challenging my thought processes and approaches, for inspiring me to think bigger, for encouraging me to be fearless, and for sharing his wisdom. I have grown significantly thanks to his guidance and support, and I look forward to continuing to learn and grow under his mentorship.

I want to thank my thesis committee - Andy, Dave and Cynthia - for critiquing my thesis proposal, offering resources and reading my thesis. On a similar note, I want to thank my SHASS writing group - Gabrielle, Rafael and Queenie - for reading parts of my thesis, critiquing my style and suggesting improvements.

I want to thank my Viral Communications family for being an extremely positive and supportive environment throughout my years at MIT. I want to thank Hisham for sharing the enthusiasm for mixed reality, for bouncing around ideas, for discussing the future of this technology with me as well as for being an inspiration and a great friend. I want to thank David for being a great collaborator on SuperGlue and friend. Speaking of SuperGlue, I want to thank Tomer, Jasmin and the Ultimate Media program for laying the foundation on which this thesis was built on. I want to thank Agnes, Britney, Kalli, Nchinda and Travis for intimidating me with their broad technical and design skills and for pushing me to work harder on myself. I want to thank Aruna, Dan, JS, Sam and Océane for being a positive and joyful source of energy that helped to keep me going when things got rough. I want to thank Deborah and Alessandra for their hard work behind the scenes and enabling me to focus on my research. Similarly, I want to thank Keira, Amanda, Linda, Sarra, Lily, Mahnaz, Rebecca, Monica and Tod for their support throughout my time at the Media Lab.

I want to thank my UROPs Johan, Helen, Sophia, Yodahe, Dagmawi, Azariah for their contributions to Enlightened. I want to highlight the contributions of Johan, who has helped with SuperGlue, Enlightened and experiment data evaluation. Outside of this thesis, I want to thank my UROPs Brian, Demar, Gohar, Nicholas, Sophia,

Hiya on their work on Defacto, and Adit, Kevin, Veerapatr, Cindy, Jessica, Thatcher, Sebastian, Thomas and Isabella, Preston and visiting students Veronika and Tejal for their work on SuperGlue modules.

I want to thank my Media Lab cohort and friends for being a source of inspiration and aspiration and I will miss those who have already left or are leaving this year. I especially appreciate the time I spent with Ravi, Teja, Gabriela, Rubez, Yusuf, Ramon, Junsu, David, Vik and wish them a great future outside the Media Lab. And to those that decided to stay, including Erik, Nikhil, Noah, Pat, Alex, Patrick, Manuj, Joanne, Hila, Brigit, Abhi, Irmandy, Manaswi and Don, I'm looking forward to continuing to spend time with you all post-pandemic!

I want to thank my fellow MIT jazz cats Eric, Shravan, Josh, Dennis, Nick, Steven, Zoe, David, Erik and Nikhil for enriching my experience at MIT musically. A special shoutout to coach Keala, director Fred and my piano teacher Doug for supporting my musical endeavors and growth.

I want to thank my friends at the International Fellowship House (Joe and Anna, David, Emilio, Diego, Gian, Uwe, Tibor, Lukas, Sepehr, Ali, Yuki, Sathiji and Emmanuel) for their support throughout the years. Especially David, who has also been challenging me physically, for following through the David Goggins 4x4x48 challenge together, and for pushing me to qualify for the Boston marathon next year. Furthermore, I want to thank my friends Barbara, Isaiah, Olivia, Jonathan, Shovan, Drew, Raquel, Megan, Aaron, Jihye, Indrayud, Karen, Piyush, Alan, Jen, Alex, Felix, Ming, Dino, Swe, Anna, Sophie, Lisa, Gloria and Mengzhen for their moral support, for being a source of inspiration and for pushing me to become a better version of myself.

Furthermore, I want to thank Araceli, Agustin and Ulrich for their empathy, support and wisdom, and for improving my life significantly.

Lastly, I want to thank my parents for all the sacrifices they made and their unconditional love and support, providing the foundation on which I could grow. To my late mother Yanyan, may your soul rest in peace and I hope that I have been and will continue to make you happy and proud. To my father Bin, thank you for instilling the value of education, discipline, hard work and time in me.

Contents

List of Figures

Chapter 1

Introduction

1.1 A Polarized United States of America

In the paper "They Saw A Game" (1954), Hastorf and Cantril showed that although the students from two rival colleges saw the same motion picture of a rough football game, they disagreed on what had transpired during the game [14]. Furthermore, each interpretation of the game was just as "real" to one person as other interpretations were to other people. However, this was not pervasive in society, even if true for sports. For example, Walter Cronkite, the newscaster famous for ending his news segments with "and that's the way it is", was voted as America's most trusted man [23], and most people believed what he reported.

The gradual deregulation of cable television in the 1970s and 1980s (e.g. alterations on syndication exclusivity, the Cable Act of 1984) stimulated massive growth of cable services [4, 2]. Not only did more than 50 million households subscribe to cable by the end of 1989, but it also enabled the popularization of superstations such as Turner's SuperStation WTBS (later only known as TBS). This diluted the reach[1] of existing networks and made it possible to profit from a niche audience. Furthermore, the abolishment of the Fairness Doctrine[2] in 1987 enabled licensees to select

[1] Reach is a broadcasting term referring to the total number of people or households exposed, at least once, to a medium during a given period. The exact measure differs from medium to medium (e.g. radio vs. television) and geographically.

[2] A policy of the Federal Communications Commission (FCC) requiring broadcast licensees to

the profit-maximizing quantity of informational programming, as has been observed in the radio market in the 90s [15]. Before that, each network vied for the whole audience and were therefore all similar[3].

These developments enabled fringe media to emerge and thrive. Rupert Murdoch, who had previously exploited fringe markets with the Sun newspaper in Britain, created Fox News on a similar basis. Ultimately, Fox News started creating and cultivating a unique perspective for their audience, and others have followed, including Rush Limbaugh and Sean Hannity with their talk radio shows. As of 2020, Fox News, Hannity (radio) and Limbaugh (radio) are the most trusted news sources for conservatives [21]. It should be mentioned that this phenomenon is not restricted to the conservative political spectrum. MSNBC, a liberal cable network that emerged in the mid-90s, has used a similar formula to become successful. In 2012, former president Bill Clinton said this about MSNBC: "Boy, it really has become our version of Fox." [26]. In December 2020, MSNBC had the most viewers per day, closely followed by Fox News and CNN [30]. The partisan dedication evident in sports is now extended to print and broadcast news.

The rise of the internet and social media in the last two decades has removed virtually all friction from publishing and given everybody a voice. Everyone can now become a channel, even fake agents such as Twitter bots or troll farms. Furthermore, social media recommendation systems and the commoditization of user attention have created echo chambers, eroding the sense of a shared reality on a much larger scale. However, blaming technology and social media for the polarization and tribalism of our societies is a trivialization of the problem. Take the internet for example: in a demographical study on polarization, Boxell, Gentzkow and Shapiro showed that the internet is not a primary driver of rising political polarization among US adults [5]. And if technology and social media were to blame, we could just fix it.

Society itself sometimes has reasons to be divided, polarized and tribal. In “The Last Time Democracy Died”, Lepore illustrates the challenges of the American society

cover issues of public importance and to do so in a fair manner [27].

[3]This observation has been explained by Harold Hotelling in 1929 in his article "Stability in Competition" [17].

in the 1930s amidst rising international fascism [22]: “American democracy, too, staggered, weakened by corruption, monopoly, apathy, inequality, political violence, hucksterism, racial injustice, unemployment, even starvation.” The current American society is also plagued by similar and different problems, including the lack of access to affordable healthcare, the threat of the coronavirus, an opioid crisis, a mental health and suicide crisis, the threat of automation and dawn of the fourth industrial revolution, exorbitant tuition fees, climate change and more.

In "The Increasingly United States", Hopkins makes the point that nationalized political behavior is a critical ingredient in today's political polarization [16]. He elaborates how and why American political behavior nationalized. Politics has become a lot less local than they used to be, and this development has been a long-term trend of nationalizing forces. Changes in national partisan politics along ideological and identity lines have been adopted to state politics. Correspondingly, people have become less knowledgeable about local politics. This can be attributed to both changes in party politics as well as the nationalization of media. Local news has been eroding, both in print and on television, contributing to the loss of a sense of shared reality in local communities. Hopkins attributes these changes in the media environment to the cause of declining state and local political engagement, using a broad range of survey data as well as looking at the historical development of TV's designated market areas.

Bottom line, the United States of America has become severely polarized since 2004. The Pew Research Center has shown that in 2014, 40% of Democrats see the Republican Party as a threat to the nation's welfare and vice-versa [7]. Surveys in 2017 indicated that the election of President Donald Trump has driven the median of the American public's political values amongst Democrats and Republicans even further apart [9]. The polarization is also reflected in the distribution of trust in the media ecosystem [8, 25]. For example, the Fox News Network is highly trusted among conservatives, but it does not find much ground among liberals. The division does not end there - a publication by the Wall Street Journal illustrates the economic division of the two main political parties [11]. Partisans have become more isolated from each

other in their real and virtual lives, exacerbating polarization [13]. As of January 2020, there is no end of polarization in sight, according to the Pew Research Center [21]. The 2021 storming of the U.S. Capitol by a mob of Trump supporters was a violent manifestation of the dangers of polarization. And even with a new president at the helm, it remains to be seen whether his efforts for unification bear fruit. And even if there will be progress, it would probably take some time to manifest itself.

Historically, American society was able to recover and move on from periods of polarization such as the 1860s (American Civil War) and the 1960s (during the Vietnam War). However, the current polarization takes place in a digital society where the media landscape (e.g. new players such as Fox News, MSNBC; fringe media such as Breitbart or Shareblue; fake news, bots and troll farms), media technology (e.g. the internet, mobile applications, social media), political (e.g. nationalized political behavior) and economic (e.g. globalization) context have all changed, and there's no turning back: it is hard to imagine a society without computers or mobile phones, or enjoying cheap goods that are not made in China; we can't pull the plug on Fox News, MSNBC, Breitbart or Shareblue either, and policies such as the Fairness Doctrine are unlikely to be reestablished in the near future.

Unlike the era of Walter Cronkite, we now live in a polarized society where the media landscape is also polarized and fragmented. Furthermore, as "They Saw A Game" illustrated, even with a common reference of reality, the perceived reality can differ significantly. It is unclear whether polarization in society causes polarization in media or vice-versa. However, we might be able to address the fragmentation by developing media that exposes people with different realities to a superset of realities, even with the knowledge that the perceived reality might ultimately differ.

1.2 The Rise of Short-Form Video

Short-form video might be a useful tool that has gained massive popularity recently. In the early to mid-2010s, Vine became the most popular and most utilized video sharing app in the market. Vine was a six-second looping video app and had over 200

million monthly active users at its peak. Twitter, a micro-blogging and social media platform, which acquired Vine in 2012, shut down Vine to cut costs in 2016, much to Vine users' disappointment [6].

Instagram, an app known for sharing polished images and videos, initially only allowed users to share videos up to one minute. For longer videos, they created IGTV. However, only 7 million of 1 billion-plus Instagram's users downloaded its standalone app in the 18 months since launch in June 2018 [10].

That number pales in comparison to the number of downloads TikTok had during the same period in the US: 80 Million. TikTok is a short-form video creation and sharing app that has recently taken over the world by storm. As of October 2020, Tiktok has surpassed over 2 billion mobile downloads worldwide.

Established entertainment and social media companies have responded to TikTok's success with their own short-form video apps. For example, Instagram launched Instagram Reels in August 2020 [18] and YouTube introduced YouTube Shorts in September 2020 [19], both an imitation of TikTok's interface.

Now, can we apply the short-form video format for news? One observation about short-form video is the fact that it is in stark contrast with traditional TV news programs' long-form video format. Short-form video enables viewers to get informed in a shorter time, especially if the content is summarized. Consequently, short-form video enables the juxtaposition of perspectives and topics without forcing viewers to watch the whole news segment. This property is especially valuable in a world where the attention span of an internet user has been commoditized and become short.

1.3 Restoring Common Ground with Short Videos

The nationalization of media, erosion of local news, low-friction publishing, fake news, partisan politics, commoditization of attention, and echo chambers have all contributed to the erosion of the shared assumptions, beliefs and trust in information that comprises ones perception of reality, all of which I subsequently refer to as *common ground*.

The goal of this thesis is to investigate whether the short-form video format has the potential to both inform as well as to expand our perspective. If the research indicates that *brevity* can open minds, the findings can be used in the efforts of restoring the *common ground*.

My hypothesis is that fragmentation and the associated tribal interpretation of news are in part driven by social factors and norms, and in part because the same interface and channel that opened up media participation for all contributes to narrowing our perspective. To address both the social and interface component of the hypothesis and to learn whether short-form news videos can open minds, I designed and built a multimedia application *Enlightened* that makes bursting your bubble easier than staying in it.

Enlightened presents short broadcast television news video clips to users without telling them where the clips are from, i.e. the news source is masked. Users can like or dislike what they see, and the application records their liking/disliking behavior. It also records how much time they spend on a particular video and knows what video it is, where it's from (e.g. Fox News, CNN, ...), and whether other users had a similar reaction, in case they watched the same video.

By visualizing and analyzing *Enlightened* usage data of beta-testers along with a 14-day user study comprising two surveys, I aim to understand whether short-form news videos can captivate people and open their minds. The metrics used to answer these research questions are elaborated in the experimentation and results section.

The remainder of this document is structured into four parts. The design section (chapter 2) presents related work aiming to open people's minds and elaborates on the design decisions and iterations made in the process. The implementation section (chapter 3) illustrates the software architecture of *Enlightened* comprising of a real-time video recording and processing engine, a recommendation system and a mobile-friendly progressive web application. The experimentation and results section (chapter 4) breaks down the research questions, shows the metrics used to answer the research questions in detail, and exhibits the results. The discussion section (chapter 5) reviews the results, reflects on ethical implications and outlines future work.

Chapter 2

Design

This chapter introduces a selection of related works addressing the issue of polarization by helping people to broaden their perspectives. Second, it motivates a mobile-first design and exhibits both a selection of contemporary mobile U.S. news apps as well as a selection of mobile non-news apps with inspiring design elements. Lastly, it illustrates and discusses the design of the user interface of *Enlightened*.

2.1 Work Related to Opening People's Minds

2.1.1 Allsides.com

The website and mobile app Allsides.com aim to help people to broaden their views by presenting users with articles across the political spectrum. They do it by rating hundreds of news outlets and writers on their bias using their patented methodology. Media articles are rated into five categories (far-left, lean-left, center, lean-right, far-right). Their mission is very close to the objective of this thesis, but only covers the text medium.

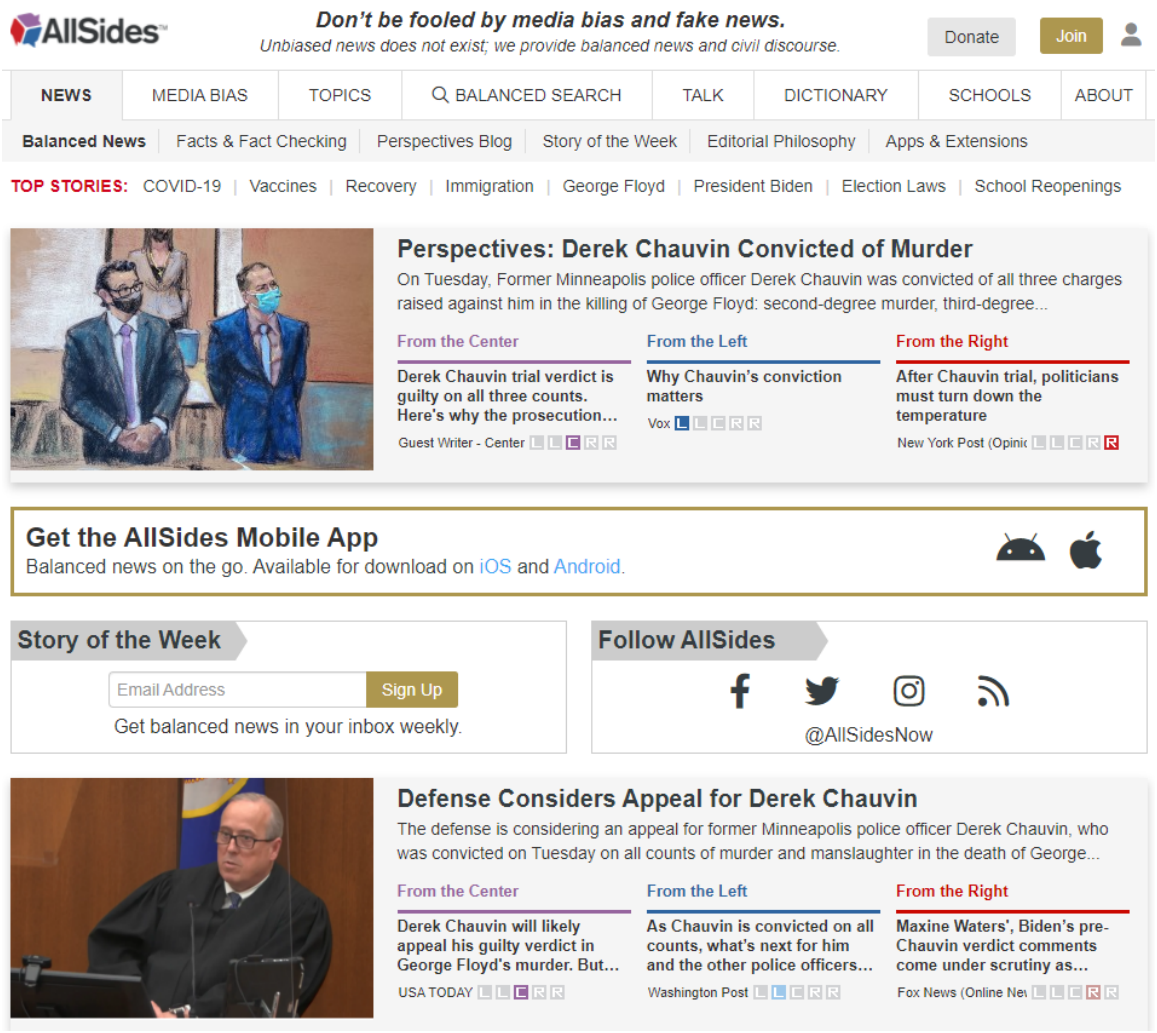

Figure 2-1: The Allsides.com website presents three viewpoints to each topic, each rated into five categories.

2.1.2 Panorama

Panorama [28], an interface developed by Jasmin Rubinovitz, was aimed to burst echo chambers. It is an open, transparent, and collaborative news exploration platform that presented different perspectives and encouraged serendipity in news exploration. It is an open-box news aggregator, where the user can decide to view everything or to filter out content. Furthermore, the user can suggest labels to news stories to train the machine learning process of classifying news by left/right, positive/negative, objective/subjective and trending/ongoing. This interface is a great exploration tool for text-based news and exposes users to a variety of sources. But just like many

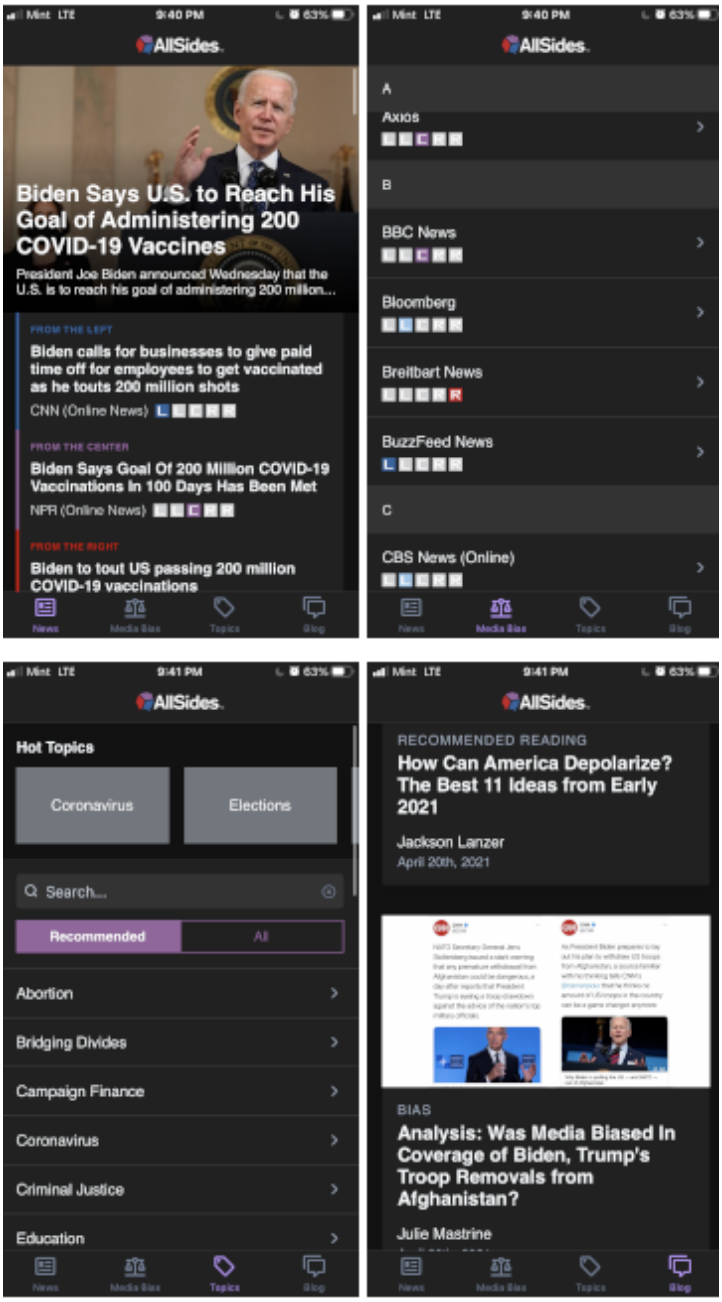

Figure 2-2: The Allsides mobile app has four main screens. The news screen (top left) presents three viewpoints to each topic, similarly to the website. The media bias screen (top right) presents the rating of a news outlet (far-left, lean-left, center, lean-right, far-right). The topic screen allows a user to search for specific news. The blog section recommends a set of readings.

news aggregation platforms or apps, by giving users the freedom of choice, it does not guarantee exposure to content outside the user's information bubble, especially if the user is not interested in perspectives different than their own.

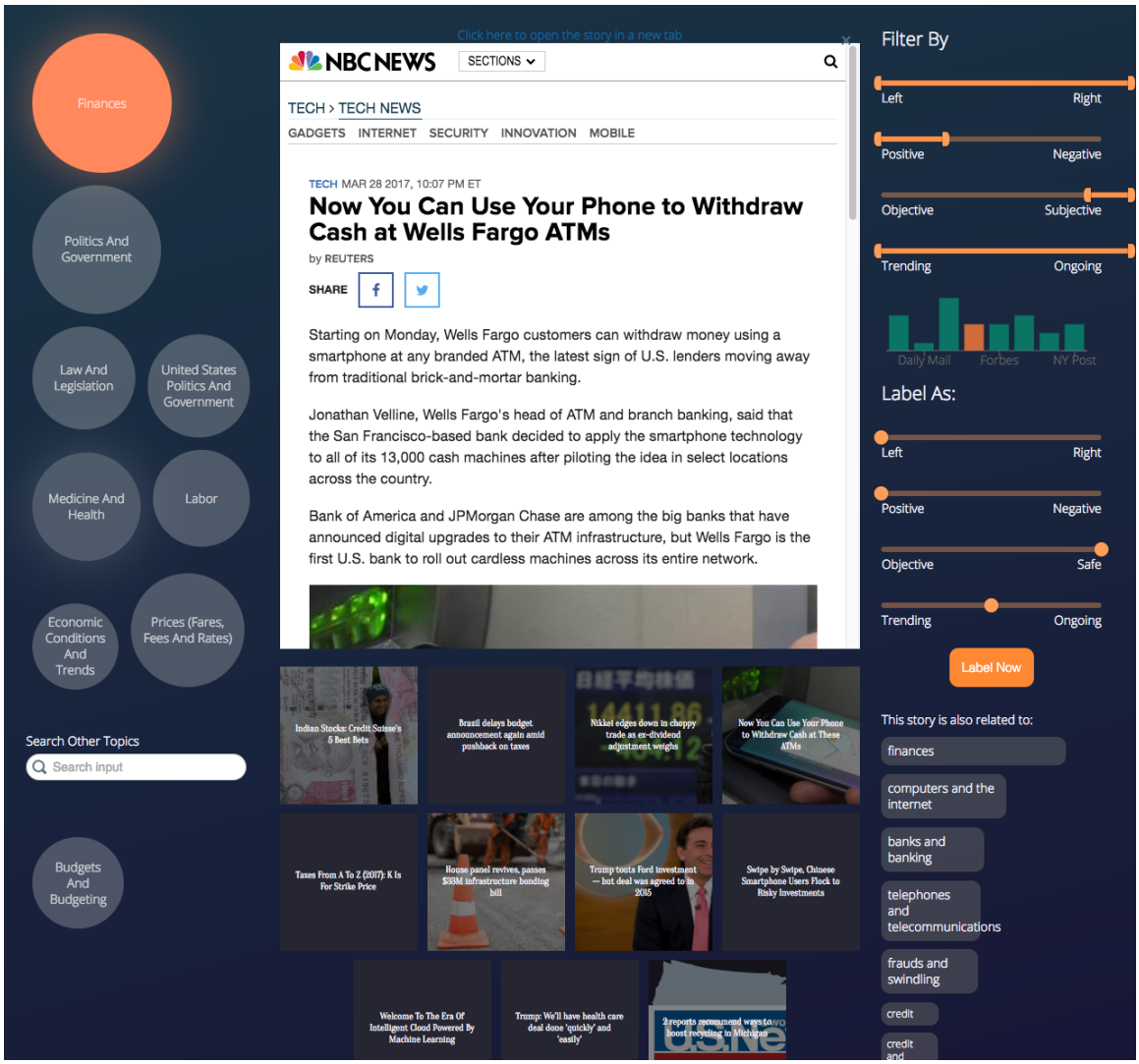

Figure 2-3: Panorama is an open, transparent, and collaborative news exploration platform that presents different perspectives and encouraged serendipity in news exploration

2.1.3 Gobo

Gobo (URL: gobo.social) is an experiment built by the former MIT Media Lab's Center for Civic Media, which Jasmin Rubinovitz has also contributed to. The goal of this experiment is to allow the user to control the algorithms that control our social media feeds. It aims to hold social media accountable for algorithmic bias and reveals the hidden posts as a result of the rules you set. Gobo also allows you to add unfamiliar perspectives into your feed using a set of filters, so that you can better

understand the range of opinions shared online. Similar to Panorama, Gobo primarily focuses on text-based media and social media algorithms (feed curation and process awareness) and the content distribution online. Gobo allows their users to link their Facebook and Twitter account.

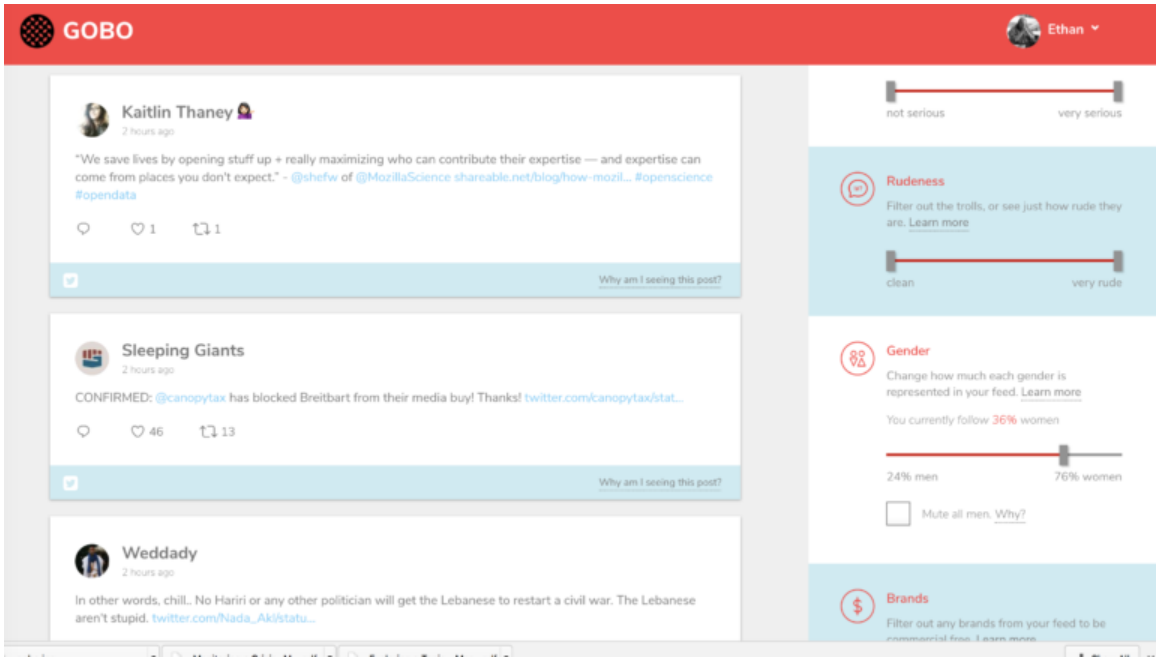

Figure 2-4: Similar to Panorama, Gobo enables the user to set a filter across different dimensions, such as seriousness (the ratio of serious news to fun stuff in your feed), rudeness (the spiciness of trolling), gender (the ratio of gender representation), brands (hiding posts from brands to make the feed commercial-free), obscurity (showing posts that do not get much love), and topical filters such as Covid or the 2020 Election. The image depicted represents an earlier version of Gobo.

2.1.4 NewsMaker

NewsMaker is a web app that prompts users to create their own favorite story out of a set of news snippets and to reflect upon their own judgment on their news sources and their content. In David Anderton-Yang's Master Thesis "Countering Source Bias in News", 67 participants across the political spectrum indicated their trust towards a set of media outlets and were instructed to create their most objective story from a set of snippets sourced from real news articles around a given topic. While 90% of the people indicated that they believed in creating an objective article, left leaning

participants predominantly chose snippets from sources they indicated they would not trust. This rather surprising result prompted some reactions, such as "I guess Breitbart isn't as whacko as I thought it was.", "Given how liberal I am, I am surprised that the sources I used were from relatively conservative sources. My first reaction was - How did I do this and how did this happen?", "... maybe news sources are not all that different from each other. That's a bit interesting to think about." [1]. The results of Anderton-Yang's experiment suggest that there is a disconnect between the sources we trust and the content we believe to be objective. However, the cause of this phenomenon has not been explored yet. David and I digitized the experiment in November 2019 and the final interface is illustrated in the Figures below.

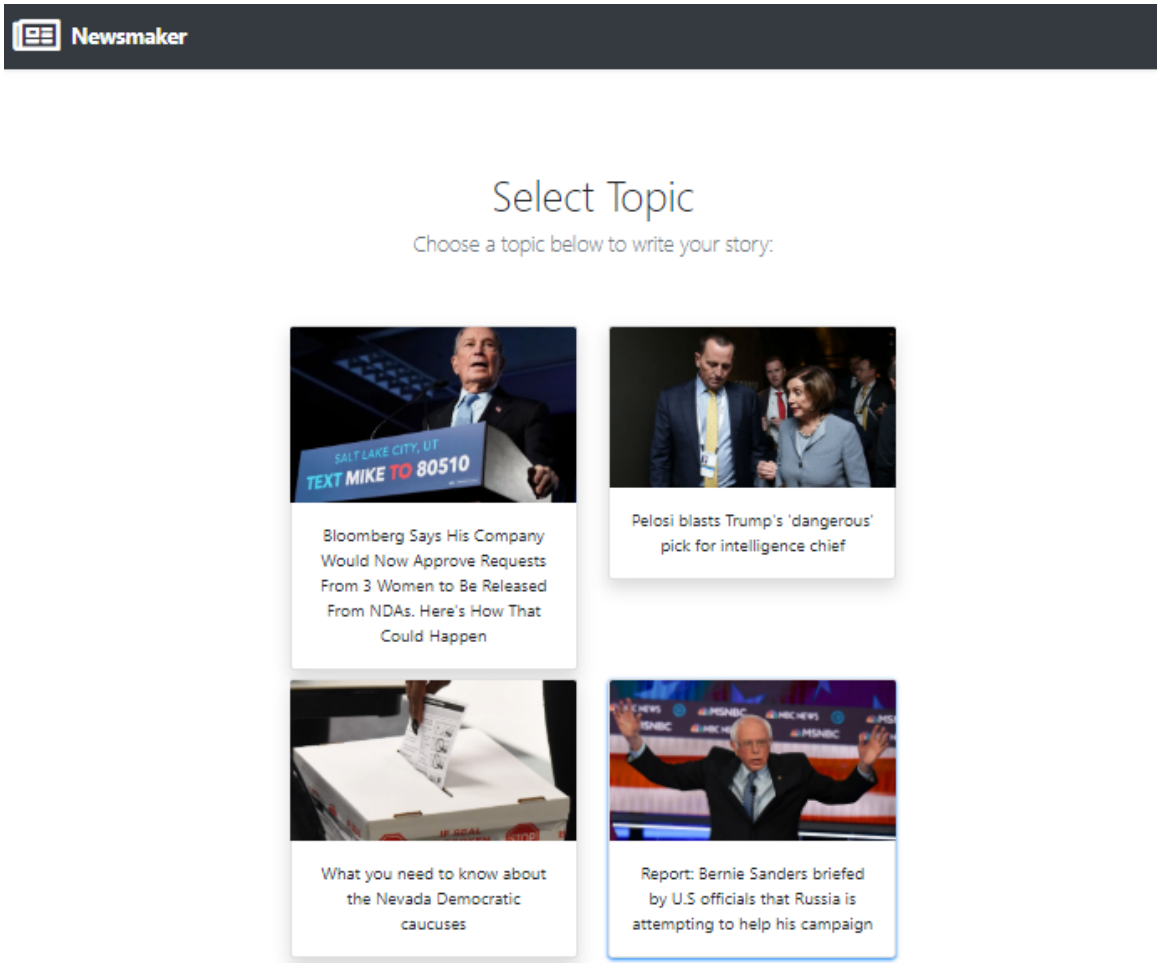

Figure 2-5: The user is prompted to select a topic to create their story on.

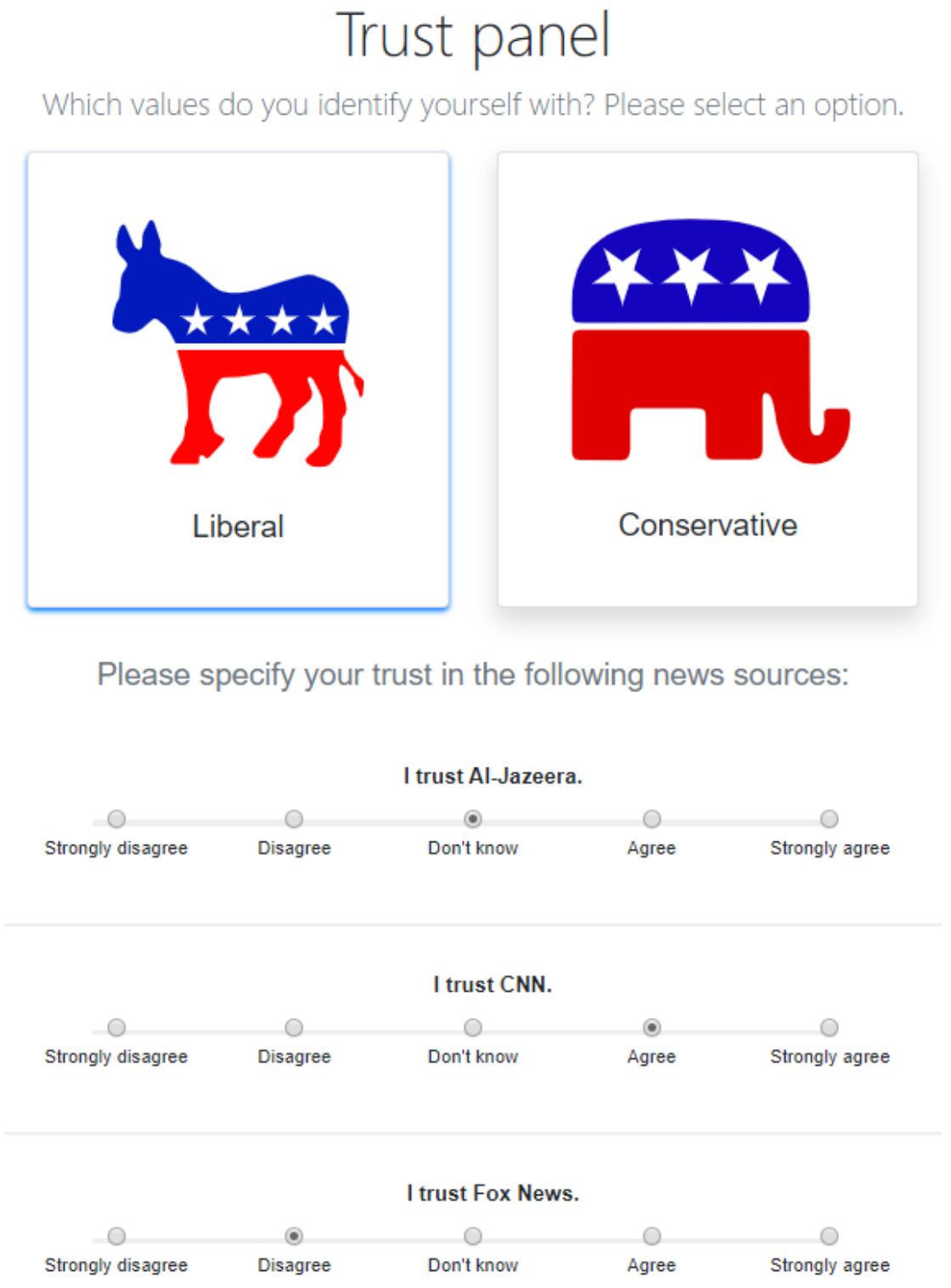

Figure 2-6: Every new user has to indicate their trust in a given set of news outlets. For the purpose of this illustration, the number of sources shown have been reduced.

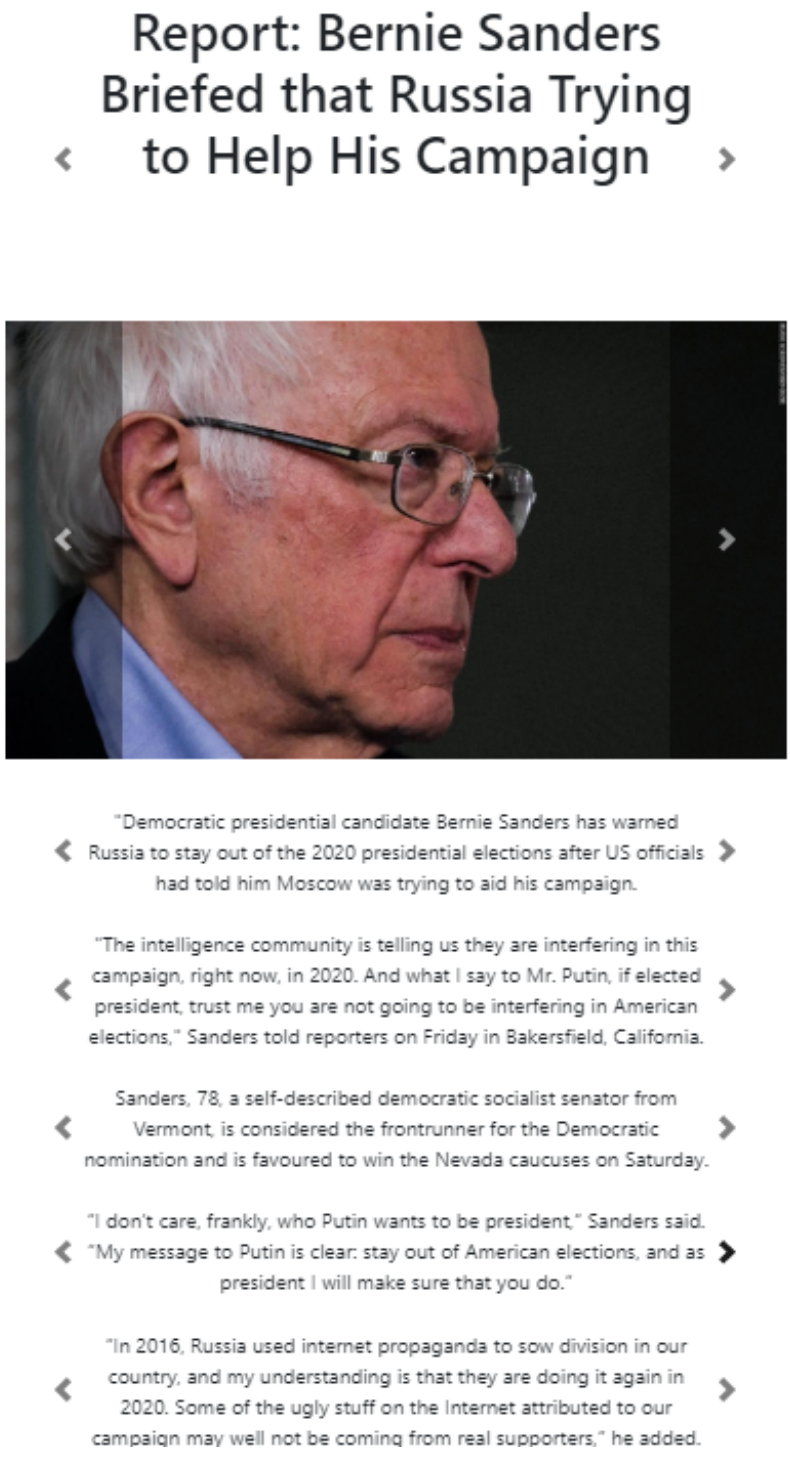

Figure 2-7: The user selects the headline, thumbnail image, and the story snippets that would make their most objective news story.

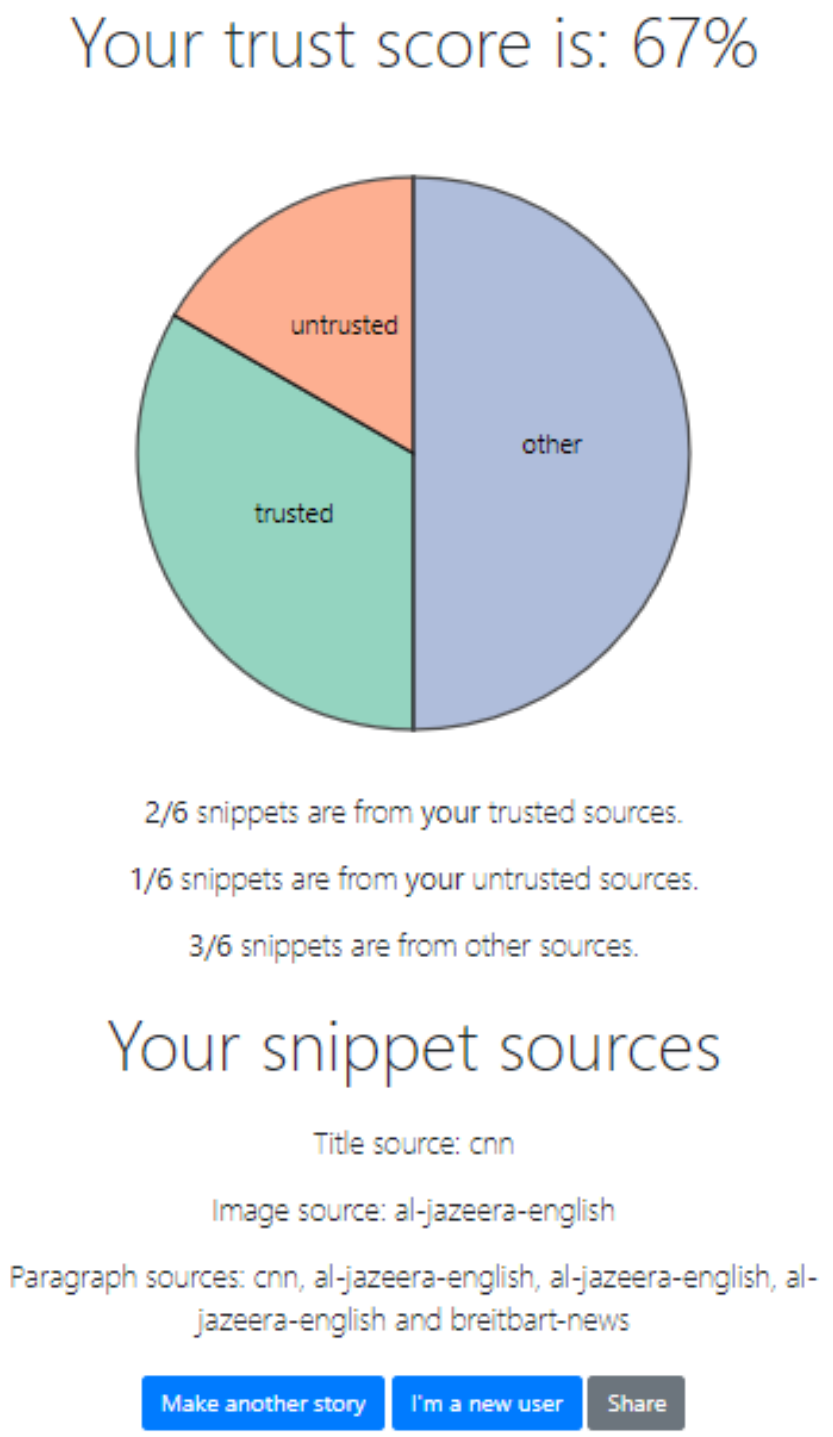

Figure 2-8: The user is shown the chosen sources. The "trust score" indicates the percentage of trusted sources selected amongst both trusted and untrusted sources. Neutral sources are not factored into the "trust score".

2.1.5 Let's See A Game

Inspired by the paper "They Saw A Game" by Hastorf and Cantril, I previously built "Let's See A Game" as part of my Master Thesis for ETH Zurich, an interface that exposes how contemporary media adds fuel to the fire of partisanship. While Hastorf and Cantril presented the same motion picture of a college football game to students from two rival colleges, "Let's See A Game" presents two opposing views ("home feed" and "away feed") of a basketball game. One main observation was that both narrators generally share the same narrative, visual elements and use the same camera feeds.

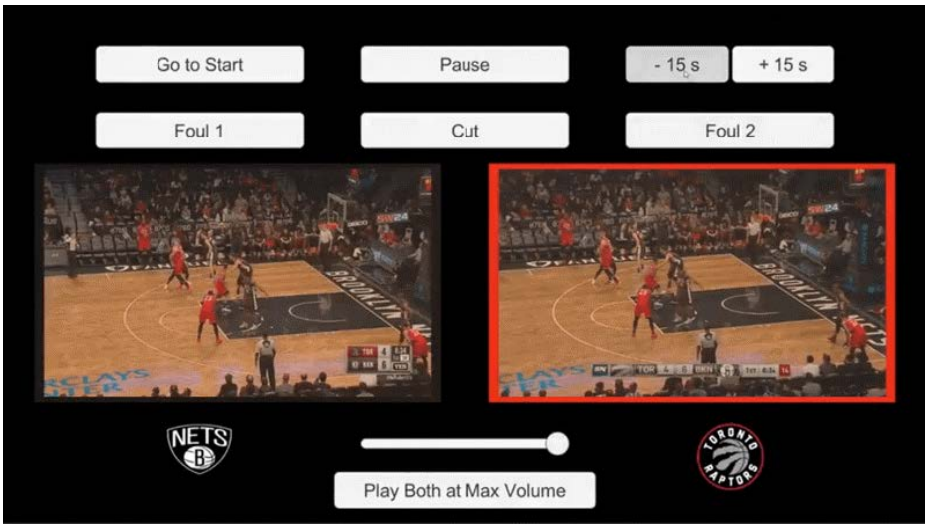

Figure 2-9: The user is exposed to two broadcast feeds (Brooklyn Nets and Toronto Raptors) of the same basketball game as shown on the left and right. Using the slider in the middle, the viewer can tune into the story of their choice and listen to the narrative. Alternatively, they can play both audio tracks at the same volume by pressing the 'Play Both at Max Volume' button.

However, when a foul, interruption or conflict occurs, the stories diverge and the narrators tend to take a stance that favors their team. Furthermore, the visual elements including camera angles are used to support that narrative. For example, at one point of a game one narrator says "[Attacker] steaming ahead – and a blocking foul, a blocking foul will be charged to Isaiah Whitehead [...] See Whitehead stepping in

in the last minute and initiating the contact. Good call." whereas the other narrator recounts "[...] Whitehead slides, you want your toughest call in basketball, that's probably one of them, right there, but Whitehead's position didn't look all that bad defensively." during the same time period.

Subsequently, I expanded this concept to televised news. However, unlike sports broadcasts that are synchronous in time, televised news across different channels typically do not share a common thread and thus are much harder to compare side-by-side. Instead of playing feeds simultaneously, they are presented sequentially.

Figure 2-10: Although the topic is introduced similarly in content, the visuals are telling a different story. Here, FOX news endorses Trump's tweet while MSNBC shows a footage of Trump using swear words.

After manually gathering and comparing a set of clips around a certain topic, I learned that in most cases, stories from conservative news outlets are reported very differently or not reported by liberal news outlets at all, and vice-versa. However, in some cases, they share a similar narration structure comprised of a short overview of topics, followed by topics themselves. Each topic is either elaborated by reporting

additional facts or by inviting opinions from people, followed by a statement or an opinion of the news program host. The latter often leads to a slant and might be one major contributing factor towards media polarization.

Figure 2-11: Both shows invite guests and both hosts ask questions that aim to steer the conversation towards hostility towards their political adversaries. This illustration shows how this can be done. On Fox news, a white, conservative politician is invited. On MSNBC, an Afro-American NFL player is invited. The headline on Fox news states "Athletes, stars launch push to protest anthem" while the headline on MSNBC states "Trump slams NFL players taking a knee, calls for firings, boycotts." The questions asked by the news hosts also lead the conversation to a fundamentally different direction.

While "Let's See A Game" is an useful tool to illustrate how each perspective adds fuel to the fire of partisanship in both sports and news, it is not necessarily suited for news consumption. Furthermore, creating dedicated content for such a format is extremely tedious as it requires a significant amount of video sourcing and editing that cannot be easily automated using machine learning.

2.1.6 Reddit - Change My View

Not all efforts in mind opening come in form of an app or a tool. The Reddit Forum *Change My View*[1] encourages people to "post an opinion you accept may be flawed, in an effort to understand other perspectives on the issue." and to challenge these posts by commenting with a conversation mindset. Tan et al. analyzed these forum posts and showed that "the interplay between the language of the opinion holder and that of the counterargument provides highly predictive cues of persuasiveness" as well as "stylistic choices in how the opinion is expressed carry predictive power". They found that "first person pronouns are strong indicators of malleability [of the opinion holder], but first person plural pronouns correlate with [their] resistance. [...] The use of more paragraphs, bold formatting, and bulleted list are all higher when a malleable view is presented. Taking more time and presenting the reasons behind an opinion in a more elaborated form can indicate more engagement." [31] While the analysis of these forum posts was based on natural language processing, the structural and stylistic lessons can be applied to our daily conversations and writing.

2.1.7 How to keep conversation alive in a polarized world

Another example of mind-opening active participation, conversation and opinion exchange is outlined in Jason Jay's TED Talk "How to keep conversation alive in a polarized world" [20]. There, he presents a set of conversational techniques to overcome gridlock in a polarized conversation in four steps: (1) locate the bait and let go (2) acknowledge the trade-off they fear (3) draw the contrast, help let them go, and (4) invent together. This framework is the result of conducting workshops with more than two thousand business executives and organisation leaders across all sectors and industries.

The first step of the framework means to recognize and let go of "the bait" - factors in play that are keeping one from making a commitment. That might be taking risks, more responsibilities, feeling right, feeling righteous about our goals, certain about

[1] https://www.reddit.com/r/changemyview/

our agenda, staying safe and ask everyone else to do the work (but not ourselves). While this step seems to be only operationally relevant, it can also help in mundane conversations, even political ones. The second step is perspective-taking, looking at the situation or opinion from other people's eyes, and acknowledging their "bait" and trade-offs that they would have to make. Asking oneself "what do they value that we may have undervalued" can help to uncover the trade-offs they might fear. The third step is to break through gridlock by breaking the trade-offs people fear they have to make, using a tool called contrasting: outlining the trade-offs you think that they have to make, highlighting the future of the relationship or the outcome, and emphasizing the willingness to collaborate to brainstorm and to create solutions. The last step is to follow-through the ideation and solution finding process.

Although the framework was developed, tested and applied to break through gridlocks in settings involving stakeholders, Jason Jay emphasizes that these conversational principles can also help in depolarizing society since the debate around social, environmental, economic issues deal with personal or organizational trade-offs [20].

2.2 Mobile First

On a different note, this section (1) outlines a set of statistics that motivate a mobile-first interface design for the purpose of presenting short-form news videos to the viewers, (2) presents a variety of U.S. mobile news app interfaces, and (3) exhibits a selection of popular mobile non-news app interfaces.

As of January 2021, more than eight in ten Americans get their news from digital devices (smartphone, computer or tablet). On these devices, two out of three U.S. adults get their news from websites or apps (68%) or search engines (65%). Amongst digital platforms, the most preferred one for news is news websites or apps [29]. Amongst digital devices, roughly six-in-ten U.S. adults (57%) often get news on a mobile device compared to desktop or laptop devices (30%), upward trending. Furthermore, both younger and older Americans have been increasingly turning to mobile devices for news [32].

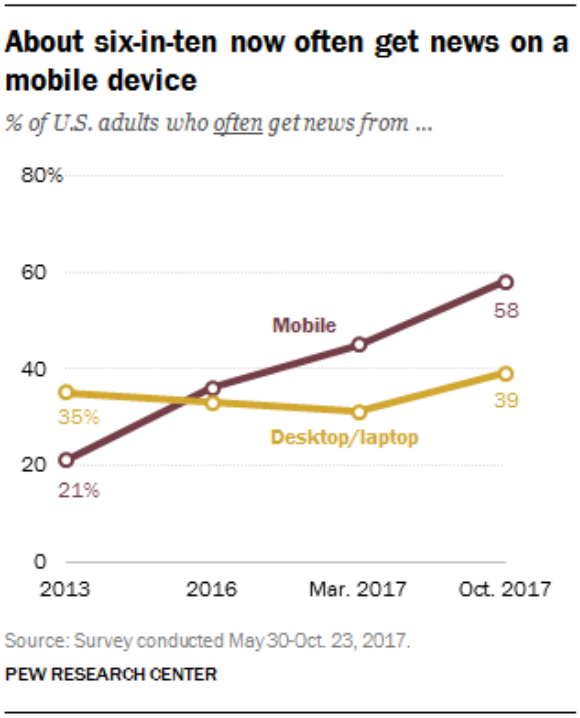

Figure 2-12: Increasingly more U.S. adults use their mobile devices to get news.

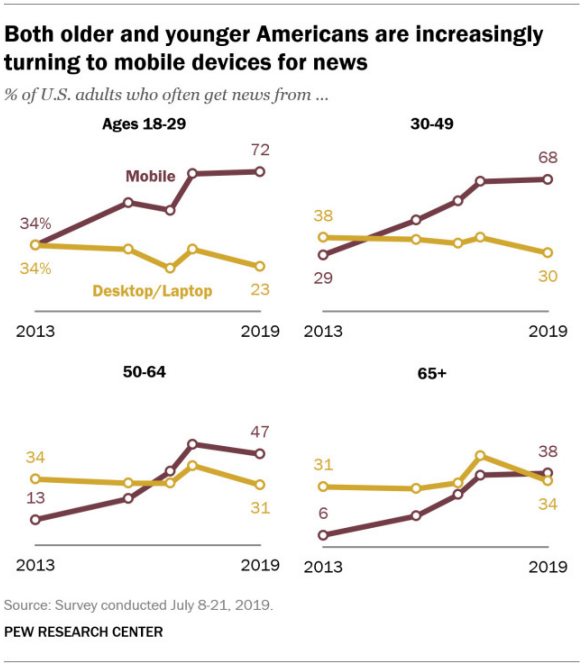

Figure 2-13: Younger and older Americans are increasingly using mobile devices for news.

However, designing a mobile-first interface for news videos for a broader audience is challenging. While most Americans prefer watching to reading the news, they predominantly do it on TV [24], rather than online.

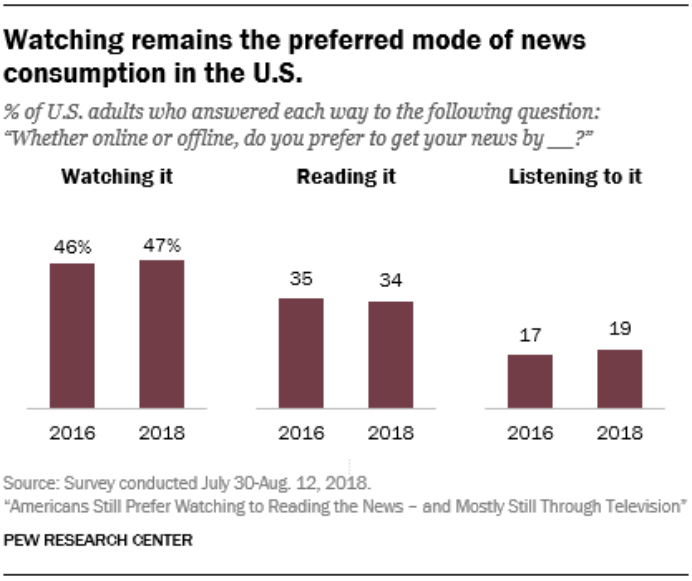

Figure 2-14: Watching remains the preferred mode of news consumption in the U.S.

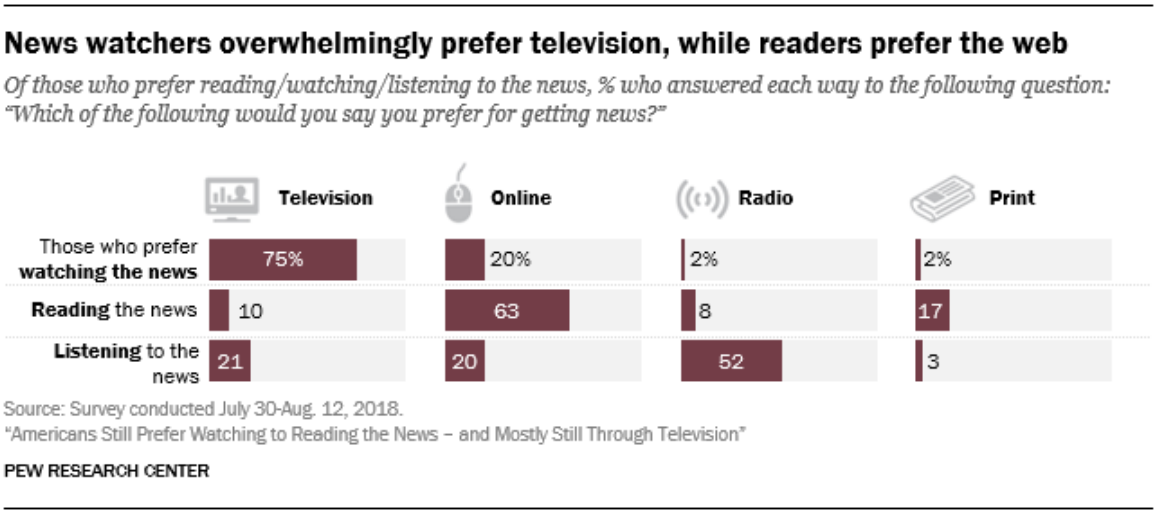

Figure 2-15: News watchers overwhelmingly prefer television over the web.

2.2.1 News Apps

As of April 2021, majority of news apps from U.S. news outlets serve video content. However, the interface design of these apps is rather diverse, as illustrated in the pictures below. Some apps have a dedicated "watch" section (Fox News, NBC News, AP News) or a "live" section (ABC News, MSNBC, CBS News), some integrate their videos at the top of or within text (Fox News, CNN, ABC News, AP News, NBC News). The vast majority use a combination of thumbnails, text description in a scrollable list (Fox News, ABC News, MSNBC, NBC, AP News, CBS News, CNN). The length of the videos varies, but they are mostly longer than two minutes. MSNBC, however, keeps their videos in the watch section short, typically between two and three minutes.

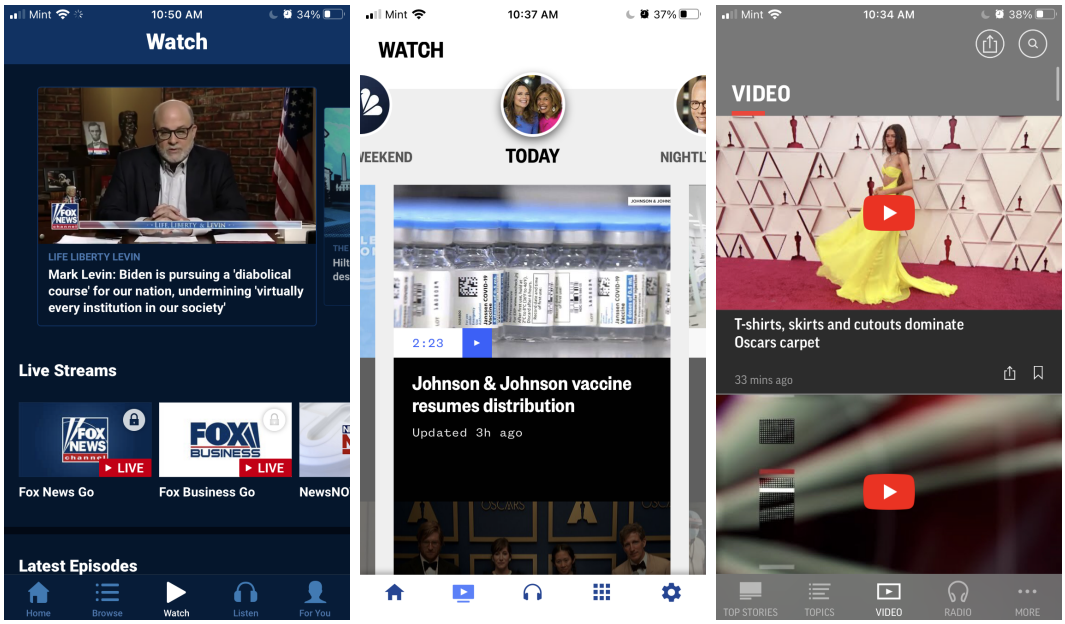

Figure 2-16: Watch section layout of Fox News (left), NBC News (middle) and AP News (right). Fox News uses a swipeable carousel of thumbnails and headlines at the top, followed by a list of live streams and their latest episodes. NBC News uses a double-carousel: the first one presents their content or program, the latter presents a list of video clips accompanied by a group of headlines. AP News simply displays a list of videos with a title.

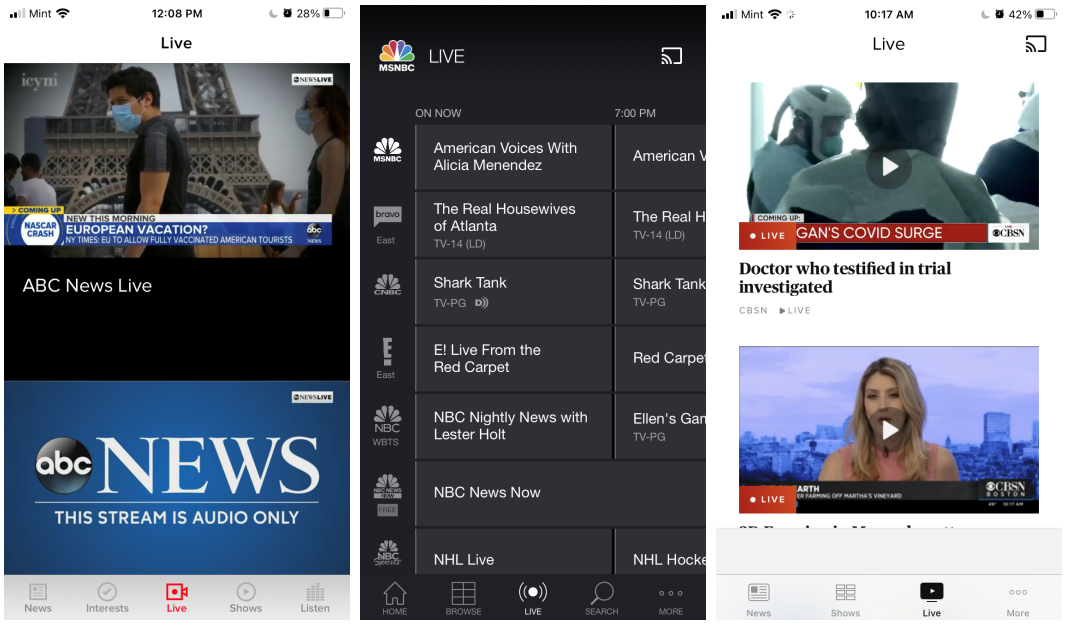

Figure 2-17: Live section layout of ABC News (left), MSNBC (middle) and CBS News (right). ABC News presents a video and audio live stream of their broadcast. MSNBC presents a table of channels and shows that the user can access. CBS News lists multiple live streams using thumbnails and headlines.

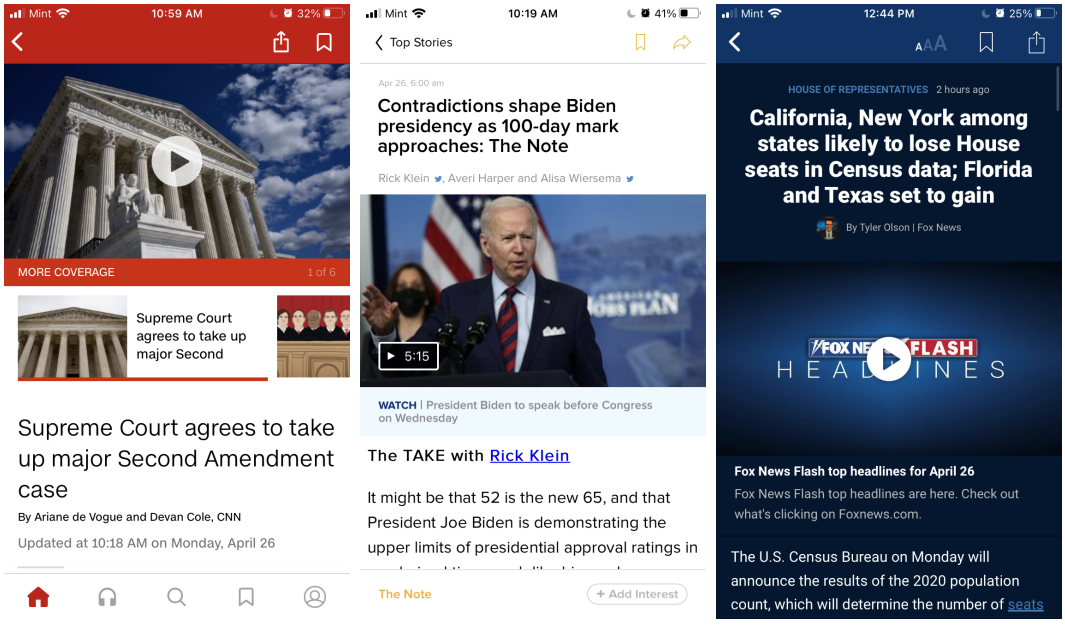

Figure 2-18: Examples of videos embedded in a body of text: CNN (left), ABC (middle), Fox News (right). They all share a similar layout.

2.2.2 Non-News Apps

This section exhibits a selection of long-form and short-form mobile non-news video apps, as well as the popular dating app *Tinder*, from which some design elements are borrowed.

The Netflix app presents entertainment content such as movies or TV shows in a long-form format. Users can browse through the catalog and have shows recommended to them. However, in March 2021, they added a "fast laughs" feature for iPhone users that mirrors TikTok's short-form video format [12].

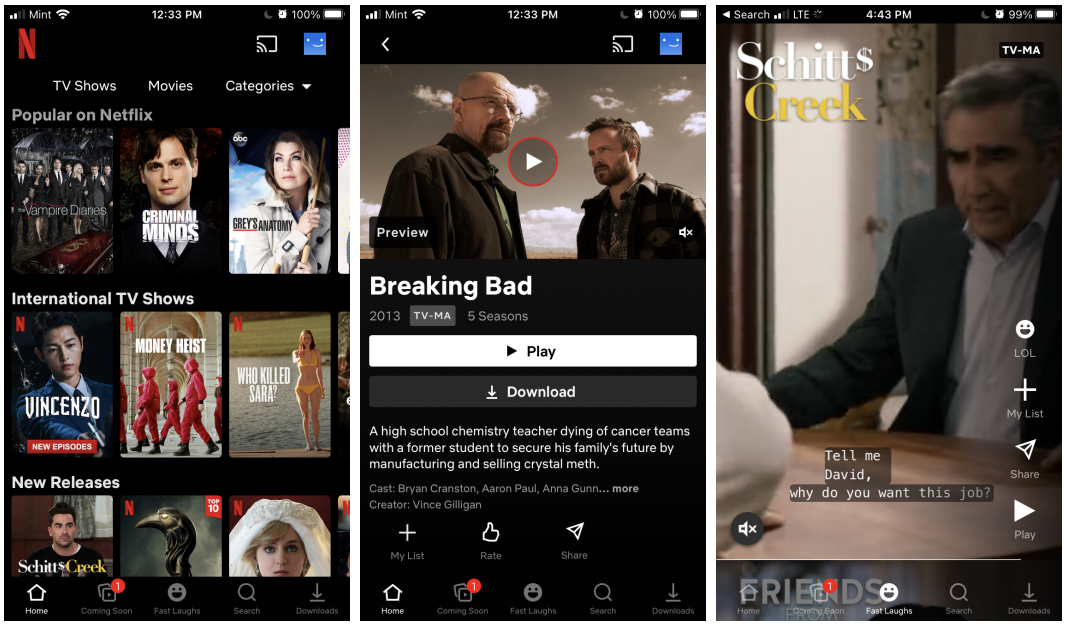

Figure 2-19: Netflix app layout: Users can select their shows by tapping on the image (left). Users can watch the preview of the show, read or download the description (middle) - if they press the play button, the video automatically expands to full screen into landscape mode and starts playing. The fast laughs feature imitates the layout of TikTok and shows video clips in vertical mode (right).

The YouTube app serves a variety of video content of all lengths. Similar to Netflix, users can browse through categories and have content recommended to them. Furthermore, users can access content they have subscribed to. The default layout is in portrait mode and the video does not expand automatically.

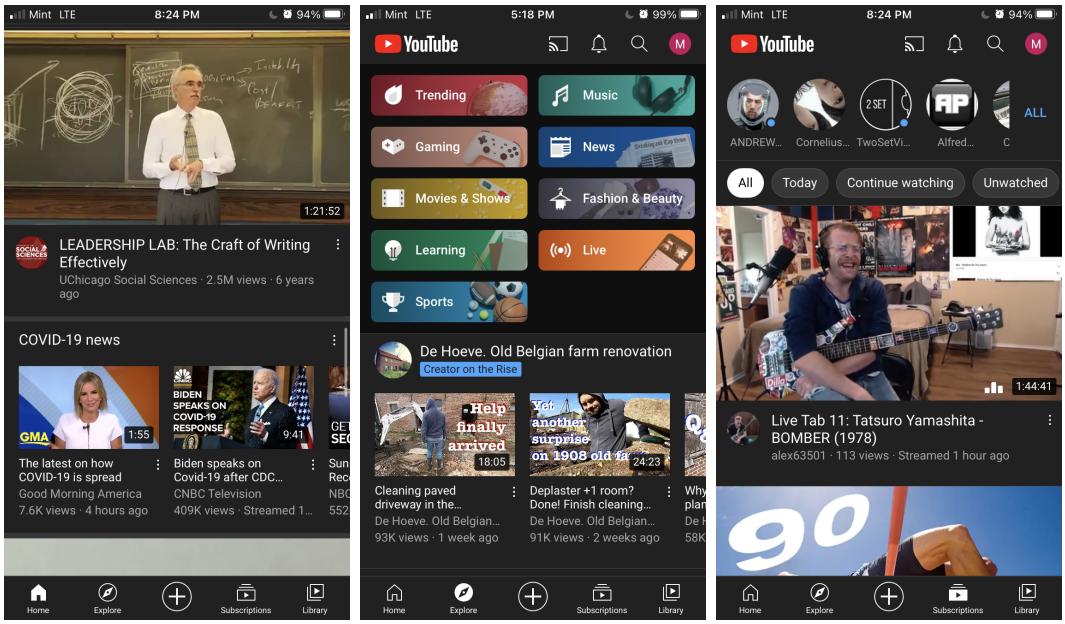

Figure 2-20: YouTube app layout: YouTube recommends a set of videos vertically as well as horizontally (left). Users can explore a broad range of content (middle), and they can watch content from channels they have subscribed to (right).

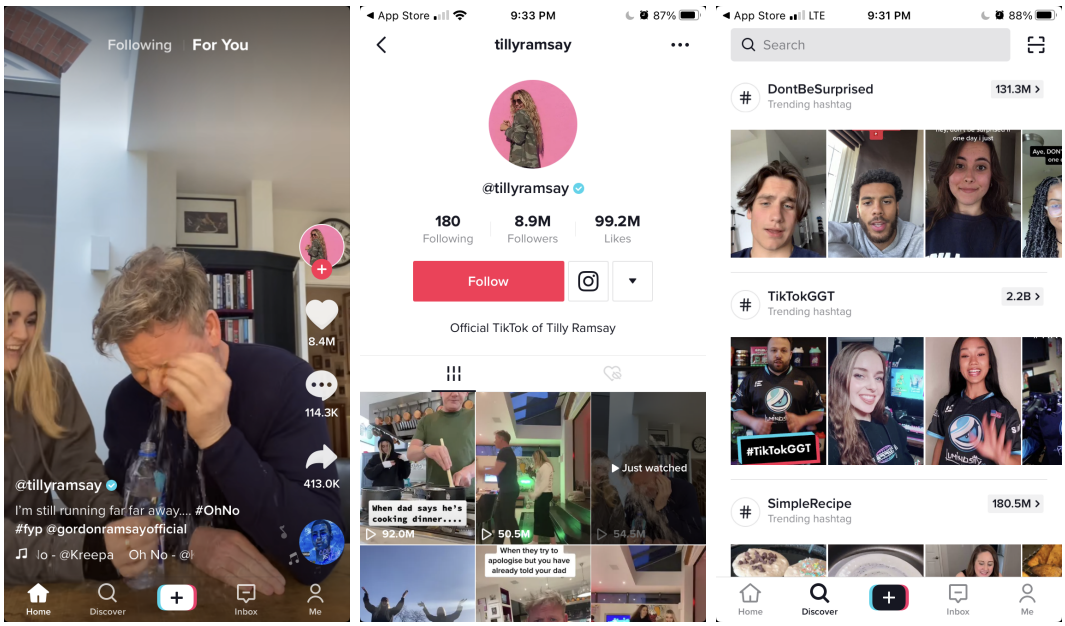

Figure 2-21: TikTok app layout: TikTok exclusively recommends videos vertically. The user can tap on a content creator's icon to watch a list of videos they have created. Users can also discover content, sorted by trending hashtags (right).

The TikTok app has a very simple interface compared to Netflix and YouTube. It exclusively serves short-form videos in portrait mode. The videos are tailored to the user on the "For You" page, based on the interactions the user had using the app. Alternatively, users can explore the content of content creators they have followed on the "Following" page. Users navigate a stream of videos by swiping up or down, and they can like, comment or share a clip. Similar to YouTube, there is a discovery page. However, the content is sorted and ranked by hashtags. Unlike Netflix and YouTube, it also has a social function - people can directly message each other. The most important feature is the content creation button (+) through which users can shoot a video, add effects and an audio clip that goes with the video.

Tinder, the dating app, has a simplistic interface design centered around swiping. Users can swipe right or left on pictures to indicate whether they like or dislike a person's profile. Alternatively, they can press the like (green heart) or dislike (red cross) button. These main design features were adopted and used for the design of *Enlightened*.

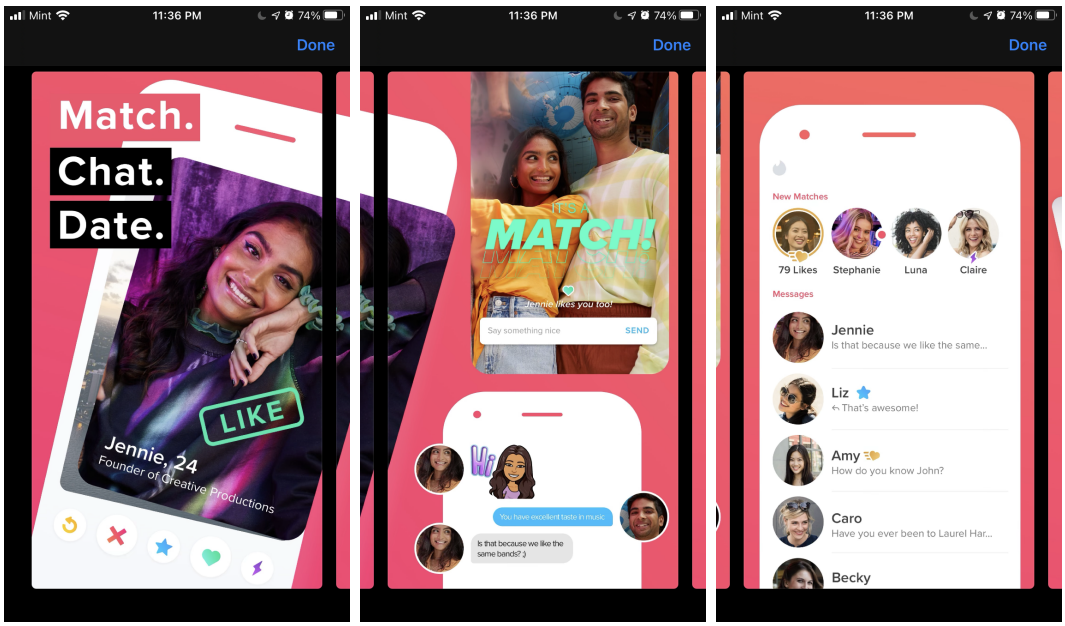

Figure 2-22: Tinder app layout: Users swipe through images of other users' profiles (left). If two users like each other, they match, and can start texting each other (middle). The second main screen displays a list of conversations with matches (right).

2.3 Enlightened

The design of *Enlightened* was an iterative process. For the sake of brevity, this section only presents the final design used for the user study, along with commentary on my decision making process that lead up to the final design. My goal was to design a fun and engaging mobile app in which users are exposed to short-form news clips sourced from news outlets with bias across the political spectrum. As users interact with the app, they should gain a broader view, understand their media source bias, and hopefully also contribute to restoring the *common ground*.

New users are introduced to the functionalities of the app through a set of welcome screens. At the end, they are prompted to choose either "donkey" or "elephant" to self-identify themselves as either liberal or conservative.

From a stylistic point of view, the name *Enlightened* and the color code revolved around the theme of illuminating darkness, a metaphor for restoring *common ground* in the current fragmented and polarized media landscape. As a result, the app's icon and the like/dislike buttons of the "Tinder mode" are light bulbs, the background color is consistently black (representing darkness). Furthermore, depending on a users choice (donkey or elephant), the user's statistics page will show either a donkey or an elephant, along with a blue or red navigation bar.

At the core of *Enlightened* is an interface that allows users to swipe through short-form news clips. Inspired by Tinder's simplistic main interface, I decided to use a similar layout. However, news videos are in landscape mode by default, and Tinder's layout is in portrait mode. Furthermore, all news apps and non-news apps exhibited in the previous sections use portrait mode as the default mode of operation. To find a middle ground between portrait and landscape mode, I designed the entry-point of the experience in portrait mode, and prompted the user to rotate their phone into "Tinder mode". In "Tinder mode", they can swipe up or down on video clips to indicate whether they like the content (and watch a little bit more) or dislike the content (and watch something else).

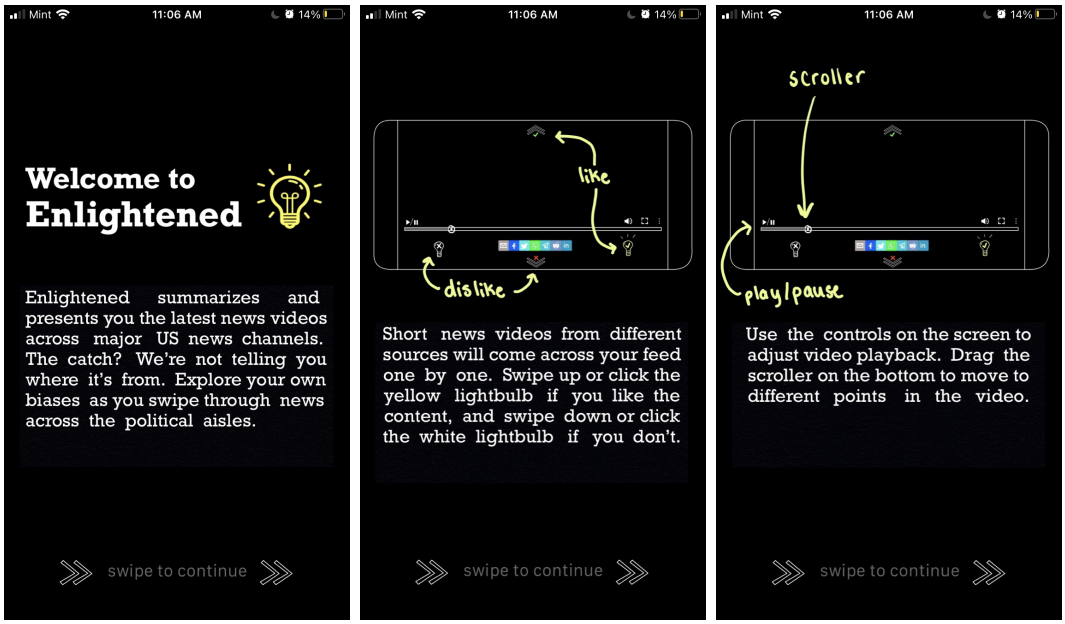

Figure 2-23: A set of welcome pages introduces the user to the app on the first start-up.

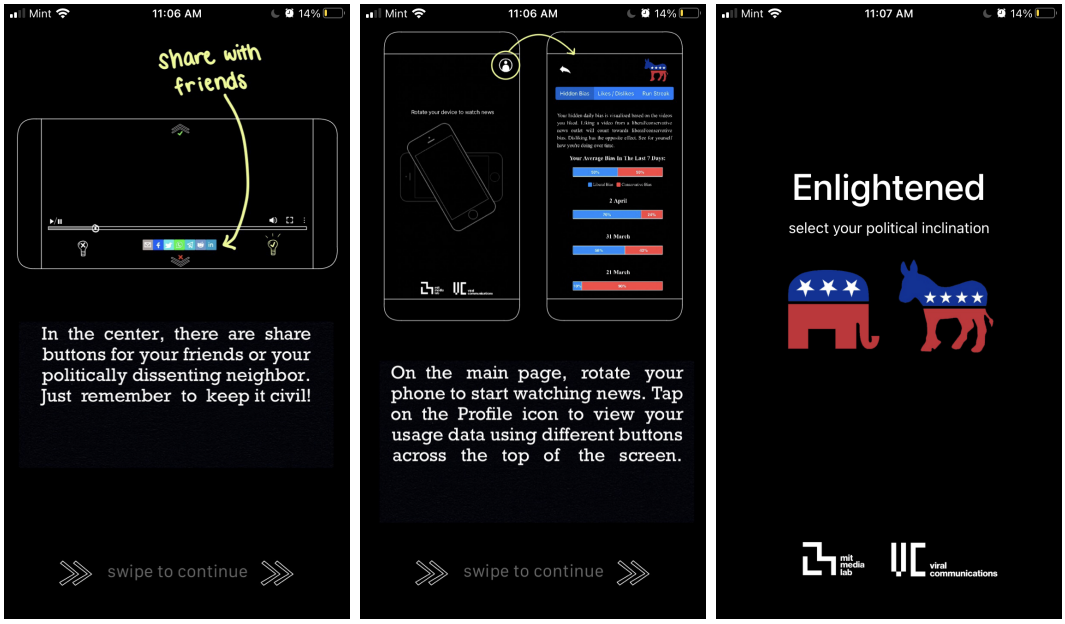

Figure 2-24: Users swipe through these welcome pages and select their political inclination.

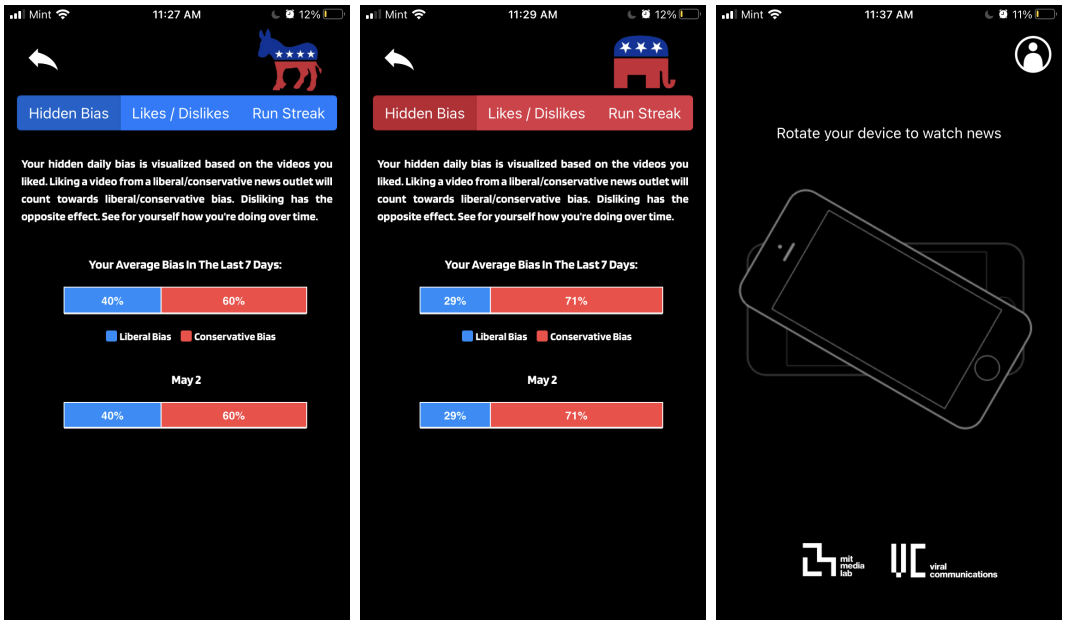

Figure 2-25: Depending on a user's choice during self-identification, a blue (left) or red (middle) statistics page layout is rendered. Users can access these statistics pages by pressing the profile icon on the main entry-point page (right).

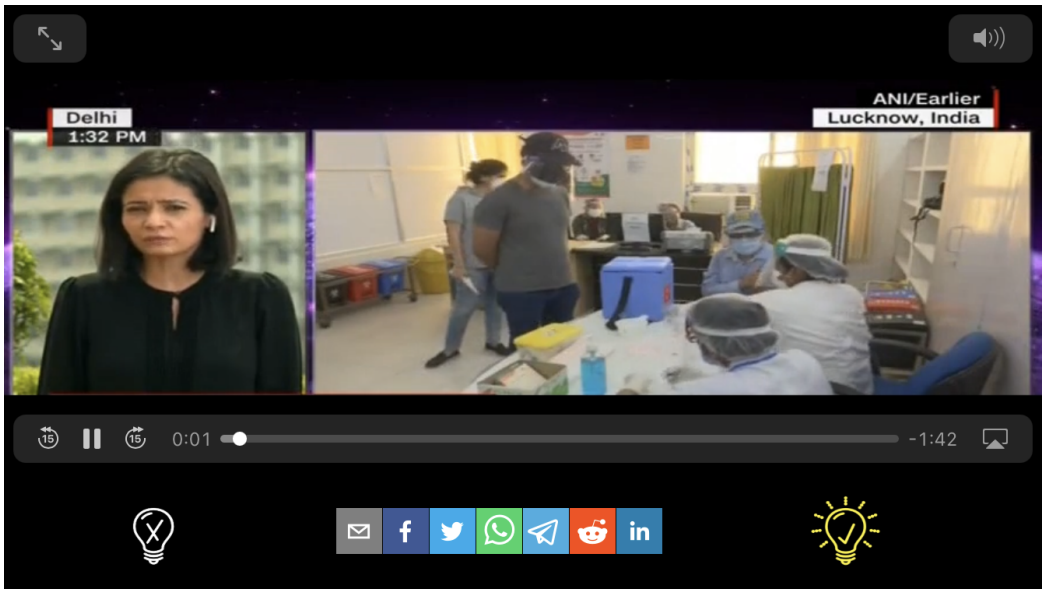

Figure 2-26: Enlightened's main interface: Users are presented a feed of latest short news clips. Users can like or dislike the content by either pressing the lightbulbs or by swiping up or down. Furthermore, they can share the content to social media platforms.

Alternatively, I could have cut the news videos vertically using algorithms such as Google's Autoflip[2] to create an experience similar to TikTok. However, in TV news there are oftentimes multiple speakers side-by-side and sometimes a clip is played where the news anchor is not shown at all. Due to the risk that the algorithm might fail in a majority of situations, I decided not to pursue this design.

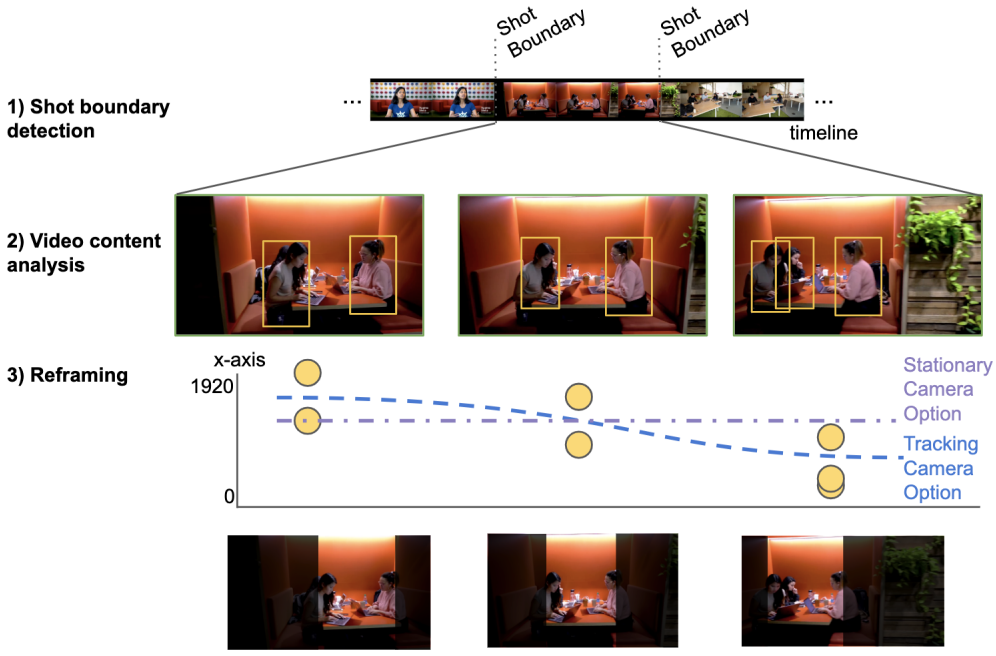

Figure 2-27: Google AutoFlip Overview: between shots, the algorithm identifies salient content before the scene is reframed and a camera mode (stationary, panning or tracking) is chosen. Afterwards, an optimal cropping window is selected for each frame.

My advisor suggested to use NewsMaker's concept of hiding the source by masking the lower third of the news videos. The main concern was that if the news source was revealed, some viewers will immediately dislike the content. The hypothesis was that by masking the source, users won't immediately dislike the content and possibly expand their perspective being exposed to opinions other than their own.

To help users to understand their media source bias, *Enlightened* visualizes a user's average bias and their like and dislike distributions across the news sources. The bias

[2]https://ai.googleblog.com/2020/02/autoflip-open-source-framework-for.html

is calculated based on the videos the user liked or disliked. Liking a video from a liberal/conservative news outlet counts towards liberal/conservative bias. Disliking has the opposite effect. The like and dislike distribution represents the aggregate likes and dislikes across five major TV news channels (MSNBC, CNN, BTV, ABC, FOX). A "Run Streak" page adds a gamification element to the experience. Users can collect a run streak if they use the app for consecutive days. Furthermore, they have access to the number of clips they watched, the total watch time and average watch time per clip.

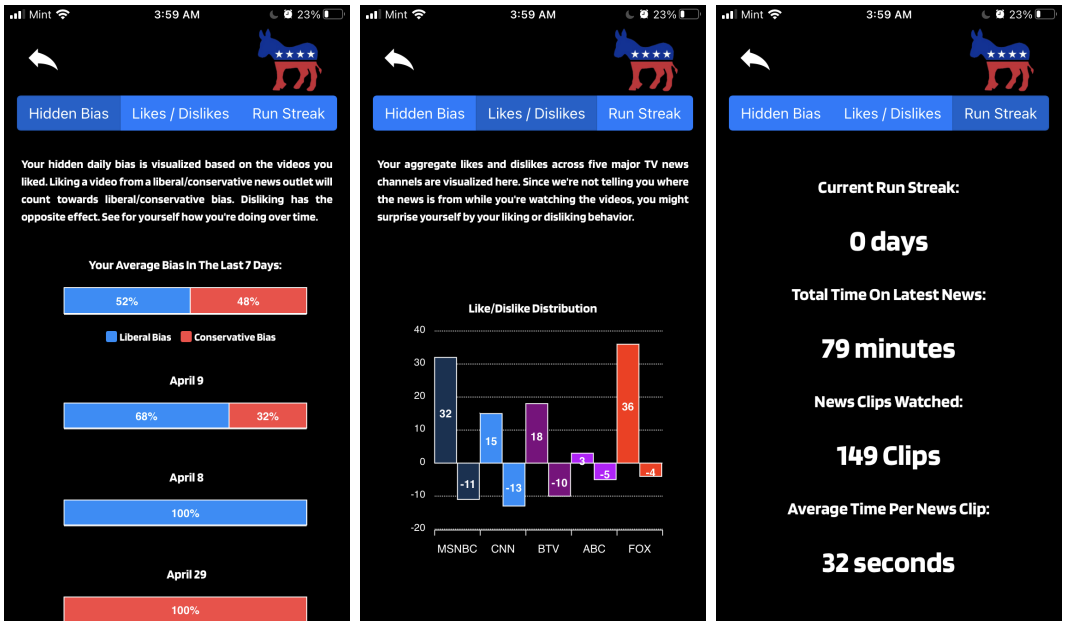

Figure 2-28: Enlightened presents personalized statistics about a users' behavioral data. The "Hidden Bias" section contains an average value and a timeline based on the videos a person has liked or disliked (Left). The "Likes/Dislikes" section simply presents the distribution of likes and dislikes (Middle). The "Run Streak" page simply contains a summary of a user's app usage data (Right).

Chapter 3

Implementation

This chapter explains how *Enlightened* was made. There are three key ingredients that make it work: a real-time news recording and processing system, a recommendation system, and a progressive web application. A system overview section shows what these software systems do on a high-level and how they interact with each other. The following sections describe and explain these systems in detail.

3.1 System Overview

The images illustrated at the end of the design chapter are screenshots of the progressive web application (PWA) *Enlightened*. A PWA is a responsive web application that brings a native app-like user experience to cross-platform web applications. The *Enlightened* PWA fetches a list of videos catered to the user from the recommendation system (RecSys). It also records the user's interaction with the videos, and stores usage related data into a database. The RecSys generates video playlists based on what a user has already watched as well as short-form videos generated with *SuperGlue*, a real-time news recording and processing system. Due to computational resource constraints, *SuperGlue* only records and processes content from five news channels (MSNBC, CNN, Bloomberg TV, ABC and Fox News), as illustrated in the design section. The summarization and source masking of the video content are part of *SuperGlue*, as well as the generation of metadata used for usage data visualization.

3.2 News Sourcing: SuperGlue

SuperGlue is the backbone for *Enlightened*. It records and processes U.S. broadcast television news to generate a broad range of metadata that can be used for organization, visualization and summarization. It is a multi-year research project of the Viral Communications Group at the MIT Media Lab that has been continuously reshaped and expanded by multiple generations of graduate students with the help of an army of industrious undergraduate students (UROPs) and visiting students, as well as MIT Media Lab's Network Computing Services Group (NeCSys). It was funded through the Ultimate Media Program from January 2013 until September 2019. This section elaborates on its current high-level architecture, its modular structure, each module and their technology stack, and the deployment structure. To ensure that previous work on SuperGlue is credited, and to avoid self-plagiarism (I have modularized and expanded a formerly monolithic SuperGlue as part of my Master Thesis for ETH Zurich as a visiting student in the Viral Communications Group), a list of people that have contributed to a part of the system is given at the end of every module section.

3.2.1 High-Level Architecture Overview

SuperGlue can be split into two functional parts: a digital video recording system (DVR) and a video processing system (*SuperGlue* modules). The DVR records and stores the video, its closed captions, the link to the video alongside with some basic metadata such as the creation date, channel information (e.g. Fox News), duration of the recording (fixed to 10 minute segments), start and end times of the program (typically 30 minutes) and genre of the video (fixed to news). The video processing system is a modular piece of software where each module extracts and stores video meta-data that can be used for subsequent modules, including summarization and news source masking.

3.2.2 DVR

The DVR hardware is located in the MIT Media Lab. An array of tuners are maintained by NeCSys and the UDP streams are broadcast within the Media Lab network. The configuration interface of the tuners was built and is maintained by NecSys. The *SuperGlue-DVR* records the UDP streams and stores the content on a virtual machine (VM) on one of NeCSys computing infrastructure along with some metadata in a MongoDB database located on a different VM. The *SuperGlue-DVR* is a Python based code deployed as a Docker container on another VM. For the scope of this thesis, the *SuperGlue-DVR* was configured only to record Fox News, MSNBC, ABC News, CNN and Bloomberg TV.

Under the hood, *SuperGlue-DVR* queries the Gracenote Developer API to receive regional TMS data every 45 minutes. It asks for the data from the current time to the next 6 hours (maximum window size) and updates the channel information of the configured channels (Fox, MSNBC, ABC, CNN and BTV). The recording process uses the TMS data, FFmpeg and ccextractor to record the videos and closed captions. The videos are recorded in a 16:9 aspect ratio using a x264 encoder and the audio is encoded with advanced audio coding (aac). As mentioned before, the recorded files are stored on a VM and their references in a MongoDB database.

The following people have contributed to the *SuperGlue-DVR* repository, in order of descending significance: Robert Tran (UROP), Tomer Weller, Jasmin Rubinovitz, David Anderton-Yang, Sebastian Mendez (UROP), Johan Cervantes (UROP), Mike Hao Jiang and Kalliroi Retzepi.

3.2.3 SuperGlue Modules

The recorded videos are subsequently going through a processing pipeline consisting of a group of *SuperGlue* modules, all of which inherit from a base module class, follow a module dependency graph, and are deployed on MIT Media Lab NeCSys machines. A detailed description of each processing module is given at the end of this subsection.

Base Module

The base module has a set of basic functionalities such as creating a processing dependency graph, managing the connection to the MongoDB database, fetching new records generated by the *SuperGlue-DVR* to process, processing one or multiple records, locking records (so that one record is processed at a time by a module), logging internal processes on Papertrail (cloud logging infrastructure), and storing files on Media Lab VMs and Amazon AWS servers. The base module is part of the requirements.txt file of all child modules.

The following people have contributed to the base module repository, in order of descending significance: Sebastian Mendez (UROP), Johan Cervantes (UROP) and Mike Hao Jiang.

Module Dependency Graph

The module dependency graph is a directed acyclical graph in which a video needs to be processed. The video metadata also include fields such as needs_proc, in_proc and done_proc representing a list of modules that still need to process the video, are processing the video, and have already processed the video. For the purpose of *Enlightened*, the graph simplifies to a linear sequence of modules through which each video goes through. Each video needs to be processed by the following nine modules to be presentable for *Enlightened*, in the following order: Thumbnail, Closed Captions, Transcription, Words, Captions Aligned, Commercial Skip, Segmentation, News Summarization and News Ticker Removal.

Deployment Structure

All modules are deployed on Media Lab VMs, and each module has multiple instances running. All modules are written in Python and have a Dockerfile as well as a requirements.txt file that specify the dependencies that need to be installed for the module to work. The Dockerfile contains a set of instructions on how a Docker image should be built. All Docker containers are a based on the python:3.7-stretch

image, base module repository and the environment variables specify the URL to the MongoDB, metadata database, URL and key to Bubble, URL and port of Papertrail, AWS key, access id and bucket name. Further dependencies vary based on the processing requirements and libraries used. Due to a varying processing rate of each module, some modules have been scaled up to avoid congestion. For example, there are six transcription containers, while only one words container is needed. Because of computational resource constraints (limited NeCSys hardware), *SuperGlue* can only process recordings from five news channels. I decided to choose Fox News, MSNBC, ABC News, CNN and Bloomberg TV as they represent a broad spectrum of opinions and perspectives. In total, one VM is dedicated to the DVR, 12 VMs are dedicated to processing, one VM to local file storage (Bubble) and metadata storage (MongoDB). Furthermore, Amazon Web Services S3 is also used for storage of *Enlightened* video clips that need to be readily accessed at all times. *Enlightened* serves the video clips stored on AWS and the RecSys queries the MongoDB data to curate and serve the video playlists to users.

Thumbnail Module

The thumbnail module is the first module of the processing pipeline. It uses the video and its metadata generated by the *SuperGlue-DVR* and creates a five second thumbnail of the video in a subprocess using

```
ffmpeg -y -i input_url -ss 5 -t start_time -filter_complex "fps=10,
scale=-1:240" output_url
```

and writes the file into Bubble. Afterwards, it logs the progress on Papertrail and updates the MongoDB document corresponding to the video by setting the thumbnail URL, as well as updating its report fields to indicate the duration of the thumbnail (5 seconds per default) and the start time of the thumbnail.

The following people have contributed to the thumbnail module repository, in order of descending significance: Tomer Weller, David Anderton-Yang, Mike Hao Jiang, Veronika Eickhoff (visiting student), Sebastian Mendez (UROP) and Johan Cervantes (UROP).

Closed Captions Module

The closed captions module extracts closed captions from the closed captions file (.srt) generated by the *SuperGlue-DVR*. It uses the pysrt library to generate formatted subtitles and full text. The formatted subtitles are represented as a dictionary of subtitle segments. Each formatted segment has a start_time, end_time and a corresponding text. The full text is created by appending the text of each formatted segment. The updated MongoDB document includes the formatted subtitles, full text as well as the report fields entailing the number of segments as well as the length of the full text.

The following people have contributed to the closed captions module repository, in order of descending significance: Tomer Weller, Mike Hao Jiang, David Anderton-Yang, Sebastian Mendez (UROP) and Johan Cervantes (UROP).

Transcription Module

The transcription module produces a transcript by converting extracting the audio form the video using FFmpeg and using sphinx4 as speech-to-text to generate a text file. The FFmpeg command to extract the audio with a format that sphinx4 can deal with (16khz mono 16bit wav) is

```
ffmpeg -i input_name -acodec pcm_s16le -ac 1 -ar 16000 output_name
```

The sphinx4 speech-to-text comes as JAR (java package file format) file, and is run as a subprocess with

```
java -Xms2048m -Xmx2048m -cp path_to_jar aligner_url wav, temp_script_path,
temp_file_path
```

The generated transcript is stored as a .txt file on Bubble and the url to the transcript is stored in MongoDB. Furthermore, the video download time, wav file conversion time and sphinx aligner execution time are stored in the report dictionary of the MongoDB document corresponding to the video.

The following people have contributed to the transcription module repository, in order of descending significance: Tomer Weller, David Anderton-Yang, Johan Cervantes (UROP), Sebastian Mendez (UROP) and Mike Hao Jiang.

Words Module

The words module breaks down the sphinx4 transcription into single words and stores them into a MongoDB database that is different from where all other metadata is stored. Each word comes with the transcription segment it belongs to, the start and end date of the transcription segment, the duration of the segment, its corresponding video ID, video URL and air date. The number of words stored are documented in the report dictionary of the document in the main MongoDB database.

The following people have contributed to the words module repository, in order of descending significance: "joel" (UROP), Tomer Weller, Mike Hao Jiang, and Johan Cervantes (UROP).

Captions Aligned Module

The captions aligned module uses the output of the transcription module to align the closed captions to the video in case there is a misalignment found. The misalignment is found by finding the longest common substring between the formatted closed caption .srt file and the speech-to-text transcription. The start times of these substrings are compared and if the start times don't match up, the time shift is noted and used to shift the timestamps in the original .srt file as well as the corresponding full text. Afterwards, the URLs to the aligned subtitles, aligned closed captions, and aligned full text are all stored in the MongoDB. The report dictionary of the document stores the number of formatted closed captions, the length of aligned full text, the longest sequence length, the time shift and the matching sequence.

The following people have contributed to the captions aligned module repository, in order of descending significance: Jasmin Rubinovitz, Veronika Eickhoff (visiting student), Sebastian Mendez (UROP), Mike Hao Jiang, and Johan Cervantes (UROP).

Commercial Skip Module

The commercial skip module finds commercials in a video and generates a new video file with no commercials and matching subtitles. However, it does not work

all the time, unfortunately. It is based on the comskip.ini file downloadable from https://www.comskip.org/. The commercial skip module downloads the video locally and runs the comskip.ini file on the video using the following command

```
comskip --output=video_path --ini=comskip/comskip.ini output_directory
```

The .edl file generated by the command is then used to generate a commercial-free video using FFmpeg and corresponding subtitles, as well as counting the number of commercials present in that video. The .csv and .xml files generated by the command are added to the metadata. Furthermore, if commercials were found, the new video and the corresponding .srt files are stored on Bubble and the link to those files are stored in the MongoDB document corresponding to the files. Futhermore, the information on whether a commercial was removed as well as the number of commercials found are stored in the report dictionary of the document.

The following people have contributed to the commercial skip module repository, in order of descending significance: Jasmin Rubinovitz, Mike Hao Jiang, Sebastian Mendez (UROP) and Veronika Eickhoff (visiting student). It should be noted that another commercial skip module was developed by Veronika Eickhoff and Johan Cervantes (UROP) using a pure computer vision based approach. However, since that module was based on Google's Vision API (and hence, rather expensive), for the scope of this thesis, only the open source and free commercial skip module was used.

Segmentation Module

The segmentation module finds story segments based on the output of the commercial skip module including the updated closed captions. It segments the closed captions based on the occurrence of "> > >" and determines the story boundaries based on commercials detection and the closed captions. If we have segments and they are news, a thumbnail is created and the text for each segment is generated. While *Enlightened* does not make use of the thumbnail, the segment text is relevant for subsequent modules on which *Enlightened* depends. The updated dictionary informs

whether the video contains news, lists the news story segments and the closed captions corresponding to commercials, as well as news story segments.

The following people have contributed to the segmentation module repository, in order of descending significance: Jasmin Rubinovitz, Tomer Weller, Mike Hao Jiang, Veronika Eickhoff (visiting student), Sebastian Mendez (UROP).

News Summarization Module

The summarization module uses extractive summarization on the text of news story segments. It uses the free open-source Python library Gensim (a variation of the TextRank algorithm, based on [3]). The TextRank algorithm is a graph based ranking algorithm where text sentences are ranked based on their importance. It executes a keyword extraction task (fully unsupervised) and a sentence extraction task. After some experimentation with the library on news story segments, I decided to use a 30% ratio, discarding 70% of the segments' text. Afterwards, the shortened text and closed captions time boundary information are used to cut the video into short clips of at least 20 seconds of length. Summarized segments that are only 5 seconds apart were merged into one major segment to avoid overly awkward transitions. These summarized news story segments provide the raw material for *Enlightened*. An older version of this module used for my Master's thesis for ETH Zurich was based on the open-source Python library summa, a different implementation of [3].

The following people have contributed to the news summarization module repository, in order of descending significance: Mike Hao Jiang, Johan Cervantes (UROP), Sebastian Mendez (UROP) and Veronika Eickhoff (visiting student).

News Ticker Removal Module

The news ticker removal module removes the lower third of the summarized news story segments, visually masking the news source. It was built by Johan Cervantes (UROP) and essentially runs the following FFmpeg command in a subprocess:

```
ffmpeg -i {} -filter:v crop="in_w:in_h/(3/2):0:0", pad="in_w:in_h+66:0:0" {}
```

3.3 Enlightened

The progressive web application *Enlightened* was built in javascript using React[1], Firebase[2], Axios[3], ApexCharts[4], Bootstrap[5], dotenv[6], env-cmd[7], React Player[8], React Router[9], React Screen Orientation[10], React Scripts[11], React Share[12] and React Swipeable[13]. The project was bootstrapped with create-react-app[14].

The following subsections explain the high-level architecture of the progressive web app, the database schema and the thoughts behind its design, how the Firebase database is used to store user interaction data anonymously, how the service worker and the manifest file operate to make the web app progressive, how routing within the app is done (including screen rotation), how the masked news clips are fetched, how the swiping works, how user interaction is tracked and how the personal user interaction data is visualized.

Prior work on Enlightened was built in React Native[15] as part of my Master Thesis for ETH Zurich. However, it was very simple and didn't have nearly as many functionalities as the PWA developed in this thesis. The MIT engineering UROPs Yodahe Alemu, Dagmawi Haile, Azariah Beyene and Wellesley College design UROP Sophia Rim used my ETH Zurich thesis work and built on top of that. However, I decided to build the PWA from scratch in February 2021 with the support of two MIT engineering UROPs Johan Cervantes and Helen Liu.

[1] https://reactjs.org/
[2] https://firebase.google.com/
[3] https://github.com/axios/axios
[4] https://apexcharts.com/
[5] https://react-bootstrap.github.io/
[6] https://www.npmjs.com/package/dotenv
[7] https://www.npmjs.com/package/env-cmd
[8] https://www.npmjs.com/package/react-player
[9] https://reactrouter.com/web/guides/quick-start
[10] https://www.npmjs.com/package/react-screen-orientation
[11] https://www.npmjs.com/package/react-scripts
[12] https://www.npmjs.com/package/react-share
[13] https://www.npmjs.com/package/react-swipeable
[14] https://create-react-app.dev/
[15] https://reactnative.dev/

3.3.1 High-Level Architecture: Progressive Web App

The progressive web app is a standalone web application serves summarized and masked news clips from *SuperGlue* to the user using the React Player npm package. A list of video_urls is fetched with the Axios npm package from the RecSys. Each video_url points to a publicly available summarized and masked news clip stored on AWS and is accessible 24/7. The navigation between the app screens uses React Router, React Screen Orientation and conditional rendering. The "Tinder interface" of *Enlightened* leverages the combination of React Swipeable, React Player and React Share npm packages. The welcome screens also make use of React Swipeable and a set of images, designed by UROP Helen Liu. The user interactions are tracked and stored in a Firebase database. And that data is retrieved and analyzed by the RecSys before it is served to the progressive web app to visualize a user's statistic. The combination of ApexCharts and Bootstrap is used to render the statistic pages. By using a service worker and a manifest file, as well as an app icon with multiple sizes, designed by UROP Sophia Rim, the web app was made into a PWA. The npm packages dotenv, env-cmd and React Scripts were used to set up the development and deployment environments of the PWA.

3.3.2 Firebase Anonymous Authentication & Database

To quantitatively understand whether short-form news videos can captivate people and open their minds, I had to think about what information I could and wanted to gather from people, and how I would want the database schema to reflect that. I did not want to collect any personal information from users to make their on-boarding experience as smooth as possible. As a result, I resorted to the anonymous authentication feature provided by Firebase. The Firebase database is a NoSQL database in which data can be stored and retrieved in a .json format. The database stores whether users identify themselves as Republican or Democrat, whether they have liked or disliked a video, what videos they have watched and for how long, and if they have shared a video, what platform they used.

3.3.3 Routing

The navigation between screens is handled using the React Router library. The components <BrowserRouter> and <Route> are set up in the App.js file (React App entrypoint), and the <Redirect> component is used to navigate between the screen components. On the screen where the user is prompted to rotate their phone, conditional rendering is used in conjunction with the React Screen Orientation npm package and its components <DeviceOrientation> and <Orientation>. The React Screen Orientation npm package detects whether the device is in portrait or landscape mode and the event listener can trigger a state change that can be used to switch to and away from "Tinder mode".

3.3.4 Video Sourcing and Swiping

The PWA uses the Axios npm package to handle GET requests to the RecSys. The RecSys serves a list of video URLs on AWS that were generated by *SuperGlue*. The <VideoPlayer> component of the React Player npm package is used to play the video using the URL. The user interaction tracking system, Firebase database and the RecSys work together to ensure that the playlist does not contain videos that the user has already watched. This is done by sending the user's unique anonymous Firebase ID as a GET request parameter to the RecSys, and the RecSys returns a new playlist without watched videos upon consulting the user's list of watched videos in the Firebase database. Unfortunately, the <VideoPlayer> component is not swipeable by default. In order to make the video swipeable, the useSwipeable function of the React Swipeable npm package is used.

3.3.5 User Interaction Tracking

A couple of functions and components were written to track a user's interaction with the app. Essentially, they ensure that the fields of the Firebase database are populated and updated over time. This is mainly done by utilizing Firebase functions firebase.database().ref(path_of_reference).once() and .update(), as illustrated in the

code snipped below:

```
firebase.database().ref(path_of_reference).once('value').then((snapshot)=> {
    userref.path_of_update.update({'myfield': update_field()});
}).catch((error) => {console.error(error)});
```

Using variants of this structure injected across multiple components, the app knows whether someone self-identified as Republican or Democrat, whether they liked or disliked a video, what videos they have watched and for how long, and if they have shared a video, what platform they used. For example, Firebase update functions are called when a user has swiped on a video or pressed the corresponding button. Furthermore, it will also retrieve the current play time of the video as the time that the user has watched the video and stores the id of the video along with the time associated with it into Firebase. At the same time, the list of the videos a user has watched is also be updated. From a React perspective, some class components had to be rewritten to functional components in order to be integrated with the npm packages. Furthermore, callback functions were used to extract lower-level component information such as the video player's current cursor location.

3.3.6 Personal Data Visualization

The personal data visualization was implemented using Apexcharts and Bootstrap. The main challenge was to crunch the user's data and format it such that the Apexcharts' components could render the data. However, the data manipulation was done on the recommendation system's side. The PWA only needed to make a GET request with the anonymous unique id as a parameter and the Axios npm package was used to help do that. The Bootstrap npm package was used to conditionally render the navigation bar of the personal data visualization pages in shades of blue or red, based on the user's affiliation (blue for Democrat/liberal/donkey, red for Republican/conservative/elephant).

3.3.7 Testing and Deployment

The npm packages dotenv, env-cmd and React Scripts were used to develop and test builds locally. An environment file (.env) was written for local development and testing. For production tests a staging branch was created where new features could be tested continuously. Both the staging and the public beta branch of the app were deployed using Heroku[16]. NecSys made enlightened.media.mit.edu a CNAME for the host of the public beta branch.

3.4 Recommendation System

The recommendation system serves a personalized playlist and personalized usage data to the *Enlightened* PWA. This is achieved by querying the *Enlightened* Firebase database and the *SuperGlue* MongoDB database and performing a set of operations to obtain a timeline of a user's bias, the average bias as well as the distribution of likes and dislikes across the news channels. The same method applies to calculate the usage statistics such as run streak, total time spent, number of news clips watched and average time per news clip.

The RecSys is a Python 3 Flask[17] application. It uses the pyrebase[18] package to connect to the Firebase database and the pymongo[19] package to connect to the MongoDB database. The database queries to the Firebase database are made by using the anonymous user id passed as a parameter in the GET request coming from the PWA. The database queries to the MongoDB database are made using the videoclip ids, as well as based on the current time. A cron job executes every five minutes to update the most recent list of video clips generated by *SuperGlue*. The cron job uses the native python cache module. Upon receiving the required data, RecSys manipulates the data using the numpy[20] package. To make the GET request from the

[16]https://www.heroku.com/
[17]https://flask.palletsprojects.com/en/1.1.x/
[18]https://pypi.org/project/Pyrebase/
[19]https://pypi.org/project/pymongo/
[20]https://numpy.org/

PWA axios npm package work on the RecSys side, the flask_cors[21] library was used. The python-dotenv[22] package was used to load environment variables used for testing during development. Similarly to the *Enlightened* PWA, RecSys is also deployed on Heroku and also has a staging branch and public beta branch. The RecSys supports four routes, one for acquiring the video playlist for the user, one for the user's average bias and bias timeline, one for the user's like and dislike distribution, and one for the user's app usage statistics:

```
@ app.route('/api/playlist', methods=['GET'])
@ app.route('/api/stats', methods=['GET'])
@ app.route('/api/likedislike', methods=['GET'])
@ app.route('/api/biastracker', methods=['GET'])
```

All data is served in a json format to the PWA, for example:

```
Playlist:
{
    [
        "https://s3.amazonaws.com/super-glue-do-not-delete/video/16147
        37411181.ce3f3268-67e6-4f06-b090-03272515587a.segment0.mp4",
        ...
    ]
}

Stats:
{
    days: 4,
    timespent: 243,
    videoswatched: 50,
    cliptime: 24
}
```

[21]https://pypi.org/project/Flask-Cors/1.10.3/
[22]https://pypi.org/project/python-dotenv/

```
Like / Dislike:
{
    likes: {
        CNN: 21,     // blue
        FOX: 5,     // red
        ABC: 14,    // purple
        BTV: 24,   // purple
        MSN: 22     // blue
    },
    dislikes: {
        CNN: 14,
        FOX: 11,
        ABC: 5,
        BTV: 19,
        MSN: 17
    }
}
BiasTracker:
{
    "avg": {
    "conservative": "50",
    "liberal": "50"
    },
    "timeline":
    [
        {
            "date": "2 April",
            "conservative": "50",
            "liberal": "50"
        }
    ]
}
```

Chapter 4

Experimentation and Results

This chapter describes the experiment setup, data collection process, and exhibits the results. The purpose of the experiment is to understand whether short-form news videos captivate people and whether they can open minds (i.e., inform and expand our perspective). The data collection process consists of a two-week study with two surveys and involves the usage of *Enlightened*. The survey results and *Enlightened* data are exhibited and visualized at the end of this chapter.

4.1 Experiment Setup and Metrics

Enlightened was built to serve as an interface that presents short-form news videos while tracking a user's interaction and gathering quantitative data. A two-week study was conducted along with an opening and closing survey to learn about the users' behavior with short-form news videos.

Whether short-form news videos captivate people can be measured by user engagement. The quantitative measurement of user engagement consists of the number of swipes per day, the total number of swipes, the total number of video watched, the time spent on each video, the total time spent on the app, the number of liked/disliked videos per day and the total number of liked/disliked videos. The qualitative measurement of user engagement consists of the answers to the following closing survey statements and question: "The app is fun and engaging", "The app is easy to use",

"Would you consider to keep using this app to inform yourself (outside of the study)?".

To understand whether short-form videos can inform and move people closer to shared reality, the quantitative data can be reused in conjunction with *SuperGlue* video metadata. For example, the source information of the news videos, the like and dislike ratio of news sources that align with someone's political inclination can be mapped out. Furthermore, the user's like and dislike distribution as well as the time they spend across all news channels can be analyzed. The qualitative measurement consists of the answers to the following closing survey statements: "The app keeps me informed and up to date", "The app shows me different perspectives", "I'm surprised by my own bias", "I'm surprised by my likes/dislikes distribution".

4.2 Data Collection

The study was advertised to all undergraduate and graduate dorms at MIT as well as the MIT Media Lab. Out of 274 anonymous users that have visited the website enlightened.media.mit.edu, 151 users have stayed on the website long enough for the Firebase database to register them. 64 users have self-identified as either liberal (51 users) or conservative (13 users) and all of them have watched at least one video. 35 users filled out the initial survey to participate in the study and 8 users completed the closing survey. The following subsections exhibit the qualitative data collected from the surveys as well as the quantitative data collected during the study period. The study period lasted two weeks, but not all users started at the same time.

4.3 Results

4.3.1 Initial Survey Data

Out of the 35 users that participated in the study, all of them were able to successfully install the app on their phone.

The majority get their news primarily from news websites or news apps (57%), followed by social media (34%), TV (3%), people (3%) and newspaper (3%).

Around half of them (51%, including 11% "news junkies") consume news on a daily basis, while roughly one third (32%) consumes news every other day. Only 14% don't care about news, and one person (3%) says it depends on the topic.

Two out of three users are U.S. citizens, and the majority of them are liberal (43%), followed by moderate (21%), independent (17%), progressive (4%) and libertarian (4%). Among non-U.S. citizens, the majority is independent (42%), followed by liberal (25%), moderate (17%) and progressive (17%). Unfortunately, no conservative user signed up for this study.

A majority of study participants are familiar with social media apps such as Facebook (86%), Instagram (83%), Snapchat (60%), Twitter (43%), TikTok (40%) and the swiping mechanism of Tinder (30%). Two out of three study participants are Gen Z and the remaining third are Millenials, with the exception of one Gen X.

4.3.2 Final Survey Data

Despite efforts to engage study participants to fill out the final survey, only 8 users completed it. The survey questions and statements are listed below in bold.

Would you consider to keep using this app to inform yourself (outside of the study)?

Five people would consider to make the app an occasional source of their news after the study, one person would even make it a primary source of their news. Two people wouldn't consider to keep using the app after the study because the interface wouldn't allow them to select the news or display a headline and because the choice of clips seemed strange.

The app keeps me informed and up to date

Two people strongly agreed, four agreed, and two disagreed.

The app is easy to use

One person strongly agreed, four agreed, one person was neutral and two disagreed.

The app is fun and engaging

Four people agreed, three people responded with neutral and one person disagreed.

The app shows me different perspectives

Six people agreed and two disagreed.

I'm surprised by my own bias

Three people agreed, two were neutral, one person disagreed and two people strongly disagreed.

I'm surprised by my own likes/dislikes distribution

Two people agreed, three were neutral, two disagreed and one strongly disagreed.

The app worked well on my device

The app worked well for seven out of eight people. Six users had iPhones, one user a Samsung Galaxy and another user a OnePlus 7T. Unfortunately, the app did not work well for the iPhone SE user.

Recommendations to improve the app

Every person had a recommendation to improve the app. One person wanted a 'neutral' button; some clips were just commercials or don't give enough context for them to decide to like/unlike it. Another user wanted a news topic filter. Two people wanted the app to remove commercials/ads. One person wanted a way to select the channel. Another wanted headlines and a way to choose the video to watch. One person wanted a non-rotation based feature to watch news.

App usage of final survey participants

The corresponding watch time per clip follows the following distribution:

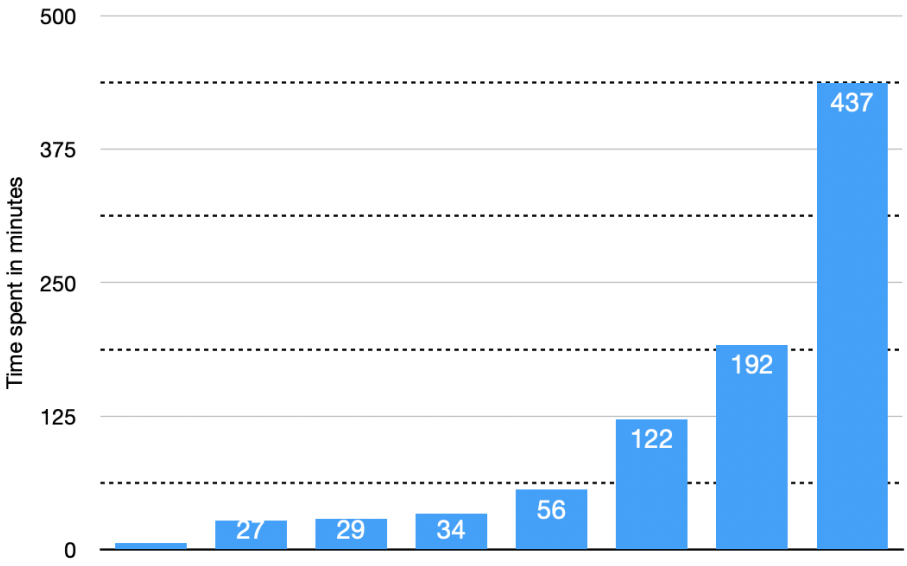

Figure 4-1: While most final survey participants spent around half an hour on the app, one person spent more than seven hours on the app. The x-axis is discrete and each value represents a person, sorted by the amount of time they have spent.

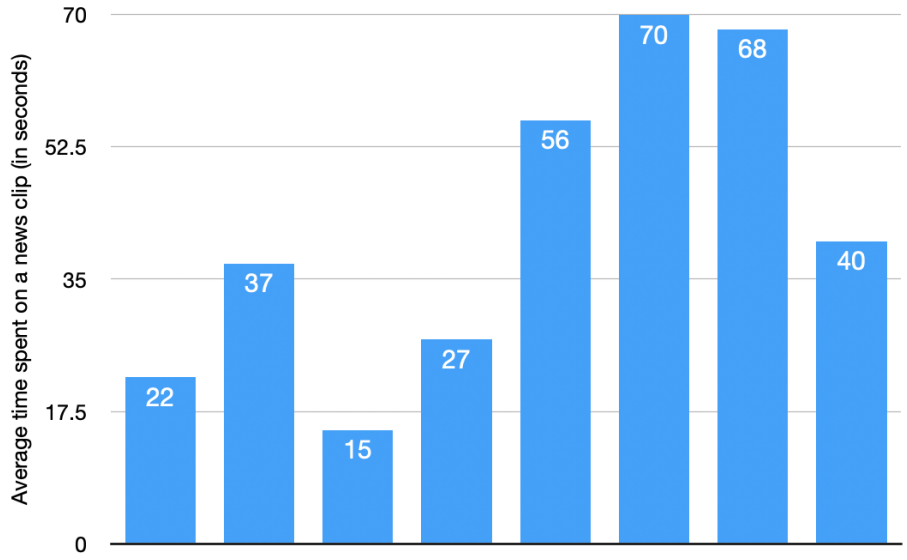

Figure 4-2: People who used the app longer tend to spend more time on a news clip. However, the person who has spent the most time on the app spends only 40 seconds on a news clip on average. Three out of four spend less than a minute on a news clip on average and everyone spent less than 70 seconds on a news clip on average.

4.3.3 Enlightened Usage Data

Due to low participation in the final survey, the Enlightened data from only eight users could be tied to the survey responses. However, a far greater number of users have visited the website and used the app in the period between April 8, 2021 and May 6, 2021. To reiterate, out of 274 anonymous users that have visited the website enlightened.media.mit.edu, 151 users have stayed on the website long enough for the Firebase database to register them. 64 users have self-identified as either liberal (51 users) or conservative (13 users) and all of them have watched at least one video. The results exhibited in this section focus on the usage data of these 64 users (subsequently referred to as *active* users) and are segmented into user engagement related data and mind-opening related data.

User Engagement Data

The average active user profile uses *Enlightened* on 2.2 days. However, the distribution of the number of days when the app was used shows a steep decay after one day.

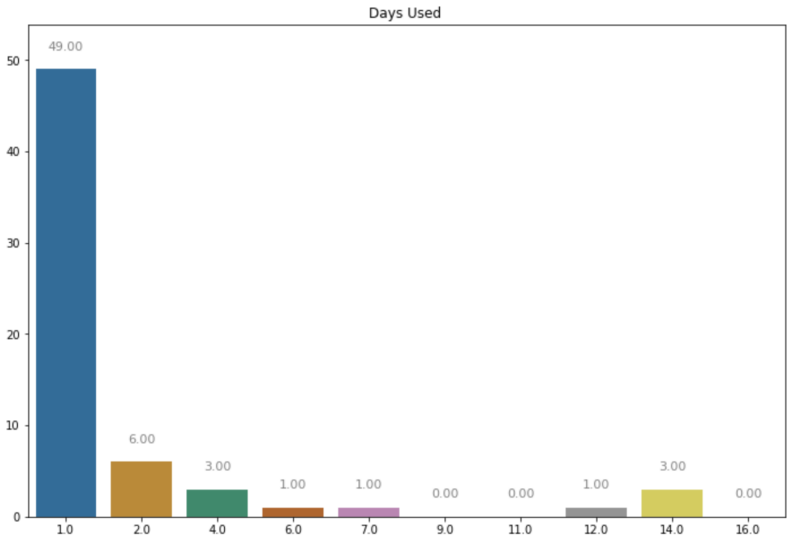

Figure 4-3: The vast majority of active users (49 out of 64) have only used the app on one single day. While most users used it for less than a week, there are four users that have followed the study diligently and used the app for 12 or 14 days, respectively. The x-axis represents the number of days a user has used the app. The y-axis represents the number of people that have used the app for x days.

The average number of total swipes per day (sum of all users' swipes on a particular day) lies at 74.04. On April 8, the public beta of *Enlightened* was launched. The period from April 8 to April 16 was when the app was advertised amongst my research group members as well as my close circle of friends. The distribution of the total number of swipes between April 8, 2021 and May 6, 2021 peaks from April 17 to April 20, the dates where the study was advertised to the MIT Media Lab community and all MIT dorms. The second minor peak at April 30 was the day when the closing survey was announced.

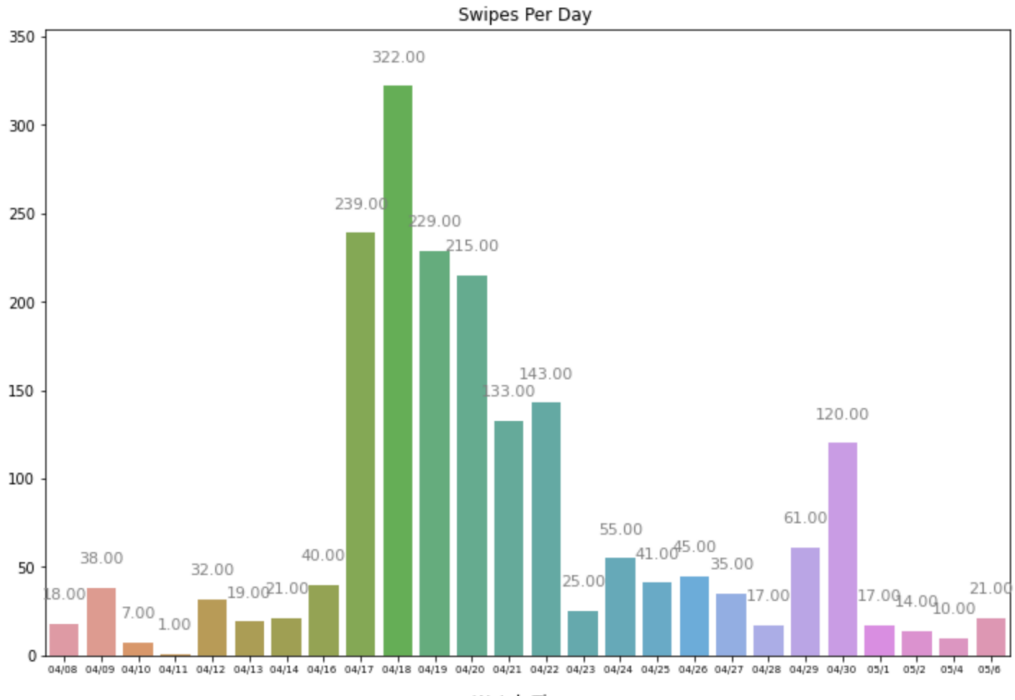

Figure 4-4: The majority of swipes occurred when the study was announced across the MIT Media Lab community and all MIT dorms. The x-axis represents the days the app was used by at least one user. The y-axis represents the total number of swipes that occurred on that day.

By aggregating the total number of swipes of a user over all days and grouping users that have swiped on the same amount of videos, one can learn that the median active user watches only seven videos in total, whereas the average active user swipes on a total of 20 videos. One user totaled 654 swipes and was considered an outlier. This data point was omitted when visualizing and calculating the average total swipes of an active user.

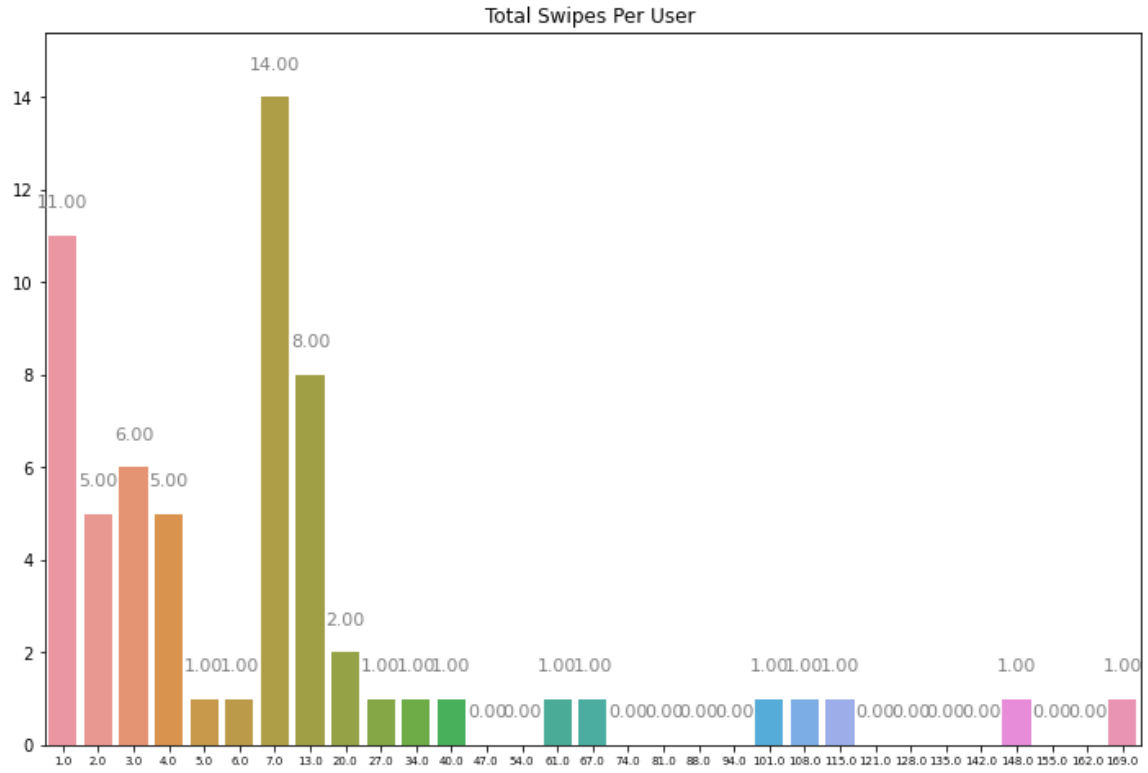

Figure 4-5: Two out of three active users swiped on seven videos in total or less. The x-axis represents the total number of swipes and the y-axis represents the number of users that totaled x swipes. For the purpose of this visualization, an outlier of one user with a total of 654 swipes was omitted in the graph.

On a daily basis, the median active user swipes on 6 videos per day on the days when the user is active. The average active user swipes on 7.8 videos on the days when they're active. However, the average is influenced by an outlier that swiped on 43.6 videos per active day. While it may seem obvious, it should be noted that the number of swipes correspond to the number of videos a user has watched.

Unfortunately, most users have spent less than five minutes on the app watching videos. The average total app usage time among active users is 6.8 minutes after removing four outliers (80 minutes, 122 minutes, 192 minutes and 437 minutes). The average total app usage time increases to 19.4 minutes if the outliers are included.

In total, 1850 news clips were watched with an average watch time of 40.0 seconds. The watch time for half of the news clips was under 30 seconds, with 500 clips that were only watched for less than 15 seconds.

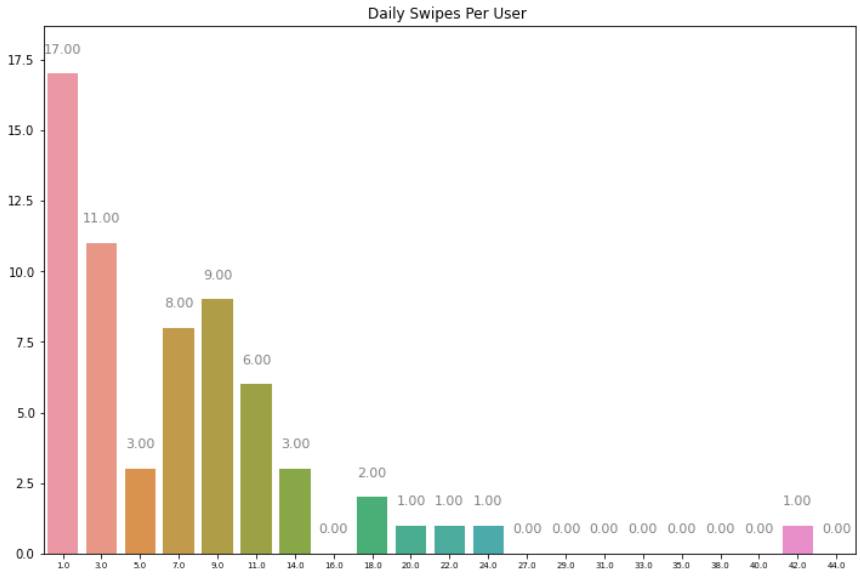

Figure 4-6: The majority of active users swipe less than six times per day. The x-axis represents the number of daily swipes and the y-axis represents the number of users that swipe on x videos per active day. The x-axis is bucketed in this chart. For example, 17 users swipe at least once and less than three times per active day.

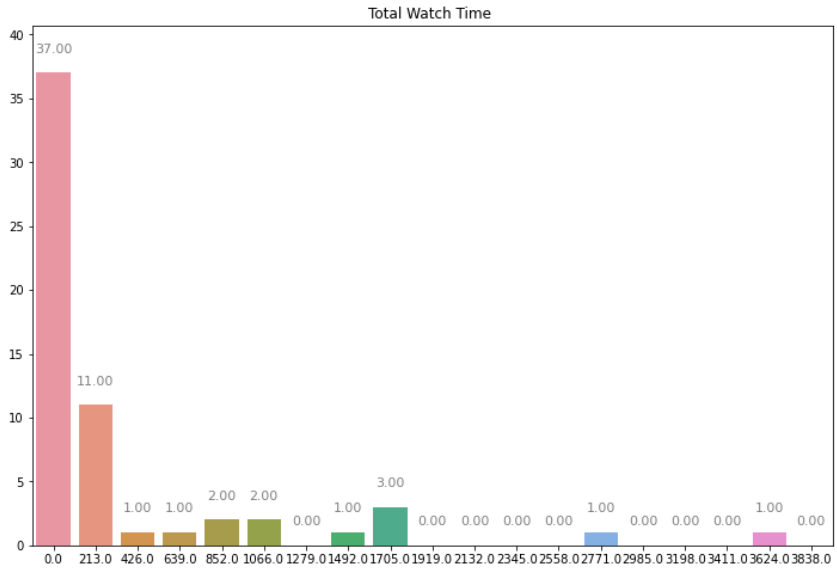

Figure 4-7: Three out of four people have spent less than seven minutes on the app watching videos. The x-axis represents the total watch time (in seconds) and the y-axis represents the number of users that spent a total of x seconds watching videos. The x-axis is also bucketed in this chart. For example, 11 users spent a total of at least 213 seconds and less than 426 seconds watching videos. For visualization purposes, four outliers (80 minutes, 122 minutes, 192 minutes and 437 minutes) were omitted.

A related statistic is the distribution of how much time one user has spent on a news clip on average. It is calculated by taking the ratio between the total watch time of the user and the number of clips watched by the user. The average time spent on a clip is 23.5 seconds after removing one outlier of 237 seconds. If the outlier is included, the average time spent on a clip increases to 27.3 seconds.

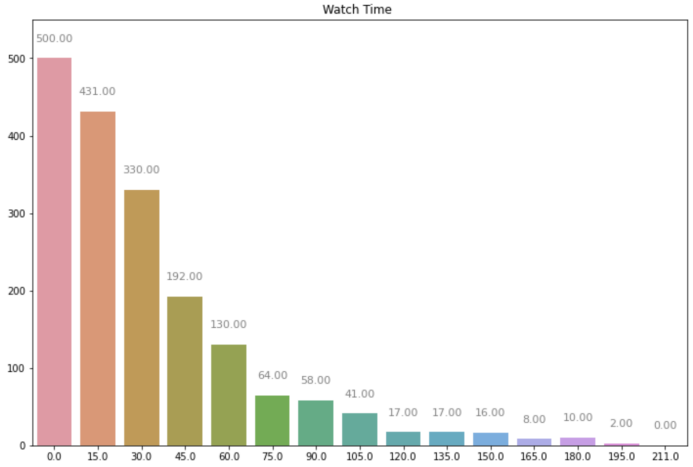

Figure 4-8: The watch time for around 4 out of 5 news clips was under a minute. The x-axis represents the number of seconds a video was watched. The y-axis represents the number of videos that had a watch time of x seconds. The x-axis is bucketed in this chart. For example, 192 video clips were watched for at least 45 seconds and less than one minute. 25 outliers were not shown in this graph.

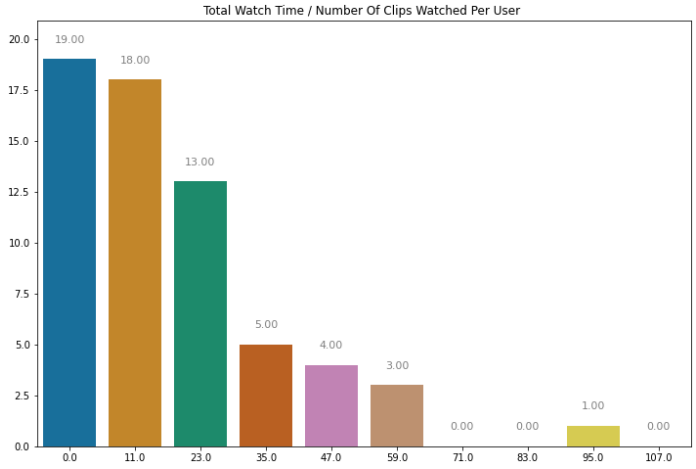

Figure 4-9: Around 3 out of 5 active users spend less than 23 seconds on a clip on average. The x-axis is bucketed and represents the average time spent on a news clip. The y-axis represents the number of people that have an average clip watch time x.

Mind-Opening Data

The average time spent on a video is around 40.0 seconds. However, the time spent differs across channels. On average, active users spent most their time on ABC news clips (46.6 seconds), followed by CNN (42.9 seconds), MSNBC (40.0 seconds), FOX (38.9 seconds) and BTV (31.7 seconds). In relative to the average, the relative delta correspond to +16.5% for ABC, +7.3% for CNN, +0% for MSNBC, -2.7% for FOX and -20.7% for BTV. The average time spent on a video differs for conservative and liberal users, as shown subsequently.

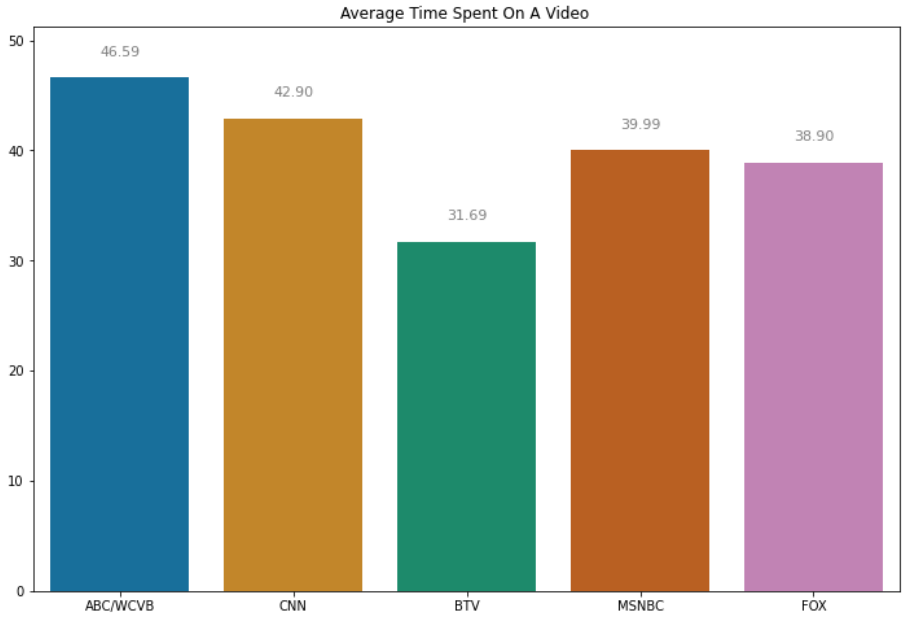

Figure 4-10: On average, active users spent most of their time on ABC news videos, followed by CNN, MSNBC, FOX and BTV content. The y-axis represents the average time spent on a video of a given channel x in seconds.

Active users who self-identified as conservative spent more time on a video on average, across all channels. On average, they spent most of their time on FOX (65.0 seconds), followed by ABC (64.2 seconds), CNN (57.4 seconds), MSNBC (42.4 seconds) and BTV (35.5 seconds). Conservative users have an average video watch time of 52.9 seconds, 15.1 seconds more than the average watch time across all active users. In relative to the conservative average, the relative delta corresponds to +22.9% for FOX, +21.4% for ABC, +8.5% for CNN, -20.0% for MSNBC and -33.2% for BTV.

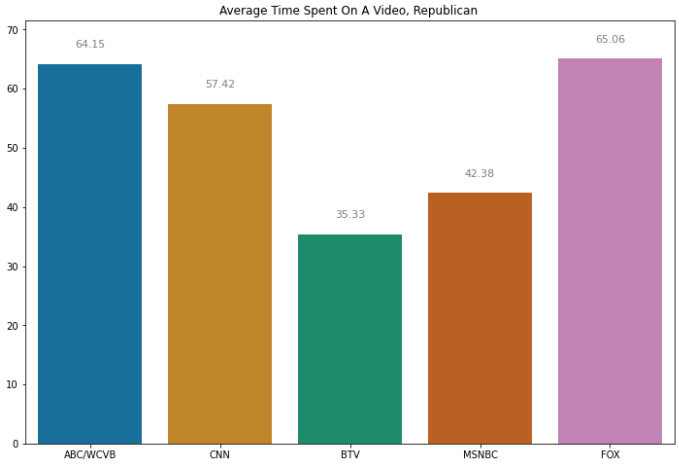

Figure 4-11: Conservative active users spent most of their time on FOX news videos, followed by ABC, CNN, MSNBC and BTV content. The y-axis represents the average time spent on a video of a given channel x in seconds.

Active users who self-identified as liberal spent less time on a video on average, across all channels. On average, they spent most of their time on ABC (41.6 seconds), followed by CNN (40.8 seconds), MSNBC (39.9 seconds), FOX (36.8 seconds) and BTV (30.9 seconds). Liberal users have an average video watch time of 38.0 seconds, 2 seconds less than the average watch time across all active users. In relative to the liberal average, the relative delta corresponds to +9.5% for ABC, +7.3% for CNN, +5.0% for MSNBC, -3.2% for FOX and -18.7% for BTV.

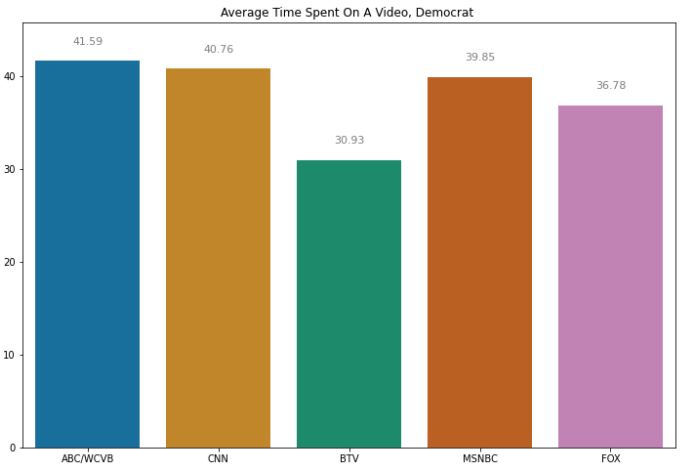

Figure 4-12: Liberal active users spent most of their time on ABC news videos, followed by CNN, MSNBC, FOX and BTV content. The y-axis represents the average time spent on a video of a given channel x in seconds.

The average ratio between likes and dislikes on a video across news channels of all active users is above one. ABC has the highest like/dislike ratio (4.57), followed by BTV (2.76), CNN (2.42), MSNBC (1.44) and FOX (1.20). These numbers are overall averages and are not weighted by the amount of liberals and conservatives that contributed to the likes and dislikes. The average overall channel like/dislike ratio is around 2.5. In relative to the average, the relative delta corresponds to +82.8% for ABC, +10.4% for BTV, -3.2% for CNN, -42.4% for MSNBC and -52.0 % for FOX.

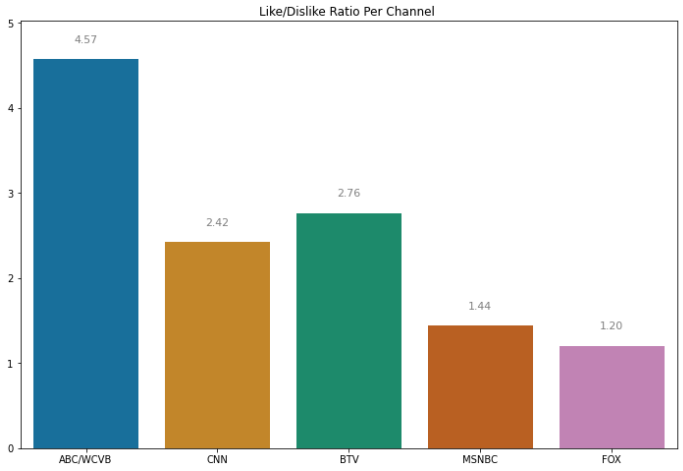

Figure 4-13: On average, active users' like/dislike ratio is highest for ABC, followed by BTV, CNN, MSNBC and FOX. The y-axis represents the average like/dislike ratio of a news channel on the x-axis.

Active users who self-identified as conservative have an average like/dislike ratio of 5.97, much higher than the average like/dislike ratio of 2.5. ABC has the highest ratio (10.8), followed by CNN (8.0), BTV (7.2), MSNBC (2.4) and FOX (1.43). In relative to the conservative average, the relative delta corresponds to +80.9% for ABC, +34.0% for CNN, +20.6% for BTV, -59.8% for MSNBC and -76.0% for FOX.

Active users who self-identified as liberal have an average like/dislike ratio of 1.62, much lower than the average like/dislike ratio of 2.5. ABC has the highest ratio (2.62), followed by BTV (1.59), CNN (1.49), MSNBC (1.25) and FOX (1.14). In relative to the liberal average, the relative delta corresponds to +61.7% for ABC, -1.9% for BTV, -8.0% for CNN, -22.8% for MSNBC and -29.6% for FOX.

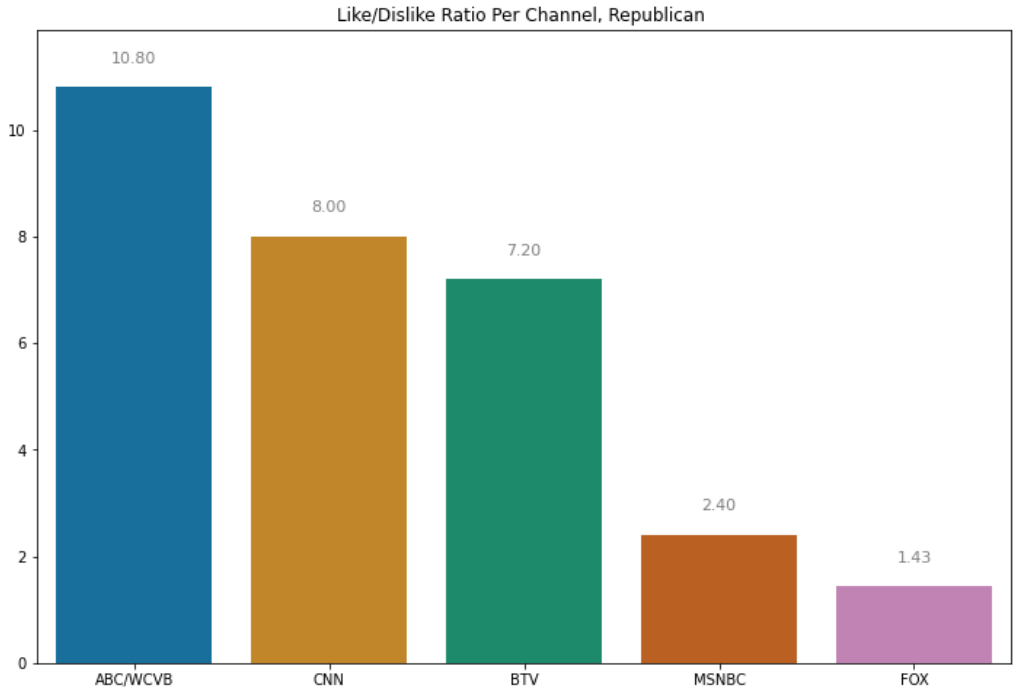

Figure 4-14: On average, conservative active users' like/dislike ratio is highest for ABC, followed by CNN, BTV, MSNBC and FOX. The y-axis represents the average like/dislike ratio of a news channel on the x-axis.

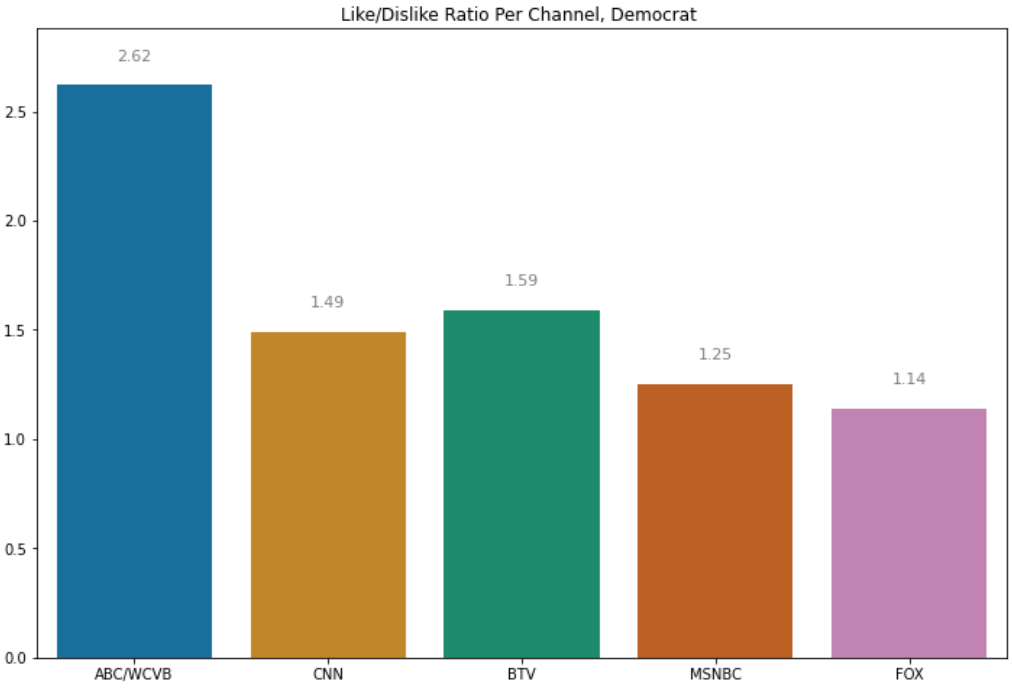

Figure 4-15: On average, liberal active users' like/dislike ratio is highest for ABC, followed by BTV, CNN, MSNBC and FOX. The y-axis represents the average like/dislike ratio of a news channel on the x-axis.

Chapter 5

Discussion

The final chapter is dedicated to answering the main research questions of this thesis: Can short-form news videos captivate people and can they open minds, that is, inform and expand our perspective? Each question is addressed in a separate section using the data collected and exhibited in the previous chapter. Furthermore, I will reflect on the ethical implications on this work, outline future work, present a set of design recommendations for *Enlightened* before concluding this thesis with a highlight of my key contributions.

5.1 Can short-form news videos captivate people?

The final survey data shows that the majority of users think that *Enlightened* is fun and engaging, easy to use, and would make the app an occasional source of their news after the study. One person even indicated that they would make *Enlightened* a primary source of their news. Most final survey participants spent around half an hour on the app over the period of two weeks, and one person spent more than seven hours on the app. However, since only 8 out of 35 study participants completed the final survey, the final survey data needs to be taken with a grain of salt. Unfortunately, it is not clear whether the study participants dropped out because of low financial incentive or external factors such as commitments outside of the study or whether the experience of *Enlightened* and short-form news videos was poor. One person indicated

in the final survey that they wouldn't continue to use the app after the study because the *Enlightened* interface wouldn't allow them to select the news or display a headline. However, that person indicated in the inital survey that they have not used TikTok before. Another person - also unfamiliar with TikTok - indicated that they wouldn't continue to use the app because the selection of clips seemed strange.

The quantitative measurements of user engagement suggest that the vast majority of people that used *Enlightened* did not find it engaging, with a handful exceptions. This observation is supported by the fact that the average active user used *Enlightened* on 2.2 days and that the vast majority of active users (49 out of 64) only used the app on one single day. While most users used it for less than a week, there are four users that have followed the study diligently and used the app for 12 or 14 days. A similar trend was observed in the distributions of swipes per day - it peaked when the study was advertised, but rapidly declined after four days. Furthermore, the median active user only watched seven videos in total and most users spent less than five minutes on the app. However, one user totaled 654 swipes and spent a total of 437 minutes on the app over the course of two weeks.

My interpretation of the qualitative and quantitative data is that there is supporting evidence that short-form news videos has the potential to captivate people, but the current interface *Enlightened* and the short-form news clips seem to be a turn-off for a majority of people, despite the evidence that the majority of final survey participants indicating that *Enlightened* is fun and engaging, easy to use, and would make the app an occasional source of their news after the study. From an interface perspective, we discover that the watch time for around 4 out of 5 news clips was under a minute and that 3 out of 5 active users spend less than 23 seconds on a clip on average. Furthermore, we learn that there is a demand to have commercials removed as well as giving users more agency to control what they see by adding a topic and news channel filter, and including contextual information such as headlines or thumbnails. This valuable feedback can be used to improve *Enlightened* and make it more accessible to a broader audience. However, it remains to be seen whether an improved and feature-rich interface can captivate people with short-form news videos.

5.2 Can short-form news videos open minds?

The final survey data indicate that the majority of users think that the app keeps them informed, up to date and show them different perspectives. On the matter of whether they were surprised by their own bias and their distribution of likes and dislikes, the results do not suggest a strong answer. As mentioned in the previous section, the final survey data needs to be taken with a grain of salt because only 8 out of 35 study participants completed the final survey.

The average watch times of *Enlightened* news clips suggest that short-form news videos can open minds. For example, both conservative and liberal users spent most of their time on ABC news videos, a source considered to be trusted by both conservatives and liberals [8]. Furthermore, the average watch time of sources representing an opposing view does not differ more than 20%. For liberal users, the average watch time of a Fox news video is only one second shorter than a MSNBC video and four seconds shorter than a CNN video. For conservative users, the average watch time of a CNN video is only 7.5 seconds shorter than the average watch time of a FOX news video. Surprisingly, conservative users spent more time on liberal sources such as CNN and MSNBC than liberal users. However, this might have been due to the small sample size of conservative users (13 users).

The average ratio of likes and dislikes across news channels also suggest that short-form videos have the potential to open minds. For example, the like to dislike ratio is greater than one for all news sources. This is surprising because one would expect that people would dislike news that oppose one's view. Furthermore, the like to dislike ratio of ABC news videos is highest for both liberal and conservative users. Interestingly, for conservative users, CNN had the second highest like/dislike ratio and FOX had the lowest ratio. This is unexpected and might be as a result of a small sample size of conservative users.

My interpretation of the results is that both the survey data and *Enlightened* usage data strongly indicate that masked short-form news videos can open minds. The majority of final survey participants think that the app keeps them informed, up

to date and expose them to different perspectives. Furthermore, both conservative and liberal users spent most of their time on ABC news videos whose like/dislike ratio was also highest. Moreover, the average watch time of sources representing an opposing view does not differ more than 20%, and only differs by four seconds for liberal users.

However, due to the small size of the study, especially in terms of people who completed the post-survey and used the app for more than one day, I refrain myself from concluding whether short-form news videos can open minds for both liberal and conservative users without conducting a larger-scale follow-up experiment. That being said, the data collected in this thesis suggest that *Enlightened* and short-form news videos show promise in opening people's minds.

5.3 Reflection on Ethical Implications

While I do not see immediate dangers or harmful implications of this work in its current state, variations of this work might cause potential harm.

For example, the interface of *Enlightened* can be repurposed to increase the effectiveness of propaganda by masking the news source, since the results of this thesis indicated that the average watch time of sources representing an opposing view did not differ by more than 20%.

From an interface perspective, I'm not sure if short-form news videos will contribute to shortening people's attention span and have an adverse effect on people's information retention and causing them to become more reliant on their biases and instincts when assessing news videos rather than involve conscious thought processes. However, if short-form news videos become the norm, there are also benefits. For example, content creators and news outlets have to really think hard about creating meaningful content - however, the danger there might be a sensationalization of the content, similar to contemporary headlines of news articles on social media.

From an engineering standpoint, the current recommendation system was only serving the latest news videos from the news outlets. However, if infused with machine

learning techniques such as reinforcement learning, an objective function maximizing user engagement and monetization could be used to add fuel to the fire of partisanship. However, machine learning could also help restore the common ground by choosing an objective function that maximizes exposure to views other than their own without compromising user engagement significantly.

5.4 Future Work

A major limitation of the results of this thesis is the small sample size of final survey participants as well as the small number of conservative users. Whether this was a result of the limitations of features of the *Enlightened* app is unclear, but I tend to believe so. To find clarity and to strengthen the results of this thesis, the wish list of features given by final survey participants can be used in conjunction with user experience studies to improve the app.

From an engineering standpoint, as mentioned in the previous section, machine learning could be used to improve the recommendation system by choosing an objective function that maximizes exposure to views other than their own without compromising user engagement significantly. Furthermore, a better advertisement detection system could be implemented to remove commercials more effectively. Currently, the news clips were summarized purely based on natural language processing. However, sound computing and computer vision techniques could also be used to improve the relevance of summarized content.

While *Enlightened* was built to address the fragmented media structure and polarization of the United States of America, the concept and interface of *Enlightened* can be applied to other polarized regions of the world that have a significantly different cultural, political, societal and media landscape to study the effectiveness of this medium in restoring *common ground*. And maybe, hopefully, we will be able to create a world that is more empathetic, inclusive, tolerant and understanding of each other.

5.5 Design Recommendations

To critically discuss and recommend on how to improve *Enlightened*, I first reiterate the statements of the final survey participants on how they would improve the app before adding my comments on how to address these recommendations.

As stated in the experiments and results chapter, everyone had a recommendation to improve the app. One person wanted a 'neutral' button; some clips were just commercials or don't give enough context for them to decide to like/unlike it. Another user wanted a news topic filter. Two people wanted the app to remove commercials/ads. One person wanted a way to select the channel. Another wanted headlines and a way to choose the video to watch. One person wanted a non-rotation based feature to watch news. Furthermore, two people (both unfamiliar with TikTok) declared that they wouldn't continue to use the app after the study because the *Enlightened* interface wouldn't allow them to select the news or display a headline.

Given these statements, I conclude that the majority of people desire more selection agency and I recommend that this should be addressed to improve *Enlightened*. This can be addressed by adding topic filters or a list of topics on the 'main' screen where the user is currently prompted to rotate their phone. Upon selecting a topic, a list of thumbnails accompanied by headlines (similar to news mobile apps) can be presented to the user. After clicking on a thumbnail, the user could be brought into "Tinder" mode.

Furthermore, I recommend that the commercials should be removed. The issue with commercials can be resolved by developing better tools (with or without machine learning) to automatically detect commercials or by including humans in the loop that flags clips with advertisements. An advantage of the latter approach is that the flagged clips can subsequently be used to train a classifier to more effectively automate the process. An intermediate solution to this problem is to add a neutral or ad flagging button in "Tinder" mode and let users know that the effectiveness of the advertisement removal algorithm is not at 100% yet.

Although a non-rotation based feature of watching news was requested, I do not

recommend to use Google's autoflip to cut the news videos vertically, as mentioned in the design chapter. From an aesthetic point of view, I also do not recommend to present the masked news videos (that appear relatively thin since the lower third is removed) in portrait mode.

I also do not recommend providing people with the option to select the news channel as my hypothesis is that it might defeat the purpose of the app of exposing users to views different other than their own (especially since the news source has been masked). However, without evidence supporting this claim, it might be instructive to test the hypothesis. In any case, the relationship of a user's agency and level of control over what they consume and their effect on whether they help opening users' minds is an interesting relationship to explore.

5.6 Key Contributions

In this final section, I am highlighting my key contributions:

- I motivated the exploration of the use of short-form videos as a medium to address the fragmented and polarized U.S. media landscape and as a means to restore the *common ground* amongst ideologically distant users.

- I exhibited work related to opening people's minds, motivated a mobile-first design, and used the inspiration from contemporary mobile U.S. news apps as well as a selection of mobile non-news apps to design *Enlightened*.

- I implemented *Enlightened* and used it to conduct a two-week user study consisting of two surveys and the daily usage of *Enlightened*.

- I evaluated the results and showed that the data suggest that despite the small size of the user study and the limited features of *Enlightened*, masked short-form news videos show great promise in engaging users and opening their minds. However, a large-scale follow-up study with an improved *Enlightened* needs to be conducted to ascertain whether short-form news videos can open minds.

Bibliography

[1] David Anderton-Yang. Countering source bias in news. Master's thesis, Massachusetts Institute of Technology, August 2019.

[2] Patricia Aufderheide. Cable television and the public interest. *Journal of Communication*, 42(1):52–65, January 1992.

[3] Federico Barrios, Federico López, Luis Argerich, and Rosa Wachenchauzer. Variations of the similarity function of textrank for automated summarization. *arXiv preprint arXiv:1602.03606*, 2016.

[4] Stanley M. Besen and Robert W. Crandall. The deregulation of cable television. https://scholarship.law.duke.edu/cgi/viewcontent.cgi?article=3609&context=lcp, 1981. [Accessed 21-May-2021].

[5] Levi Boxell, Matthew Gentzkow, and Jesse M Shapiro. Is the internet causing political polarization? evidence from demographics. Technical report, National Bureau of Economic Research, 2017.

[6] Daniel Brown. The death of vine, and the volatile nature of new media. *Researching Sociology*, 2017.

[7] Pew Research Center. Political Polarization in the American Public. Available: http://www.people-press.org/2014/06/12/political-polarization-in-the-american-public/, June 2014. [Accessed 21-May-2021].

[8] Pew Research Center. Where News Audiences Fit on the Political Spectrum. Available: https://www.journalism.org/interactives/media-polarization/table/trust/, October 2014. [Accessed 21-May-2021].

[9] Pew Research Center. Political Polarization in the American Public. Available: https://www.people-press.org/2017/10/05/the-partisan-divide-on-political-values-grows-even-wider/, October 2017. [Accessed 21-May-2021].

[10] Josh Constine. Instagram drops IGTV button, but only 1% downloaded the app. Available: https://techcrunch.com/2020/01/18/igtv-button-gone/, January 2020. [Accessed 21-May-2021].

[11] Aaron Zitner Danny Dougherty. Democrats and Republicans Aren't Just Divided. They Live in Different Worlds. Available: https://www.wsj.com/graphics/red-economy-blue-economy/, September 2019. [Accessed 21-May-2021].

[12] Patrick Flemming. Wanna see something funny? fast laughs is here. Available: https://about.netflix.com/en/news/fast-laughs-is-here, 2021. [Accessed 21-May-2021].

[13] James G. Gimpel and Iris S. Hui. Seeking politically compatible neighbors? the role of neighborhood partisan composition in residential sorting. *Political Geography*, 48:130–142, 2015.

[14] Albert H. Hastorf and Hadley Cantril. They saw a game; a case study. *The Journal of Abnormal and Social Psychology*, 49(1):129–134, 1954.

[15] Thomas W Hazlett and David W Sosa. Was the fairness doctrine a "chilling effect"? evidence from the postderegulation radio market. *The Journal of Legal Studies*, 26(1):279–301, 1997.

[16] Daniel J Hopkins. *The increasingly United States: How and why American political behavior nationalized*. University of Chicago Press, 2018.

[17] Harold Hotelling. Stability in competition. In *The collected economics articles of Harold Hotelling*, pages 50–63. Springer, 1990.

[18] Instagram. Introducing Instagram Reels. Available: https://about.instagram.com/blog/announcements/introducing-instagram-reels-announcement, August 2020. [Accessed 21-May-2021].

[19] Chris Jaffe. Building YouTube Shorts, a new way to watch & create on YouTube. Available: https://blog.youtube/news-and-events/building-youtube-shorts/, September 2020. [Accessed 21-May-2021].

[20] Jason Jay. How to keep conversation alive in a polarized world | Jason Jay | TEDxBocaRaton. Available: https://www.youtube.com/watch?v=giBLRbVC0p4, 2018. [Accessed 21-May-2021].

[21] Mark Jurkowitz, Amy Mitchell, Elisa Shearer, and Mason Walker. Us media polarization and the 2020 election: A nation divided. *Pew Research Center*, 24, 2020.

[22] Jill Lepore. The last time democracy almost died. *The New Yorker*, 2020.

[23] Walter Cronkite Memorial. Walter Cronkite Memorial - About Walter Cronkite. Available: https://wcm.missouriwestern.edu/about-walter-cronkite/. [Accessed 21-May-2021].

[24] Amy Mitchell. Americans still prefer watching to reading the news – and mostly still through television. Available: https://www.journalism.org/2018/12/03/americans-still-prefer-watching-to-reading-the-news-and-mostly-still-through-television/, 2018. [Accessed 21-May-2021].

[25] Gordon Pennycook and David G. Rand. Fighting misinformation on social media using crowdsourced judgments of news source quality. *Proceedings of the National Academy of Sciences*, 116(7):2521–2526, 2019.

[26] Politico. Clinton: MSNBC 'our version of Fox'. Available: https://www.politico.com/story/2012/01/clinton-msnbc-is-our-version-of-fox-071601, January 2012. [Accessed 21-May-2021].

[27] Kathleen Ann Ruane. Fairness doctrine: History and constitutional issues. *J. Curr. Issues Crime Law Law Enforc*, 2(1):75–89, 2009.

[28] Jasmin Rubinovitz and Andrew Benjamin Lippman. Panorama. Available: https://www.media.mit.edu/projects/panorama/overview/, 2018. [Accessed 21-May-2021].

[29] Elisa Shearer. More than eight-in-ten americans get news from digital devices. Available: https://www.pewresearch.org/fact-tank/2021/01/12/more-than-eight-in-ten-americans-get-news-from-digital-devices/, 2021. [Accessed 21-May-2021].

[30] Statista. Leading cable networks in the United States in December 2020 by number of total day viewers. Available: https://www.statista.com/statistics/347040/cable-networks-viewers-usa/, December 2020. [Accessed 21-May-2021].

[31] Chenhao Tan, Vlad Niculae, Cristian Danescu-Niculescu-Mizil, and Lillian Lee. Winning arguments: Interaction dynamics and persuasion strategies in good-faith online discussions. In *Proceedings of the 25th international conference on world wide web*, pages 613–624, 2016.

[32] Mason Walker. Americans favor mobile devices over desktops and laptops for getting news. Available: https://www.pewresearch.org/fact-tank/2019/11/19/americans-favor-mobile-devices-over-desktops-and-laptops-for-getting-news/, 2019. [Accessed 21-May-2021].