UNDERSTANDING AND COMBATING FILTER BUBBLES IN NEWS RECOMMENDER SYSTEMS

BY

PING LIU

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Computer Science
in the Graduate College of the
Illinois Institute of Technology

Approved _____
Adviser

Mustafa Bilgic

Chicago, Illinois
May 2022

ACKNOWLEDGEMENT

I want to thank my thesis advisor, Dr. Mustafa Bilgic, for recruiting me as his Ph.D. student, guiding my study with unlimited patience, providing me with numerous research opportunities, and supporting me with a research assistantship. I thank Dr. Aron Culotta for introducing me to my first Ph.D. research project, supporting my first conference travel and presentation, and collaborating on my thesis work with me. I am incredibly grateful and proud to work with Dr. Bilgic and Dr. Culotta, both are excellent researchers and great educators.

I would like to thank Dr. Matthew A. Shapiro, Dr. Kai Shu, and Dr. Cynthia Hood (thesis committee) for agreeing to become my thesis committee members, taking valuable time to read and comment on my thesis, and always responding to my requests within minutes. I appreciate the help and inspiration from my collaborators, Dr. Libby Hemphill, Dr. Jiaqi Yan, Karthik Shivaram, and Joshua Guberman.

Special thanks go to my friends and classmates at IIT, Xin Liu, Jiaqi Yan, Ruo Yang, Yao Kang, for all your accompany when we played, studied, collaborated, and prepared for job interviews together. I would like to give my thanks to my lab mates, Ruo Yang, Juanyan Wang, Jawahar Panchal, Manali Sharma, and Caner Komurlu, in the past six years for their companionship, support, and feedback.

I would like to thank my parents, Baotang Liu and Ji Zhang, for their understanding and financial support for my master's study - during which I figured out my interests in the area of machine learning. Last but not least, many thanks to my wife Dr. Xi Huo, for being my primary reason to come to the U.S. and for consistently motivating and encouraging me to pursue a career in computer science.

This material is based upon work supported by the National Science Foundation under grants #1350337 and #1927407.

AUTHORSHIP STATEMENT

I, Ping Liu, attest that the work in this thesis is substantially my own.

In accordance with the disciplinary norm of Science and Engineering (see IIT Faculty Handbook, Appendix S), the following collaborations occurred in the thesis:

Dr. Mustafa Bilgic advised the whole thesis work as my Ph.D. advisor.

Dr. Mustafa Bilgic, of Illinois Institute of Technology, Dr. Aron Culotta, of Tulane University, and Dr. Matthew A. Shapiro, of Illinois Institute of Technology, acquired the NSF funding of the project, *Understanding the Relationship between Algorithmic Transparency and Filter Bubbles in Online Media*, in 2019. They guided and advised all results in this thesis, including the design of experiments and interpretation of the outcomes.

Karthik Shivaram, of Tulane University, conducted predictive labeling process for the dataset discussed in Chapter 3, and visualized the main results of the study in Chapter 4.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Algorithmic personalization of news and social media content aims to improve user experience. However, there is evidence that this filtering can have the unintended side effect of creating homogeneous "filter bubbles" in which users are over-exposed to ideas that conform with their pre-existing perceptions and beliefs. In this thesis, I investigate this phenomenon in political news recommendation algorithms, which have important implications for civil discourse.

I first collect and curate a collection of over 900K news articles from over 40 sources. The dataset was annotated in the topic and partisan leaning dimensions by conducting an initial pilot study and later via Amazon Mturk. This dataset is studied and used consistently throughout this thesis.

In the first part of the thesis, I conduct simulation studies to investigate how different algorithmic strategies affect filter bubble formation. Drawing on Pew studies of political typologies, we identify heterogeneous effects based on the user's pre-existing preferences. For example, I find that i) users with more extreme preferences are shown less diverse content but have higher click-through rates than users with less extreme preferences, ii) content-based and collaborative-filtering recommenders result in markedly different filter bubbles, and iii) when users have divergent views on different topics, recommenders tend to have a homogenization effect.

Secondly, I conduct a content analysis of the news to understand language usage among and across various topics and political stances. I examine words and phrases used by the liberal media and by the conservative media on each topic. I first study what differentiates the liberal media from the conservative media on each topic. I then study common phrases that are used by the liberals and the conservatives on different topics. For example, I examine which phrases are shared by the liberal articles on guns and conservative articles on abortion. Finally, I compare and visu-

alize these words using different clustering algorithms and supervised classification methods.

In the last chapter, I conduct an extensive user study to find possible solutions to combat the filter bubbles in the political news recommender systems. I designed a self-contained website that enables a content-based news recommender system and indexed 40,000 U.S. political articles. I recruited over 800 U.S. participants from Amazon Mechanical Turk (approved by IRB). The qualified participants are split into control and treatment groups. The users in the treatment group are provided transparency and interaction mechanisms, which grant them more control over the recommendations. Our results show that providing interaction and transparency a) increases click-through rates, b) has the potential to reduce the filter bubbles, and c) raises more awareness about filter bubbles.

CHAPTER 1

INTRODUCTION

In this chapter, I first address the motivation for understanding and combating filter bubbles in news recommender systems and then outline the remaining chapters of the thesis.

1.1 Background and Motivation

Machine learning is a subfield of artificial intelligence where an agent learns to understand and interpret its environment and makes predictions for future actions. Machine learning can be broadly categorized into three types: supervised learning, unsupervised learning, and reinforcement learning. Traditional supervised learning methods learn to make a prediction using training data samples. The learning process usually involves choosing an appropriate model, defining an objective function, and optimizing it to estimate the model structure and parameters.

The use of machine learning algorithms is currently at an unprecedented scale. There are many reasons for the increased pace of adoption of machine learning. Internet, social networks, smartphones, IoT devices, and many others have made the collection of data a lot easier [1, 2, 3]. The reduction of costs in hard disk and cloud storage have made data storage more affordable. The increased speed of CPUs, increased availability of GPU programming, increased RAM sizes, and advances in distributed computing have made the analysis of large datasets more efficient. Increased availability of open-source machine learning and data analysis packages has made the application of machine learning algorithms more accessible. Together with many success stories, all of these have significantly promoted the use of machine learning in both personal space and industrial applications. Examples include speech recogni-

tion [4, 5], face detection [6, 7], recommender systems (e.g., movies, songs, products, *etc.*) [8, 9], question-answering systems including search engines like Google and Bing [10, 11], medical diagnosis [12, 13], autonomous driving [14, 15], fraud detection [16, 17], and so on.

Machine learning algorithms have also been used to provide personalized curation of news, blogs, and social media posts to improve user experience and increase user engagement. However, there is mounting evidence that this automated filtering leads to "filter bubbles," in which users are over-exposed to ideas that conform with their pre-existing perceptions and beliefs, prompting intellectual isolation [18]. In this thesis, I investigate this phenomenon in the context of political news recommendation algorithms, which can have significant and often confounding effects with regard to how people perceive consensus and mobilize around partisan and policy issues [19, 20, 21, 22, 23].

Prior work typically simplifies the problem space by reducing user preferences to a single partisan score (e.g., strong liberal to strong conservative) [24]. However, this ignores the nuanced and varied preferences users have by topic. For example, a user may have conservative views on abortion but liberal views on health care. In this work, I am interested in understanding how a user's preferences influence the behavior of recommendation algorithms and the diversity of news content to which they are exposed. To achieve so, I propose and design several recommender system models and analyze the findings between the interaction of different political typologies and filter bubbles.

To start analyzing the filter bubble effects in the news recommender system, I first collect over 900K news articles from 41 different news sources. By external sources and Amazon Mturk's annotation, all the articles are labeled into different topics, and partisan leans. Then, I conduct three studies: 1) extensive simulations

of several recommender systems and different political typologies, 2) content analysis of the news to study the choice of words by different political ideologies and their potential effect on content-based news recommenders, and 3) user studies to study the effect of transparency and interaction on filter bubbles. The contributions of this thesis are as follows:

- I collected and labeled over 900K U.S. news articles across fourteen political topics and five ideological grades.
- In our simulation work, I conducted simulation studies to investigate how different algorithmic strategies affect filter bubble formation. By drawing on Pew studies to sample different political typologies, I investigated heterogeneous effects based on users' pre-existing preferences.
- In our content analysis work, I investigated how liberal and conservative media frame their news content on different topics. I studied the words that distinguish the liberals from the conservatives per topic. I also studied the words that are similar in context but used by different political ideologies.
- In our user study, I designed a news recommender system website, which provides interactive tools to let members adjust users' preferences to change the ranking output. Our user study results showed that introducing transparency and interaction tools for news recommendations has the potential to alleviate the filter bubble effects, and help users understand how the filter bubbles are formed and how the system works.

1.2 Thesis Outline

1.2.1 Related Work. In Chapter 2, I review and discuss the related work on the definition and different types of recommender systems. I mainly focus on previous

studies about filter bubbles in social media and the news recommender system. In particular, I discuss the content analysis research methods which were used to analyze how people understand situations and activities arising from political news, such as automated frame analysis.

1.2.2 Data Collection and Annotation. In Chapter 3, I first describe our data collection methodology. I collected a news collection of over 900K articles. I describe how the dataset was collected, annotated, and sampled. I also describe the pipeline I built and provide statistical analysis of the data in detail.

1.2.3 Simulation in News Recommender System. In Chapter 4, I first explain how to generate different political typologies for simulation purposes. Then I provide details on setting up the news recommender simulation in our experiments. The simulation results, such as comparing the personalized recommenders with an oracle recommender, are analyzed and discussed. I also show a variety of measurements of filter bubbles I developed for different algorithms and political typologies.

1.2.4 Content Analysis in News Articles. Motivated by the findings of the previous chapter, where I showed that the content-based recommender system inevitably generated biases, such as polarized use of language, I conduct a content analysis to study the different language constructs used by different political typologies. In Chapter 5, I investigate how the language is used in a single topic and different topics from different partisan sources and the juxtaposition of words that are similar in context but used by different political ideologies.

1.2.5 User Study in News Recommender System. In Chapter 6, I conduct a user study on filter bubbles of the news recommender system. I built and indexed 40,000 labeled articles into a content-based recommender website. Over 800 Amazon Mechanical Turkers participated in a pre-survey, and over 100 of them participated in

the full study. I discuss our analysis of the effects of transparency and user-controls in the recommender system and present the major findings.

1.2.6 Conclusion. Chapter 7 summarizes the main findings and contributions of my thesis. I also discuss future possible directions on filter bubbles in the news recommender systems.

CHAPTER 2

BACKGROUND AND RELATED WORK

2.1 Recommender systems

Recommender systems' main premise is that the users have many choices (movies to watch, items to consider to buy, news to read, *etc.*), oftentimes too many to choose from, and the system is designed to find what the user might be interested in and filter out the rest. The recommender algorithm aims to learn the users' preferences and show them the most relevant items. The applications of recommender system vary widely, such as movie recommendations [25, 26], music recommendations [27], article recommendations [28], friend recommendations [29], etc.

Content-based recommender system [30] is one of the most commonly customized approaches to generate recommendations. Specifically, via machine learning classifiers, a content-based recommender tries to learn the user's essential content features which can be used to identify the items with maximal probability to be liked by the user. The "content" features can be a bag of words or embedding representations of the description of the item, a tabular representation of the features describing the item, and a combination of the two. A personalized classifier commonly uses the features to learn the habits and preferences of each individual user.

Another prevalent personalization algorithm is the collaborative filtering method. The main idea of collaborative filtering is to identify similar interests or similar users and use the similarities to rank the recommended items. There are multiple ways to achieve collaborative filtering, such as item-item, user-user, or matrix factorization [31, 32, 33]. Both item-item and user-user methods typically calculate the ratings by averaging the ratings from similar users and items. On the other hand, matrix

factorization decomposes the user-item matrix into the matrix product of two low-dimensional matrices. The gradient descent methods are generally used to optimize the matrix. Unlike content-based algorithms, collaborative filtering usually depends on the completion of neighboring users' history, and also faces cold-start problem for a new user or a new item that does not enough historic records.

One inevitable problem with recommender systems is that they often introduce various biases into the recommendation process. One major bias type is the popularity bias in ranking-based recommenders [34, 35, 36]. Such bias ranks popular items higher than less popular items. The prior distribution of the items causes the popularity bias in most cases. Anchoring bias is another major issue that the user's preferences may be affected by the existing records that the system learned previously [37, 38]. The optimized algorithms impact users' experiences during the recommendations. As a result, these biases may cause filter bubbles (also called echo chamber) effects, which is discussed in the next section.

2.2 Filter bubbles

Two primary, intersecting factors – technological and psychological – contribute to the formation of filter bubbles. The technological component refers to filtering algorithms that are designed to increase user engagement by presenting users with content that they are more likely to click on [39, 40, 41]; the psychological component refers to the tendency for users to seek out or be more accepting of information that is consistent with their preexisting attitudes and beliefs [42, 43, 44, 45, 46]. For our purposes, news source and content are central to both of these factors [47].

Much attention has been given to filter bubbles in the context of social media. For instance, research on filter bubbles has shown that, with regard to Twitter, segregation is neither uniform across ideological orientations nor across the range of topics

available for consumption [48]. On Facebook, Bakshy *et al.* [49] examined 10 million users to quantify individual exposure to diversified news, finding that liberals are less likely to encounter ideologically cross-cutting news content than conservatives, a finding consistent with parallel research of Twitter [50]. Yet, online and offline political engagement can increase with exposure to this cross-cutting news, particularly when it originates from individuals not necessarily in one's own filter bubble, i.e. individuals with whom one has weak connections [51]. Beyond news articles themselves, and highlighting the role of influential elites in filter bubble formation [52], comments about content on Facebook and YouTube can also be predictors of echo-chamber formation [53, 54, 55].

Beyond social media-based experiments, and given that, in the U.S., nearly one-fifth of Democrats and Republicans obtain news in a filter bubble-like dynamic [56], efforts have been made to simulate recommender systems to more closely observe filter bubble dynamics. These simulations are able to control select parameters, altering specific characteristics of the online environment. Epstein *et al.* [57], for example, evaluated "Search Engine Manipulation Effects" and confirmed that ranking bias shifts the behavior of the voting population, thus increasing the vote share for targeted candidates. This finding has since been confirmed via experiments using representative samples of the American public [58]. Elsewhere, Geschke *et al.* [59] constructed an agent-based model to test the emergence of the filter bubble effect, while Chaney *et al.* [60] and Jiang *et al.* [61] built a simulation environment defining and measuring the filter bubble effect across a variety of recommender algorithms.

Ultimately, filter bubbles have significant and often confounding effects with regard to how people perceive consensus and mobilize around partisan and policy issues [62, 19, 20, 21, 22, 23]. Without some form of intervention, there are significant implications for how one is able to properly receive and process information, accurate

or otherwise. Information distortions may not consistently have lasting effects [63], but filter bubbles can affect voters' election-related decisions nonetheless [20].

A number of strategies that aim to alleviate filter bubbles are proposed. Masrour *et al.* [64] study filter bubbles created by network link prediction algorithms and propose a framework that utilizes adversarial learning to create more heterogeneous links in the network. Bhargava *et al.* [65] propose providing transparency and content control mechanisms to the users to combat filter bubbles on social media. On the news consumption domain, "bias alerts" sent to users can be considered partially effective in mitigating the voting-related implications described above [57]. Providing accuracy reminders before news is consumed may minimize the likelihood that people will trust and share potentially inaccurate information [66, 67]. Yet, one's understanding of what is truly inaccurate is confounded by news source. Specifically, Dias *et al.* [68] find that source identification by users may help identify implausible news content from trusted new sources while simultaneously making it more difficult to identify plausible news content from untrusted news sources. This only reinforces the need to use bias alerts and accuracy reminders before news is consumed and perhaps periodically afterwards, too.

Previous studies show that individuals, especially those with solid attitudes, prefer to receive content and information that is consistent with their stances [69, 70]. Thus online media and networks would design recommender systems that expose people to political information consistent with their preexisting views [24]. However, sometimes individuals would make their own choices on the contents which may play a more vital role than the algorithm itself [49]. For example, some research shows that in Twitter, users from the liberal side are more likely than users from the conservative side to participate in cross-ideological dissemination of political information [48]. On the contrary, some other work shows that conservatives prefer to follow me-

dia and political accounts classified as left-leaning than the reverse [71]. Therefore, recommender systems may or may not create "echo chambers" on people's political views, and their actual effects may be caused by many factors.

2.3 Automated frame analysis

Political articles usually express the political opinions of the author explicitly. Most articles influence the readers for particular political purposes by phrasing the sentences from certain angles. Such expressions of events and facts are usually called media framing. However, the media frame would limit the points and may mislead the readers. Some previous research also shows that news diversity is crucial and important for readers to receive the facts.

Boydstun *et al.* [72] propose policy frames code-book that provides issue-general and issue-specific approaches to fifteen different frames. The other researchers in this domain extensively study these defined frames. A corpus is extracted and stemmed for each frame by different annotation stages [73]. Supervised deep learning models achieve promising accuracy to classify the frames at sentence level in news articles and tweets [74, 75]. Frame detection task is found to be time resilient through classification tasks [76].

Many researchers also focus on the framing and narratives on specific topics to understand policy influences [77]. For example, immigration is typically a divisive topic that is discussed extensively by both liberal and conservative sides. The study of framing on immigration shows that equivalency frames around immigrants themselves have little impact on perceptions [78]. Lawlor *et al.* [79] finds that framing is disproportionately negative when talking about refugees rather than immigrants. Other critical policy issues, such as guns, trade, defense, elections, environments, education, and many others, are well studied in framing analysis to conclude that

frames around policies do have significant impacts [80, 81, 82, 83, 84, 85].

2.4 User studies for recommender systems

One of the main challenges of evaluating recommender systems is to design an appropriate user study to investigate the specific hypotheses. Pu *et al.* [86] proposes an evaluation framework that consists of two components 1) decision accuracy-effort framework [87]; 2) the user-trust model. The first is to measure objective goals, such as accuracy, interaction effort, and time; the second is subjective measures, such as trust and satisfaction that are typically measured through post-questionnaires. Knijnenburg *et al.* [88] proposes a pragmatic procedure to support the user-centric evaluation of recommender systems that assign participants to conditions, log interaction behaviors, measure the subjective experience, and analyze the collected data. We adopt similar ideas in our user study, where we measure both quantitative metrics, such as filter bubbles, and ask subjective questions, such as awareness of filter bubbles, as post questionnaires.

Several papers have conducted user studies on recommender systems in various domains. In the work of [89], Garcin *et al.* presents the comparison of offline and online accuracy evaluations, and concludes that click-through rate is not always a good indicator to assess the performance of a the recommendation system. Gaspar *et al.* [90] studies the attention bias (e.g., position bias or visual bias) in recommended movie lists. Ghori *et al.* [91] demonstrates that users appear to possess a cognitive model of recommender systems. More recently, researchers [92] propose and utilize the power of natural language processing techniques to design a recommender system that results in more transparency, engagement, and trustworthiness for the users.

2.5 Summary

In this thesis, I create a large news collection dataset (over 900K news) to

conduct our studies. Unlike much of the previous work that simply define media sources as liberal or conservative, I adopt the definition from *www.allsides.com* website to get a more nuanced political spectrum of the sources, ranging from extreme liberal to extreme conservative. Similarly, unlike much of previous work where users are simply treated as liberals or conservatives, I build on the Pew Study to build 11 political typologies (including solid liberals and core conservatives who have uniformly liberal/conservative views on issues, but also groups that have diverse views, such as "devout and diverse"). I also study the users' behaviors not just overall, but per topic, on 14 topics, that we have identified and used to annotate our dataset.

Having identified the need to account for all the factors I discussed, I investigate precisely how machine learning algorithms create and exacerbate filter bubbles for individuals with varying political views and on various issues. I apply the idea of multi-dimension of political ideology in our simulation study (Chapter 4), as well as in the user study (Chapter 6).

CHAPTER 3

DATA COLLECTION AND ANNOTATION

For our study, we require a large set of news articles annotated by both political stance and topic. In this chapter, I summarize our data collection and annotation process. Our overall approach is to use the news source as a proxy for political stance, and to use text classifiers to assign topic(s) to each article.

3.1 News article collection

To collect a range of political news articles, we first identified 41 featured news sources from *www.allsides.com*, which annotates each source with a *political stance* in $\{-2, -1, 0, +1, +2\}$, ranging from very liberal (-2) to very conservative (+2). The ratings are based in part on user surveys of the perceived slant of the news source.

To collect articles, we next query the Twitter API with the URL of each source to identify tweets that contain links to news articles. We then crawl each URL and collect the title, source, and content of each article. We processed these queries continuously from September, 2019 to August, 2020, resulting in over 900K articles. The resulting data is summarized in Table 3.1. Popular sources from each stance include DailyBeast (-2): 17K articles, New York Times (-1): 47K, Forbes (0): 74K, Fox News ($+1$): 36K, and Brietbart ($+2$): 28K articles. Each article is annotated with the partisan score of its source.

While this process gives us a broad range of articles from across the political spectrum, it is of course not without some sampling bias. E.g., articles shared on Twitter differ from a uniform random sample of all articles from all news sources. However, given that our focus is on articles likely to be read and shared by users, this

Table 3.1. Statistics of the collected news articles.

stance	interpretation	# sources	# articles	% articles
-2	extreme left	10	93,700	10.1
-1	moderate left	11	282,432	30.3
0	neutral	8	286,639	30.8
+1	moderate right	4	93,279	10.0
+2	extreme right	8	175,998	18.9

sampling methodology seems appropriate for our purposes. Another limitation is that not all articles from certain source in fact have the same leaning with its source. For example, a news source may republish or reprint some articles from different sources or include commentary articles that differ with its general stance. While this may introduce some label noise at the article level [93], we expect this to have limited impact in aggregate.

3.2 Topic classification

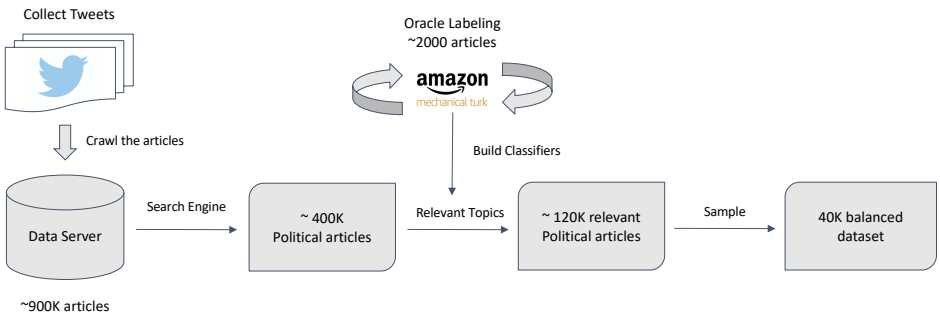

Figure 3.1. The pipeline to collect and label the data.

From the 900K articles we collected, our next goal is to build a classifier to annotate each article with the topics it discusses. To do so, we trained a two-stage classifier: one to determine if the article is relevant to U.S. politics, and a second one to assign one or more topics to the article.

To collect training data, my colleagues and I first independently annotated a sample of documents with political relevance and topics. Through several discussions and iterative refinement, we arrived at the following list of 14 topics: *abortion, environment, guns, health care, immigration, LGBTQIA, taxes, technology, trade, Trump impeachment, US military, welfare, US 2020 election*, and *racism*.

To increase the training sample, we next sampled additional documents to be annotated using Amazon Mechanical Turk. Using our annotations as a guide, we identified 12 high-quality AMT annotators, and had them annotate 3,250 total documents, of which 2,086 were annotated as politically relevant. The label distribution of this annotated dataset can be seen in Table 3.2.

From these labeled data, we next trained a binary classifier to determine if the article is relevant to U.S. politics or not. For this we used a standard logistic regression model using tf-idf features. The performance of this classifier is denoted in Table 3.3.

For topic classification, as it is a multi-label classification task, we trained 14 independent binary classifiers (one per topic). As the label distributions is highly imbalanced, we used SMOTE (Synthetic Minority Oversampling Technique) [94] to over-sample the positive class. Each of these topic classifiers uses logistic regression and tf-idf based features. These classifiers were separately optimized using a 5-fold cross validation loop with grid-search using the F1-score as the optimization metric. Table 3.4 shows the final cross-validation results for each topic. While F1 is generally

high, we note that the classifier has smaller F1 score for the technology and welfare topics. For technology, this is likely do to ambiguity of whether an article is related to U.S. politics – e.g., an article about Facebook's earnings is not relevant, but one that discusses new regulations is. For welfare, this topic is much broader than the rest, covering everything from cash assistance programs to homelessness issues. More training data would likely help here.

3.3 Article sampling

With the two classifiers described above, we then annotated all collected articles with political relevance and topic. Table 3.5 shows the predicted topic distribution of those articles determined to be relevant and to have at least one topic assigned. To ensure that the final sample has a uniform distribution of political stance, we randomly sample 8K articles from each stance, resulting in the final topic distribution in the final two columns in the table. (Note that many articles have more than one topic assigned.) Given the high fraction of articles about the 2020 election and Trump's impeachment, we additionally down-sampled these topics to ensure a broader diversity of articles.

Table 3.2. The label distributions of the training data for topic classification.

Topic	**Negative Labels**	**Positive Labels**
LGBTQIA	1,972	114
abortion	1,909	177
environment	1,963	123
guns	2,014	72
health care	1,947	139
immigration	1,978	108
racism	1,986	100
taxes	1,963	123
technology	2,032	54
trade	2,006	80
trump impeachment	1,803	283
us 2020 election	1,725	361
us military	2,001	85
welfare	2,002	84

Table 3.3. The performance of relevance classifier.

Accuracy	**F1**	**Recall**	**Precision**
0.7865	0.8307	0.7909	0.8773

Table 3.4. The F1 scores of the topic classifiers.

Topic	**F1**	**Topics**	**F1**
abortion	0.942	environment	0.898
guns	0.906	healthcare	0.785
immigration	0.853	LGBTQIA	0.894
racism	0.776	taxes	0.848
technology	0.538	trade	0.839
impeachment	0.888	US military	0.773
US election 2020	0.847	welfare	0.598

Table 3.5. Topic distribution for the simulations.

	before sampling		after sampling	
topics	# articles	% articles	# articles	% articles
abortion	3,421	1.7	1,382	2.6
environment	4,329	2.2	1,656	3.2
guns	4,647	2.4	1,787	3.4
healthcare	14,823	7.6	5,444	10.6
immigration	10,736	5.5	4,308	8.3
LGBTQIA	2,848	1.5	1,126	2.1
racism	10,051	5.1	4,069	7.9
taxes	8,187	4.2	3,055	5.9
technology	3,722	1.9	1,379	2.6
trade	6,739	3.4	2,323	4.5
impeachment	45,989	23.4	6,811	13.2
US military	17,205	8.8	9,409	18.3
US election 2020	57,996	29.6	6,501	12.6
welfare	5,413	2.7	2,054	4.0
# labels	196,106		51,304	
# articles	167,431		40,000	

CHAPTER 4

NEWS RECOMMENDER SYSTEM SIMULATIONS

In this chapter, I conduct extensive simulations on two recommender systems, nine political typologies, and fourteen political topics. The purposes of these simulations are two-fold: first, I would like the simulations to provide insights and guide us in designing our user studies. Second, I would like to run extensive experiments under a large set of conditions: different recommender systems (for e.g., introducing varying degree of randomness), different groups of users (for e.g., solid liberals, core conservatives), different topics (for e.g., abortion, welfare), and varying duration of recommendations (short-term effects and long-term effects). Running a cross-product of all these conditions through user studies is impracticable; hence, I first run extensive simulations to guide us our user studies, which I will discuss in Chapter 6. This chapter is based on material that was published in WWW'21 [95] available via DOI:10.1145/3442381.3450113.

Rather than treating users as simply "liberals" or "conservatives", I draw on recent Pew surveys of political typology [96], and simulate nine classes of users (e.g., solid liberals, disaffected Democrats, country first conservatives, etc.) with differing partisan preferences across 14 news topics. I conduct simulation studies to compare the articles recommended by *content* and *collaborative filtering* algorithms with those articles recommended by an "*oracle*" approach that observes the user's true preferences. This allows us to measure the change in diversity of recommendations introduced by the recommendation system versus what would be expected based solely on the user's true preferences. Specifically, I compare recommendation diversity and user utility measures to address the following research questions:

- **How do user preferences influence the diversity of recommendations?** We find that users with more extreme preferences are shown less diverse content but have higher click-through rates than users with less extreme preferences.

- **How do filter bubbles vary by the type of recommendation system?** We find that the filter bubbles created by content-based recommenders and collaborative filtering are markedly different. Content-based recommendations are susceptible to biases based on how distinctive the partisan language used on a topic is, leading to over-recommendation of the most linguistically polarized topics. Collaborative filtering recommenders, on the other hand, are susceptible to the majority opinion of users, leading to the most popular topics being recommended regardless of user preferences.

- **How does recommendation diversity vary for users with heterogeneous preferences?** We find that when users have divergent views on different topics, recommenders tend to have a homogenization effect. For example, if a user is conservative on most issues, but liberal on health care, they are shown more conservative articles on health care than desired. The reasons again differ based on the type of recommender: for content-based, lexical overlap between topics can mislead the recommender; whereas for collaborative filtering, a small group of users with heterogeneous preferences are "subsumed" by a majority group that has less diverse views.

4.1 Simulation framework

In order to study the relationship between user preferences and recommendation systems, we would ideally conduct large-scale user studies to observe real-world interactions. However, given the challenges of conducting such studies, we instead build on the growing line of research conducting simulation studies of recommendation

systems [97, 60, 98, 61].

To conduct such a simulation, we must make some assumptions about the interaction model. Our approach largely follows that of prior work [60, 98], though here we use real news articles annotated by stance and topic. We assume that each user has a predefined, fixed set of preferences over articles they would like to read. These preferences are parameterized by the topic and stance of the article; e.g., a user may prefer to read a liberal article about healthcare more than a conservative article about immigration. As we are interested in short-term effects of recommenders, for this study we assume that user preferences do not change over time, though this is of course an important consideration for future studies.

The simulation proceeds by first showing the user an article. We then simulate the user's response: either "like" or "dislike," sampled proportional to the user's preferences. With this feedback, the recommender updates its model to re-sort the remaining articles, then shows the next article to the user.

In the following sections, we describe this process in more detail, including the user profile model, a user-choice model, and specific recommendation engines we implement.

4.1.1 User utility model. We represent each user's preferences with a two-dimensional matrix of utility values $U = \{u_{ij}\}$, where $u_{ij} \in [0, 1]$ indicates the user's utility for reading an article on topic i with political stance j. (Thus, U is a 14×5 matrix.) Large values indicate greater utility and therefore a larger probability of clicking on an article with topic i and stance j.

We wish to investigate how recommender behavior varies with heterogeneous utility matrices. Rather than randomly generate these matrices, in order to make them more reflective of reality, we sampled them based on Pew surveys of U.S. po-

litical typologies [96]. This comprehensive survey attempts to identify more nuanced political ideologies than a simple left/right spectrum. The survey contains many questions relevant to our identified topics above. E.g., for abortion, there is a survey question asking whether abortion should be legal in all/most cases. For immigration, there is a question asking whether immigrants strengthen or weaken the country. Pew clustered the responses to identify nine political types: *solid liberals, opportunity Democrats, disaffected Democrats, bystanders, devout and diverse, new era enterprisers, market skeptic Republicans, country first conservatives*, and *core conservatives*. These types capture a number of common heterogeneous ideologies – for example, the devout and diverse type leans conservative on issues of abortion and LGBTQIA, but leans liberal on race and health care. Similarly, the market skeptic Republicans lean liberal on issues of trade and taxation.

For each political type, then, we have a list of survey responses indicating the fraction of respondents who agree with the statement (e.g., 92% of solid liberals think that abortion should be legal in all/most cases). In our simulations, to generate a new user, we first pick a political type, then sample a utility matrix based on these survey responses.

We convert these responses into a utility matrix as follows: for each survey question, we separate the responses into quantiles (0-20%, 21-40%, etc.), and assign the response to one of the five political stance categories $\{-2, 1, 0, +1, +2\}$. Thus, the fact that 92% of solid liberals think abortion should be legal means that their primary stance is -2 on abortion. To generate the utility value for each topic/stance pair, we first sample a utility value for the primary stance using a Beta distribution centered on their survey response (e.g., $Beta(.92, 1)$ for the running example). We then decay this value for the other stances for this topic as a function of standard deviation of responses on this topic (i.e., a measure of how divisive this topic is). We

then repeat this process for each topic. Table 4.1 shows an example utility matrix for the devout and diverse profile.

As with any simulation, one can question how reflective the simulated users are of the real world. The key aspect that these utilities do capture, however, is a broad spectrum of ideologies with which we can investigate variation in recommender behavior.

Table 4.1. An example of the utility matrix for a "devout and diverse" user.

topics	-2	-1	0	+1	+2
abortion	0.276	0.411	0.546	**0.682**	0.546
environment	0.298	0.505	**0.711**	0.505	0.298
guns	0.332	0.490	**0.648**	0.490	0.332
healthcare	0.515	**0.711**	0.515	0.319	0.122
immigration	0.045	0.285	0.525	**0.766**	0.525
LGBTQIA	0.250	0.423	0.596	**0.769**	0.596
racism	**0.815**	0.575	0.335	0.095	0.010
taxes	0.080	0.283	0.486	**0.689**	0.486
technology	0.228	0.397	0.567	**0.737**	0.567
trade	0.400	0.511	0.622	**0.733**	0.622
Trump impeachment	0.313	0.452	**0.592**	0.452	0.313
US military	0.171	0.362	0.553	**0.744**	0.553
US election 2020	0.180	0.395	**0.610**	0.395	0.180
welfare	**0.860**	0.582	0.304	0.025	0.010

4.1.2 User interaction model. Given a user's utility matrix, we next must simulate their behavior when presented with a recommended article. To do so, we follow the approach of prior work [60]. To represent each article, we create a binary matrix of the same shape as the user utility matrix, containing 1 in cell (i, j) if the article has been assigned topic i and stance j. (Recall that the topic is derived from the text classifier, and the stance from the news source.) To sample whether a user will "like" or "dislike" an article, we first flatten both the utility matrix and the item matrix into 1d arrays, then compute the dot product between them. We then sample a value from a *Beta* distribution centered on this dot product value. Finally, a random number is generated and compared to the sampled value to determine the action of the user. Algorithm 1 formalizes this process.

Algorithm 1 The pseudo-code of the user interaction model.

Input: u – the user vector; v – the item vector

Output: B – a Boolean variable to indicate whether the user is going to like this item or not

$v_{ui} = Beta^1(dot(\mathrm{u}, normalized(\mathrm{v}))$
$p_{ui} = v_{ui} \times Beta^1(0.98)$
if $Random < p_{ui}$ **then**
  return Like
else
  return Dislike
end if

In the algorithm, the function takes the user vector u and the item vector v. We calculate the dot product with u and normalized v to constrain the output as a probability from 0 to 1. Following previous work [60], we choose a modified $Beta^1$ distribution (for which the mean and standard deviation are given) to calculate the probability p_{ui} the user will click the given article. A random number is generated

and used to determine whether the user will click this article, given p_{ui}.

4.1.3 Recommender models. We implemented five recommender systems, including a random recommender (as a baseline), a content-based recommender, a collaborative filtering recommender, an oracle recommender, and a hybrid recommender.

Random recommender: A random news recommender randomly selects the articles from the pool without replacement.

Content-based recommender: A content-based recommender (CBR) is a user-personalized model that learns the user's preference, given the user's previous interactions. We treat this as a binary classification problem – given an article, will the user like or dislike it? As training data, we seed the model with 700 simulated examples per user, sampled uniformly for each topic. We train a standard logistic regression classifier separately for each user, using tf-idf word features from each article. During the simulation, the training data is updated after each user interaction, and the model is retrained. Note that the classifier does not observe the stance and topic assignments for each document – this simulates the situation where neither the structure nor values of the user's utility matrix are known to the recommender.

Collaborative Filtering recommender: A collaborative filtering recommender (CFR) uses the concept of similarities between users and items and recommend similar users the 'liked' items from each other's 'like' history. We use nonnegative matrix factorization [99] on the user-item matrix to construct the collaborative filtering recommender.

Oracle recommender: We also implement an oracle recommender, which observes the user's utility matrix and news' topic and stance matrix. This algorithm samples documents proportional to the user's probability of liking these documents. This base-

line enables us to observe what biases are introduced by the recommender algorithms versus those that are inherent in the user's pre-existing preferences.

Hybrid recommender: A simple way to try to reduce filter bubbles is to inject random recommendations into the user's article list. We are interested in how the systems behave as the amount of randomness is injected. How quickly does the diversity increase as we introduce randomness? To investigate this, we consider three settings for each recommender above: randomness as 0% (totally personalized), 50% (hybrid), and 100% (totally random).

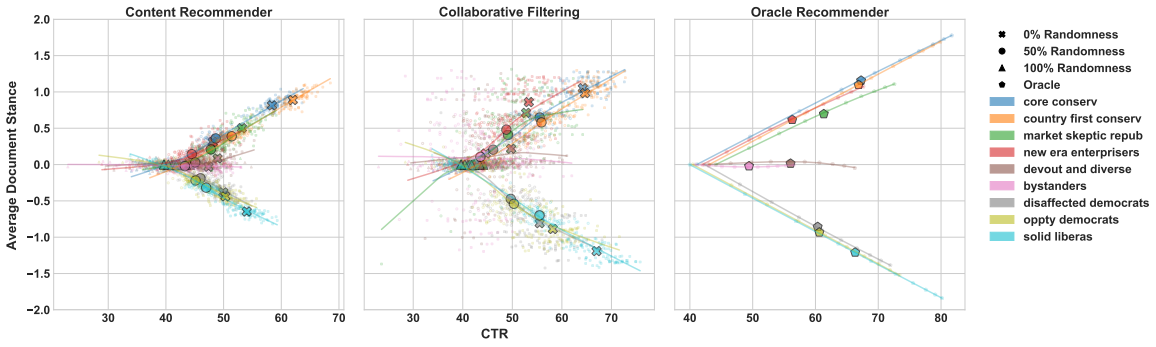

Figure 4.1. Simulation results by political typology, showing click-through rate vs average document stance for three levels of randomness.

4.2 Experimental metrics and results

In order to answer the three questions we proposed in the beginning of Chapter 4, we designed simulations to study recommender behavior for users of different political types. In this section, we formulate our filter bubble metrics and the details of experimental setup, then discuss the experimental results.

4.2.1 Problem formulation and metrics. Let V be a collection of news articles. Each article $v \in V$ is associated with one or more of 14 topics introduced in Section 3.2. Let U be a group of users. Each user $u \in U$ belongs to one of the nine political types introduced in Section 4.1.1. In each simulation run, every user u is recommended N articles, one at a time. For each recommended article i, we simulate a binary

random variable r_i, where $r_i = 1$ mean the user clicks on /likes the article and $r_i = 0$ means they do not.

We propose the following metrics to study the filter bubble effect of different algorithms on different political types.

Click-through rate: The click-through rate (CTR) is the fraction of recommended articles that the user clicks on. A high CTR indicates that the algorithm can deliver accurate recommendations to the users, and thus has high utility. The CTR is defined as follows.

$$CTR = \frac{\sum(r_i)}{N}, 1 \leq i \leq N \tag{4.1}$$

Average document stance: Average document stance is the average partisan score of the articles that are *shown* to the users. Letting $s(v_i) \in \{-2, -1, 0, 1, 2\}$ be the partisan score for article v_i, then the average document stance for a sequence of recommended articles is:

$$\overline{s} = \frac{\sum s(v_i)}{N}, 1 \leq i \leq N \tag{4.2}$$

Normalized stance entropy: Let p_i represent the fraction of articles that are shown to the users and that have stance i. Normalized stance entropy is the entropy of this distribution, normalized by $\log m$ so that its maximum is 1, where $m = 5$ in our case, representing the five stances:

$$\textbf{entropy} = \frac{-\sum_{i=1}^{m} p_i \log p_i}{\log m} \tag{4.3}$$

A high value of normalized stance entropy would indicate a smaller filter bubble effect since the stances of the shown articles are more diverse.

Normalized topic entropy: Similar to normalized stance entropy, we also measure the diversity of topics. This provides a measure of topical diversity, in addition to stance diversity above. The metric is the same as Equation 4.3, where p_i is instead the probability of articles having topic i in a sequence of recommendations, and $m = 14$ since there are 14 topics. A low value of normalized topic entropy indicates that the recommender is recommending documents in a small set of topics.

4.2.2 Experimental setup. We generate 100 synthetic users for each political type following the user utility model described in Section 4.1.1. To initialize the recommendation models, we initially bootstrap 50 articles per topic for each user, resulting in 700 articles in total. Then the recommender recommends 1,000 articles, one by one, in a sequence and updates the algorithm after each recommendation. The CBR and CFR have three different randomness settings as we mentioned in the previous section.

We simulate the oracle recommender explicitly as follows. For a given political type, for every article v, we calculate the probability p_v that the given political type would click that article if they are shown that article, based on their user profile. To study varying degrees of randomness in the oracle recommender, we compute a sampling weight for each article as $\exp(w \times p_v)$ where w is a hyper-parameter. We sample K articles from our dataset, using weighted sampling without replacement. We repeat this process M times. The probability q_v that the article will be shown by the oracle is the fraction of samples that contain v. When $w = 0$, each article has $\exp(0 \times p_v) = 1$ weight, resulting in uniform sampling, and hence results in the random algorithm. As $w > 0$, articles that have a higher chance of being clicked gets a higher weight.

Once we have the shown (q_v) and click (p_v) probabilities, we can calculate the expectations for the CTR and all other metrics for all the political types using the

whole dataset. We choose to use K as 1000, and M as 5000 in our case. For the hyper-parameter w, we vary the value from 0 (totally random) to 9 (optimal personalized solution). For comparing CBR and CFR to the oracle recommender, we use w that achieves a similar CTR for that prototype, and analyze where the CBR and CFR differ from the oracle. This analysis provides us if the recommenders carry any bias other than what is specified by the user preferences.

4.2.3 Experimental results.

How do user preferences influence the diversity of recommendations? We first investigate how the user's political type influences the diversity of the recommended documents. Because there is a strong relationship between diversity and utility (i.e., CTR), we are particularly interested in their trade-off. We consider content-based recommender, collaborative filtering recommender, and the oracle recommender. For each, we have varying levels of randomness through the hybrid recommendation approach. In this way, we can plot how the CTR varies with filter bubble measures such as average document stance, stance entropy, and topic entropy. We would like to determine how this trade-off varies by political type.

Figure 4.1 shows the main results of CTR versus average document stance. Each panel summarizes the results of multiple simulation runs. Each dot represents the result for one user. For content-based recommender and collaborative filter recommender, each political type has three settings, which are 0% randomness, 50% randomness (hybrid recommender), and 100% randomness (random recommender). The larger symbols (e.g., circle, triangle, and cross) represent the centroids of each setting. For the oracle recommender, the randomness is controlled by the w parameter, where w ranges from $w = 0$ (fully random) to $w = 9$ (user preferences are given high priority). We also fit a LOWESS curve for each political type to visualize the tradeoff between CTR and document stance.

The first observation is that more extreme political types have both higher CTR and higher magnitude document stances. E.g., when no randomness is used, country-first conservatives have over a 60% CTR, and an average partisan score of nearly 1.0 for both content-based and collaborative filtering recommendations. On the other hand, more moderate political types, such as bystanders and devout & diverse, do not attain such high CTRs. These results make clear the intuitive finding that the more extreme a user's preferences are, the more extreme their recommendations will be, and that it is easier to find articles that they are likely to click.

We can also see from the third panel that the oracle is able to achieve even higher CTRs, though to do so it must recommend even more extreme and homogeneous documents. Figure 4.2 shows a similar result instead using stance entropy as a measure of diversity. For more extreme users, stance entropy decreases more quickly as CTR increases.

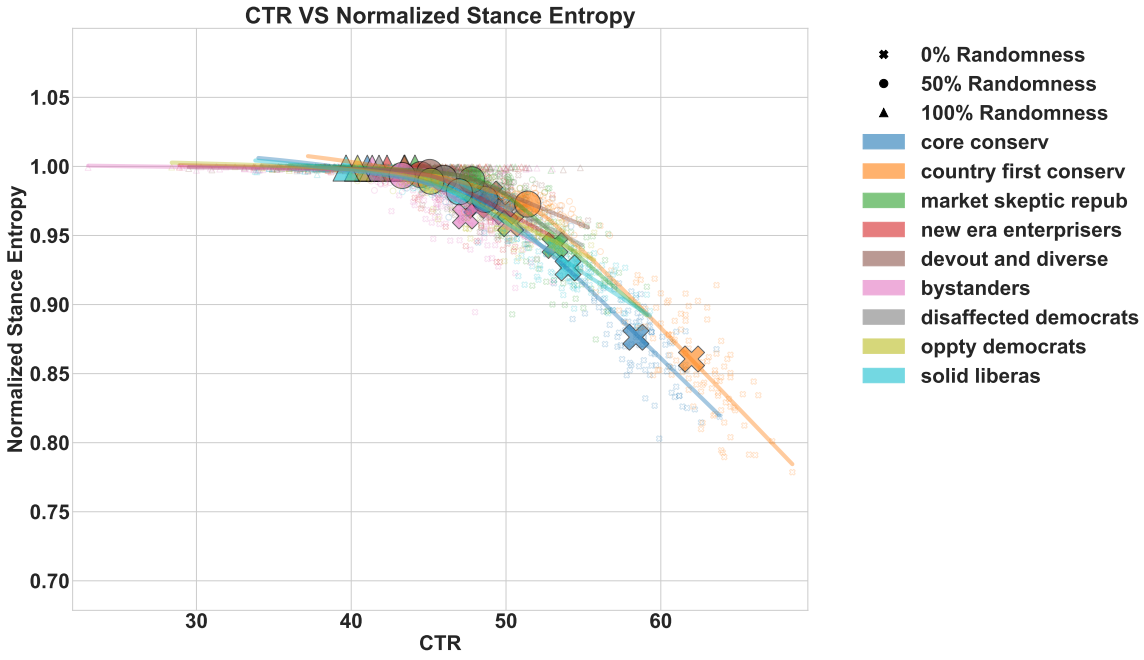

Figure 4.2. Click-through rate vs normalized stance entropy for the content-based recommender.

Examining these figures, there is a notable difference in the recommendation

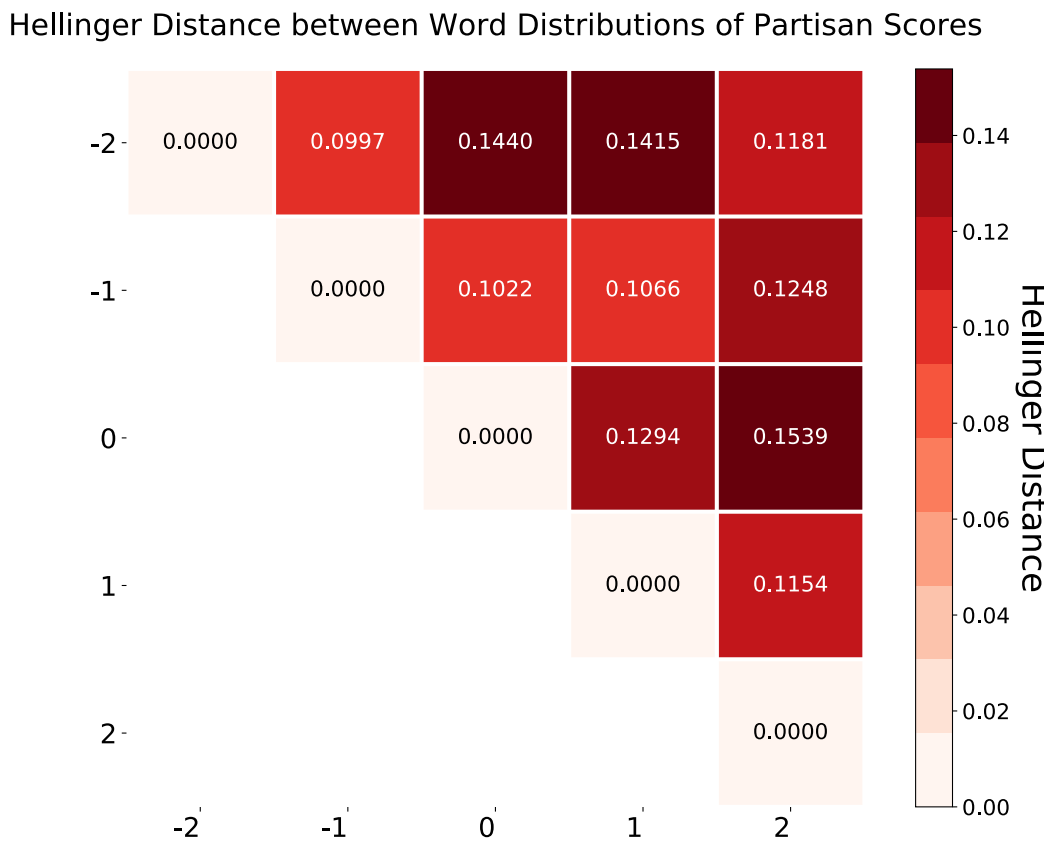

Figure 4.3. Hellinger Distance between different Partisan Scores

behavior for left-leaning versus right-leaning users. In the first panel of Figure 4.1, we see that right-leaning users ultimately exhibit higher CTRs, and more extreme partisan scores, than left-leaning users. Furthermore, we only see this difference in the content recommender, not for collaborative filtering or oracle recommenders. Upon further inspection, we conjecture that this is in part due to the asymmetry in the textual similarities between documents of different partisan scores. In particular, it appears that articles with score 0 are more similar to left-leaning articles (scores -2, -1) than they are to right-leaning articles (scores +1, +2). The result is that the content-based recommender has a more difficult time distinguishing between -2 and 0 articles than it does distinguishing between +2 and 0 articles. To further investigate this, we fit five different multinomial bag-of-words models, one per partisan score, by

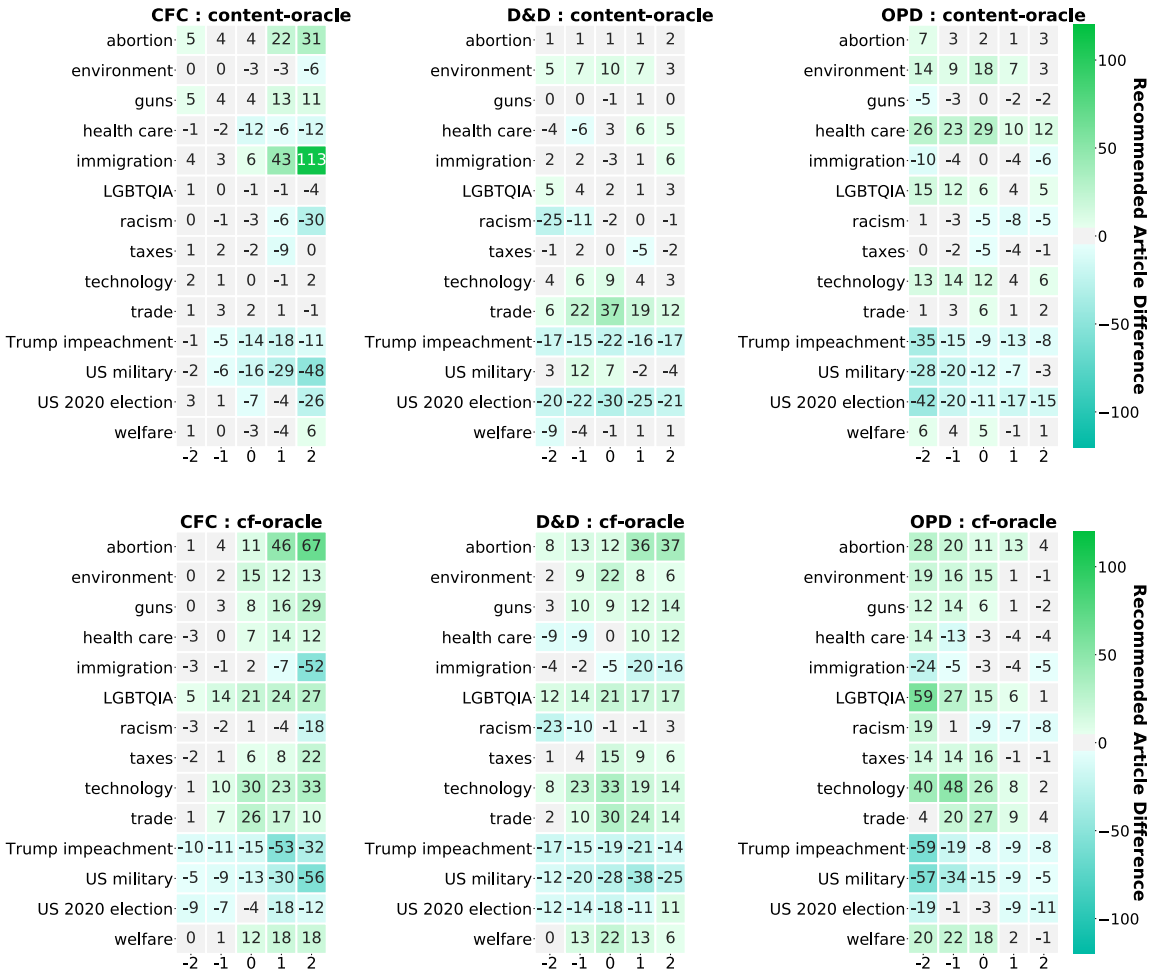

Figure 4.4. Difference in the number of articles recommended by the content-based and collaborative filtering recommenders as compared to the oracle recommender. Results are the average of 1,000 recommendations for 100 users from three user types: country first conservatives (CFC), devote and diverse (D&D), and opportunity Democrats (OPD).

grouping together all articles with the same partisan score. We then compute the Hellinger distance between each pair of multinomials to determine how similar the word distributions are. We find that the differences between -2 and 0 (.1415) and -1 and 0 (.1022) are substantially smaller than that between +2 and 0 (.1539) and +1 and 0 (.1294), further supporting this interpretation.

How do filter bubbles vary by type of recommendation system? As we have just seen, different recommendation systems can have different impact on filter bubble

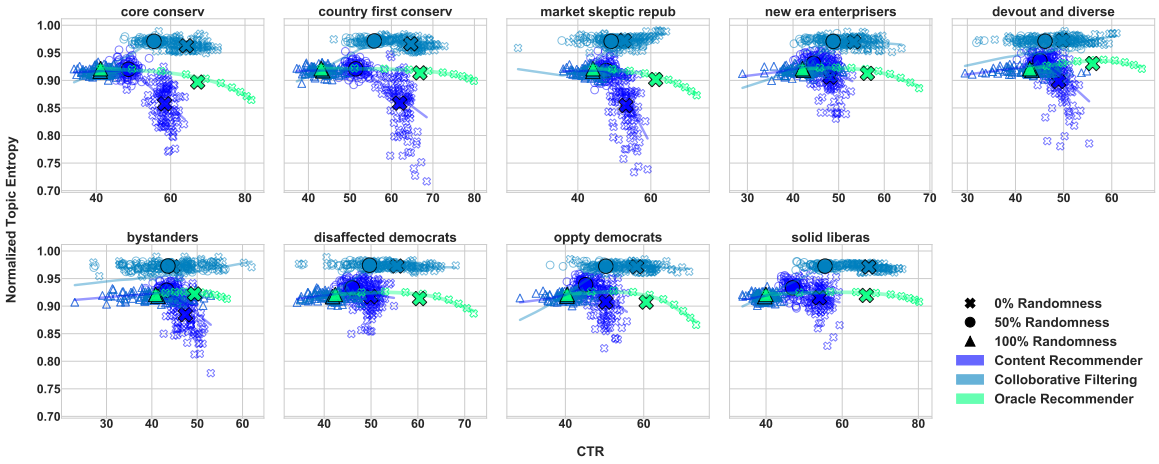

Figure 4.5. Click-through rate vs normalized topic entropy for all recommenders. The content-based recommender exhibits much lower topic diversity than others.

formation. In this section, we further compare CBR and CFR to their comparable oracle recommender counterpart to investigate possible biases introduced by CBR and CFR into the recommendation processes. To do so, we first compute the average number of articles recommended from each topic/partisan score pair for each political type, using the versions of CBR and CFR with the highest overall click-through rate. We then compare these values with the corresponding recommendations provided by the oracle recommender.[1]

Figure 4.4 shows the results for three political types: country-first conservatives (CFC), devout and diverse (D&D), and Opportunity Democrats (OPD). Each cell in the heat map displays the difference between the average number of articles recommended by either CBR/CFR and those recommended by the oracle. For example, in the top left panel, we see that the content-based recommender shows on average 113 more immigration/+2 documents than the oracle does to country-first conservatives. By examining these results, we can identify a few trends that characterize the different sorts of bias introduced by either content-based or collaborative

[1]We select the randomness hyper-parameter w to result in an oracle with the same click-through rates as the CBR or CFR method it is being compared with.

filtering recommenders.

For CBR, a key source of bias is **linguistic polarization**. For some topics, there is a clear distinction between the language used in right-leaning articles versus left-leaning articles. For example, in the immigration topic, terms like "illegal" and "alien" are much more likely to appear in right-leaning articles, while terms like "undocumented" are more common in left-leaning articles. In such cases, it will take few training examples for the recommender to develop an accurate model of user preferences, resulting in an over-recommendation of such topics. Furthermore, this can often result in a feedback loop, wherein immigration/+2 articles are recommended and clicked on, further reinforcing the over-recommendation of such articles.

This behavior is most noticeable in the immigration/+2 cell of the first panel of Figure 4.4. We can further see this behavior in Figure 4.5, which shows that content-based recommenders tend to have lower entropy over topics shown than the other two recommendation models for all of the political types at the extreme ends.

For collaborative filtering, we identify two sources of bias. The first is that the distribution of preferences across all users will influence the popularity of some topics over others. For example, across all political types, abortion and trade have high utilities, so they tend to be over-recommended across all user types. We also observe that minority groups tend to be 'subsumed' by larger groups. For example, the devout and diverse group appears to be grouped with more right-leaning groups and hence recommended more right articles across almost all topics, whereas the opportunity Democrats are grouped with left-leaning groups and hence are recommended more left articles across almost all topics, as the bottom row of Figure 4.4 shows.

A final source of bias that affects both recommendation systems is the overall makeup of the pool of articles to be recommended. As Table 3.5 indicates, topics

such as US military, US election, and impeachment are the most common. The initial bootstrap for CBR and CFR had equal articles from each topic (50 articles from each topic), hence these topics were underrepresented compared to their representation in the overall pool. Thus, articles from these topics tend to be under-recommended by CBR and CFR systems compared to the oracle recommender, which does not have a bootstrap and hence is unaffected by it.

How does recommendation diversity vary for users with heterogeneous preferences? The biases described above can also have effects on users with heterogeneous preferences. For example, Devout and Diverse users lean right on most issues, but lean left on issues of race, welfare, and health care. Both content-based and collaborative filtering systems under-recommend left leaning articles on these topics, but for different reasons. For collaborative filtering, the devout and diverse users are clustered together with other right-leaning users (e.g., core conservatives). Because those other users have right-leaning preferences for race and welfare, the devout and diverse users are recommended similar articles. Similarly, while the content-based recommender over predicts immigration/+2 for country-first conservatives, the collaborative filtering algorithm instead *under* predicts this category. The CFC type is most distinct because it is more conservative on immigration than "typical" right-leaning users, and so they are grouped together with these more typical users and shown less extreme views on immigration.

The explanation for the content-based recommender is more nuanced. A central issue is that there is keyword overlap across topics that can mislead the recommender. For example, the keyword "baby" correlates with right-leaning articles both for the abortion topic and the health care topic. Because D&D users lean right on abortion issues, after clicking on several right-leaning abortion articles, the recommender may also start to recommend right-leaning health care articles, contrary

to their preferences. Similar behavior occurs between the welfare and taxes topic, where the term "socialist" correlates with right-leaning articles for both topics. As D&D users lean right on taxes but left on welfare, left-leaning articles on welfare are under-recommended.

Together, these examples suggest that recommender systems can have a homogenization effect on such users, for example by pushing D&D users to more typical right-leaning articles, and by pushing opportunity democrats to more typical left-leaning articles, even though their true preferences are more mixed. Importantly, we do not see such behavior for the oracle recommender, but rather these are artifacts of the biases of recommendation systems that learn imperfect models of user preferences.

4.3 Summary

In this chapter, I discussed how we designed the simulation framework to understand the relationship between political typology and news recommendation algorithms. We found that a) users with more extreme preferences were shown less diverse content but had higher click-through rates than users with less extreme preferences; b) filter bubbles created by content-based recommenders and collaborative filtering were markedly different for different reasons; c) when users had divergent views on different topics, recommenders tended to have a homogenization effect.

The findings of this chapter guide the next two chapters. First, I analyze the news content and linguistic polarization in detail in the next chapter (Chapter 5). Then, I design and conduct user studies using a content-based recommender system in Chapter 6.

CHAPTER 5

CONTENT ANALYSIS OF THE NEWS ARTICLES

I conduct a content analysis of over 100,000 labeled political news articles in this chapter. I use supervised and unsupervised learning to examine how news articles from different political views are constructed using distinctive phrases in various topics and political stances. I discover and explore the common and distinct words and phrases used by liberal and conservative media. I will discuss the content analysis results of single topic analysis, cross-topic analysis, and finally a juxtaposition of words that are similar in context but used by opposing sides.

5.1 Single topic analysis

5.1.1 Data preparation. I use the same dataset that I introduced in Chapter 3. In that chapter, each news article was labeled by its source in five discrete scales (-2, -1, 0, $+1$, and $+2$) and multi-labeled with one or more of fourteen topics (e.g., abortion, race) using logistic regression classifiers trained on a subset of the data that was annotated by MTurkers. In this chapter, I first examine the articles from two different political stances: liberal versus conservative. Therefore, I omit the articles from the center, i.e., the sources with a partisan score of 0. I label the articles as "liberal" if their source partisan scores are -2 or -1 as "conservative" if their source partisan scores are $+1$ or $+2$.

For the unsupervised learning approach, I use all the labeled articles rather than the sampling methods used in Chapter 4. The statistics of the dataset I used are shown in Table 5.1. Each row shows the number of articles in either liberal or conservative categories. The major topics include U.S. election 2020, Trump impeachment, U.S. military, etc. There are 74,615 articles on the liberal side and 55,022 articles on

Table 5.1. The statistics of the liberal and conservative articles.

Topic	**# Left articles**	**# Conservative articles**
LGBTQIA	1,611	749
abortion	1,277	1,637
environment	1,756	1,186
guns	1,897	1,891
health care	7,229	3,203
immigration	4,677	3,981
racism	5,268	3,137
taxes	3,201	2,144
technology	1,770	852
trade	2,455	1,668
trump impeachment	20,961	15,980
US election 2020	25,085	21,632
US military	7,579	5,156
welfare	2,372	1,230
# labels	87,138	64,446
# articles	74,615	55,022

the conservative side.

In the single and cross-topic analyses, I focus on two topics: "abortion" and "guns." I study how the liberal media and the conservative media frame the discussion on these topics. I use machine learning techniques, such as classification and clustering, to distinguish the words that they use on a single topic (e.g., abortion), and the common words that they use across different topics (e.g., guns versus abortion).

5.1.2 Methodology and results.

To make the results intepretable and simple, I use a bag-of-words representation, in combination with a linear classifier, logistic regression. I first vectorize the whole dataset into a bag-of-words representation using CountVectorizer implemented in scikit-learn [100]. For fine-tuning the parameters of the vectorizer (e.g., binary indicator, minimum document frequency, etc.) and to filter out the most stop words, I use WordNet to maximize the percentage of the verbs, nouns, adjectives, and adverbs in the representation. I sample 1,200 articles on abortion for both liberal and conservative classes to ensure balanced distribution. The same process is done with the topic of guns as well. I fit one logistic regression model to the articles on abortion, predicting whether they appear in a liberal news outlet or a conservative news outlet. I fit another logistic regression for the guns topic.

Table 5.2 presents the top 20 words for the abortion topic and Table 5.3 presents the top 20 words for the guns topic, ranked by the magnitude of the weights. The column 'DF' is the document frequency of the term. The weights of features are also provided.

Some interesting observations can be found in the abortion topic. Word 'anti' is the most informative feature in the liberal because liberal-leaning media would like to report anti-abortion movements. Also, in the articles, liberal-leaning articles would

Table 5.2. The top features for the left and for the right on the topic of abortion. L-words are the words assigned to the Left (liberals) by the classifier and R-words are the words that are assigned to the Right (conservatives). DF is the document frequency and weight is the coefficien of the classifier.

L-words	**DF**	**weight**	**R-words**	**DF**	**weight**
anti	804	-0.68	pro	1161	0.94
reproductive	987	-0.42	baby	543	0.53
access	1054	-0.41	healthcare	226	0.51
donald	426	-0.39	foundation	209	0.42
colleague	204	-0.35	taxpayer	335	0.42
legislator	224	-0.35	industry	191	0.42
potential	193	-0.34	abortion	2384	0.4
position	483	-0.33	left	501	0.39
university	314	-0.32	born	335	0.38
cut	166	-0.32	fact	577	0.38
identical	111	-0.32	unborn	475	0.38
ban	797	-0.31	life	1608	0.37
email	159	-0.29	reported	429	0.37
evangelical	66	-0.28	democrat	837	0.34
mike	219	-0.28	aborted	165	0.33
helped	131	-0.28	report	462	0.33
personal	317	-0.28	circuit	320	0.32
passing	168	-0.28	history	383	0.31
appointment	186	-0.28	spending	107	0.31
option	220	-0.28	june	426	0.3

Table 5.3. The top features for the left (liberals) and for the right (conservatives) on the topic of guns.

L-words	**DF**	**weight**	**R-words**	**DF**	**weight**
mass	1088	-0.19	amendment	1021	0.28
donald	499	-0.19	gun	2371	0.26
school	664	-0.18	control	1500	0.24
victim	359	-0.18	democrat	1020	0.23
measure	706	-0.17	confiscation	241	0.21
congress	659	-0.17	reported	456	0.2
body	131	-0.14	cnn	242	0.19
dead	287	-0.13	citizen	593	0.16
brady	131	-0.13	claim	318	0.16
federal	943	-0.13	firearm	1489	0.15
including	906	-0.13	abiding	419	0.15
killed	707	-0.13	claimed	199	0.15
association	707	-0.11	anti	384	0.15
statement	467	-0.11	left	532	0.14
news	648	-0.11	push	321	0.14
lobby	228	-0.11	criminal	631	0.14
elementary	151	-0.11	ban	906	0.13
massacre	308	-0.11	gov	326	0.13
handgun	518	-0.11	continued	273	0.13
passed	599	-0.1	bloomberg	167	0.12

call Donald Trump rather than President Trump. The liberal media also talks about 'reproductive' rights and 'access' to abortion. In the conservative class, 'baby' and 'unborn,' and 'aborted' are some of the top words. The words 'left' or 'leftist' are also among the top words in the conservative articles, talking about the liberal side.

In Table 5.3, similar patterns can be observed as well. In the liberal-leaning articles, there are several words related to violence, such as 'body,' 'dead,' 'killed,' and 'massacre.' On the conservative side, articles talk about the legal side of guns, such as 'amendment,' 'control,' 'confiscation,' etc.

These two examples clearly show that the liberal media and conservative media use markedly different language when they talk about the same topic. Content-based recommenders are expected to pick up on these keywords depending on the user's perspective on these topics. This can be quite problematic, especially for users with diverse views, especially if one content-based recommender is used across all topics, because the same words, regardless of which topics they appear in, would contribute to an article being ranked high (or low), leading to the homogenization of recommendations across topics.

5.2 Cross topic analysis

In this section, I discuss the results of the cross-topic analysis. More specifically, I am interested in the same words used on different topics and stances. I again present the results on two topics: "abortion" and "guns."

5.2.1 Cross-section of binary classifiers. I used the same processing that I introduced in the previous section: train binary classifiers for topics of abortion and guns separately. The classifiers are trained to distinguish the liberal articles and the conservative ones.

I first categorize the features into liberal or conservative for each topic, using

the weights of the respective classifier. Since we have four sets of weights from two classifiers, we take the intersection of these four sets (we did not do that for the liberal and conservative in the same topic). Each feature would have two weights from two classifiers. We then rank the features based on the harmonic mean of the two weights (see Equation 5.1).

$$H_i = \frac{2w_1^i w_2^i}{w_1^i + w_2^i} \tag{5.1}$$

I next summarize the findings.

- **Liberal abortion ∩ Liberal guns**: donald, including, federal, university, republican, access, group, measure, day, passed, researcher, issue, week, analysis, expert, series, comment, view, message, building, version, district, argued, argue, car, private, office, washington, country, interview, service, reform, spent, association, case, statement, conservative, july, prompted, representative

- **Conservative abortion ∩ Conservative guns**: democrat, reported, left, cnn, foundation, report, presidential, host, joe, continued, demand, pro, leftist, gun, outlet, claimed, medium, released, coronavirus, abiding, liberal, mayor, stated, protect, hopeful, beto, banning, fired, liberty, real, democratic, march, illegal, lawful, population, fact, rourke, wrong, cover, bloomberg

- **Liberal abortion ∩ Conservative guns**: anti, control, ban, favor, criminal, work, false, politician, institute, city, personal, legally, amendment, tuesday, class, comply, department, early, goal, gov, concluded, responsible, belief, innocent, option, carry, order, operation, fraction, event, man, began, supporter, entire, defund, political, material, denying, purchase, push

- **Conservative abortion ∩ Liberal guns**: body, lobby, mass, regulation,

killed, legislation, history, school, death, news, secretary, urban, received, child, obama, prevent, kill, criticism, doe, thought, member, health, support, wayne, room, restriction, represent, wanted, lobbying, sign, brown, embrace, policy, bringing, development, special, 000, matter, receive, violate

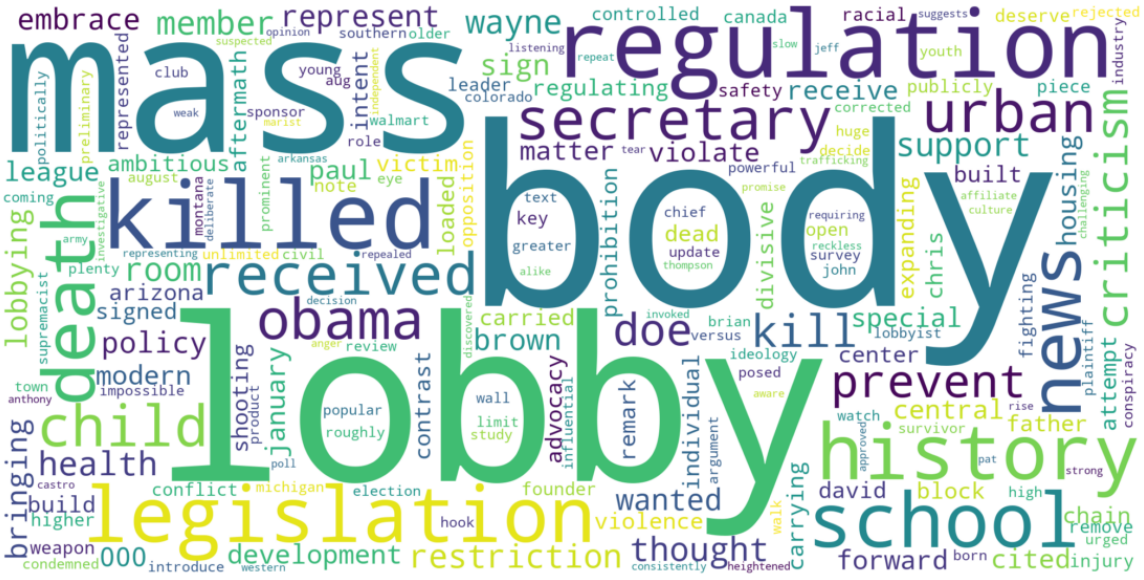

Figure 5.1. The word cloud of the top features for "conservative articles on abortion" and "liberal articles on guns."

The intersections of the liberal words on both topics and the intersection conservative words on both topics are consistent with the single topic analyses presented earlier. The intersections of different ideologies on cross-topics are quite interesting. For "left abortion + right guns" for example, both sides talk about legal aspects and access, such as *control, ban, legally, comply.* For "left guns + right abortion" however, the language is more inflammatory as both sides criticize the other using violence-related words, such as *killed, death, body, kill, violate.* I plot the word clouds in these two categories in Figures 5.1 and 5.2.

5.2.2 Clustering approach. Another way to find the language usage in different topics from different stances is to use unsupervised clustering approaches. I next

Figure 5.2. The word cloud of the top features for "liberal articles on abortion" and "conservative articles on guns."

describe this approach.

5.2.2.1 Problem formulation. Let D be a collection of news articles. Each document d has two attributes: a categorical topic label $l_{[d]}$ and a binary stance label $s_{[d]}$. A document consists a sequence of tokens $< t_1, t_2, ...t_i, t_{i+1}, ...t_n >$. Each token t has a unique associated embedding vector $\vec{t}$.

Let set U^i be a collection of articles with topic l_i. U^i is divided into two subsets, U^{i+} and U^{i-}, based on their stance labels. Each token t that appears in U^i has a stance score in topic l_i based on the definition of log-odds:

$$m_t^i = \log \frac{\text{percent}(t \in U^{i+})}{\text{percent}(t \in U^{i-})} \tag{5.2}$$

In two topics l_i and l_j, one token t can appear in both U^i and U^j with two scores m_t^i and m_t^j. We are interested in the tokens that have different signs m_t^i and m_t^j. The token set T^{ij} is a set of tokens that $m_t^i > 0$ and $m_t^j < 0$. Our objective

is to cluster the tokens in both T^{ij} and T^{ji} separately. It should be noticed that $T^{ij} \cap T^{ji} = \emptyset$. If we denote the token in the T^{ij} as t^*, and T^{ji} as t^{**}

$$L = \sum_{k=1}^{n} w(t_k^*) \sum_{i=1}^{c} u_{ik} ||\vec{t_k^*} - \vec{v_i}|| + \sum_{k=1}^{n} w(t_k^{**}) \sum_{i=1}^{c} u_{ik} ||\vec{t_k^{**}} - \vec{v_i}|| \tag{5.3}$$

where $w(t_k) = |m_t^i * m_t^j|$ and u_{ik} is a binary indicator that indicates which cluster token t belongs to. $\vec{v}$ is the vector of the centroid i. The Equation 5.3 defines the joint optimization function on two sets. The individual loss functions are similar to the loss function defined by the K-means algorithm.

5.2.2.2 Experimental results. We cluster the tokens in T^{ij} and T^{ji} separately since they are mutually exclusive. We used the K-means algorithm to cluster the words. For each word, we use a part-of-speech tagger to label each word with its part of speech role. Then, we label each cluster with a part of speech label using the majority of the labels of its tokens. I present two informative "adjective" clusters in Table 5.4.

I chose to present "adjective" clusters since I found that adjective words are the most informative ones to use the articles' sentiment. As discussed before, the left and right abortion use violent words to describe how they oppose abortion or guns. The center of cluster 'horrific' in this case clearly shows the language they used. On the other side, the opposed set used words (e.g., downright, dishonest, perilous, etc.) to show their political leaning, again to the other side.

$$H_i = \frac{2w_1^i w_2^i}{w_1^i + w_2^i} \tag{5.4}$$

5.3 Juxtaposition words analysis

In the previous sections, I analyzed the usage of the same words in different

Table 5.4. Two example clusters on the topics of abortion and guns.

Cluster index	**Examples**	
Liberal guns & Conservative abortion	Cluster Center	horrific
	Cluster Terms	abhorrent, unspecified, vile, needless, verbal, deadliest, deadly, ...
	Example Sentence	"This is horrific," Republican National Committee chairwoman Ronna McDaniel wrote on Twitter, adding that Northam "is defending born-alive abortions."
Liberal guns & Conservative abortion	Cluster Center	downright
	Cluster Terms	dishonest, perilous, stupid, unsettling, foolish, worse, unethical, bizarre, ...
	Example Sentence	Their theory of gun rights is downright radical—and would have shocked the framers of the Second Amendment.

contexts, different topics, and different ideologies. Here, I focus on words that are similar in context but used by different ideologies on the same topic. I examine the juxtaposition of similar words that are used in a given single topic. I propose and define the concept of juxtaposition words as the two words used in the same context but used by different partisan leaning sources. In this section, I analyze and present results on the immigration topic and on the topic of guns.

5.3.1 Data preparation. Different from the data processing in the single topic analysis, the data in this section are drawn from only extreme views (e.g., articles with stance (+2) and (−2)). Moreover, instead of articles that are labeled with a

topic, I access all articles that contain one or more words from a given list on a topic.

5.3.2 Problem formulation. I formulate the finding of juxtaposition words as follows. Let D be a collection of news articles in topic t. Each document d consists of a sequence of tokens $< t_1, t_2, ...t_i, t_{i+1}, ...t_n >$ in vocabulary V. Each token would be classified as either 'Liberal' or 'Conservative' by fitting a simple binary classifier based to the extreme liberal and extreme conservative documents.

5.3.3 Methods. I separate the tokens into liberal token set V_l and conservative token set V_c based on their classified labels. For each t_l in V_l, and each t_c in V_c, I would like to find such juxtaposition pairs as:

$$\text{Juxta}(t_l, t_c) = Sim(\text{Embed}(t_l), \text{Embed}(t_c)) \tag{5.5}$$

where Embed is a word embedding of a token, and Sim is a similarity function between two embeddings. I trained a word2vec model [101] to represent the word embeddings. The Sim could be any similarity function that calculates the inverse distance functions given two vectors. I used cosine similarity. I experiment with two topics: immigration and guns. I train a binary logistic regression classier M for each topic to classify whether the article is $(+2)$ or (-2). Each token t in the V would be assigned a weight from the coefficients of model M. I determine that the token t is liberal if its weight is negative; otherwise, it would be assigned to conservative.

I first sort the tokens based on their weights, and I take the top 300 tokens and bottom 300 tokens, which are the most conservative and liberal words in the specific topic. For all possible 300×300 pairs, I calculate the cosine similarity for each pair, then pick the top 25 pairs based on the similarity and present these pairs.

5.3.4 Experimental results.

Table 5.5. Top 25 juxtapositions on the topic of immigration.

Left-Term	**Right-Term**	**Similarity**
center	centre	0.78
children	babies	0.70
days	years	0.78
detained	arrested	0.77
detained	apprehended	0.77
detained	deported	0.70
employees	workers	0.71
immigrants	aliens	0.76
immigrants	migrants	0.75
people	americans	0.72
published	released	0.73
republican	democrat	0.74
republicans	democrats	0.91
says	said	0.82
says	added	0.78
seekers	migrants	0.70
spokesperson	spokesman	0.89
told.	said	0.74
told	noted	0.68
trying	attempting.	0.87
week	year	0.79
weeks	years	0.77
wrote	said	0.72
wrote	noted	0.70
wrote	added	0.69

I present the top 25 juxtapositions for the immigration topic in Table 5.5. I cherry-picked some of the most informative ones and presented them in Table 5.6. For the immigration topic, liberal sources and conservative sources have quite diverging and strong opinions. For example, when referring to immigrants, extreme liberal sources prefer the term ‘undocumented’ whereas extreme conservative sources prefer

the term 'illegal.' Extreme liberals focus on 'fees' paid for immigration purposes whereas extreme conservatives focus on 'costs' to the taxpayers.

Table 5.6. Example juxtapositions on the topic of immigration.

Left-Term	**Right-Term**
undocumented	illegal
women	men
people	Americans
republicans	democrats
flee	escape
fees	costs
death	murder
violence	violent
employees	workers
companies	businesses

I present the top 25 juxtapositions for the abortion topic in Table 5.7. I cherry-picked some of the most informative ones and presented them in Table 5.8. We also cherry-pick some informative juxtapositions of the topic of abortion in Table 5.8. It is interesting to see that the extreme liberal sources use the term 'women' often and the extreme conservatives focus on 'girls.' Another interesting observation is the gender-neutral usage of 'spokesperson' versus the use of 'spokesman.' A final example is the use of 'rights' (perhaps referring to women's rights) on the extreme liberal side and the use of 'freedom' (perhaps referring to religious freedom) on the extreme conservative side.

Table 5.7. Top 25 juxtapositions on the topic of abortion.

Left-Term	**Right-Term**	**Similarity**
argued	noted	0.74
argued	concluded	0.72
argued	stated	0.68
arguing	saying	0.63
backing	support	0.63
bills	legislation	0.63
christian	catholic	0.70
confirmed	reported	0.71
congressman	senator	0.70
congressman	governor	0.67
country	nation	0.78
early	late	0.71
gop	democratic	0.75
group	organization	0.65
groups	organizations	0.76
mcconnell	schumer	0.76
north	south	0.81
officials	authorities	0.67
pregnancies	abortions	0.65
prosecutors	authorities	0.67
republican	democratic	0.79
republican	democrat	0.74
study	report	0.62
women	girls	0.66
women	mothers	0.63

5.4 Content analysis and recommender system

Content analysis is motivated by the findings of simulation work in the previous chapter. Content-based recommender system utilizes the language and content features as input, to learn the personalized recommender models. In this chapter, we would like to understand the biases of content-based recommender through the content itself.

Table 5.8. Example juxtapositions on the topic abortion.

Left-Term	**Right-Term**
women	girls
women	mothers
anti	pro
contraception	abortion
nationalist	leftist
spokesperson	spokesman
rights	freedom

One of the biases in the simulation study is linguistic polarization. We observed a set of examples, e.g., 'undocumented immigrant' vs. 'illegal alien.', which could be easily well trained and identified by a content-based classifier. We also observed that such polarized pairs also have higher coefficients to determine the ranking decisions of the recommendations. The result of such bias is that the system may over-recommend articles under topics that satisfy users' preferences.

We also observe the homogenization effect in the content-based recommender systems. There are keywords overlapped across topics that can mislead the recommender. The first category of overlapped keywords is topic-irrelevant keywords. For example, in the juxtaposition analysis, I found that 'spokesperson' appears in the liberal articles, and 'spokesman' appears in the conservative articles. Imagine a user with liberal ideology on most topics but only has conservative views on gun control. The homogenization effect would treat 'spokesperson' as a positive indicator in gun controls articles. Therefore, the user would receive liberal articles about guns, the opposite of his political view. Another category of overlapped keywords cross topic

is topic-relevant keywords. For example, the word 'cost' is discussed primarily in the immigration and tax articles on the conservative side, but it also appears in the liberal articles about healthcare issues. The word itself is related to both topics but could be used under different contexts. The content-based recommender would not be able to differentiate the stance of 'cost' itself.

I provided the first steps in understanding and analyzing these phenomena based on the content analysis of the articles. We have observed polarized language usage in different topics that may lead to the bias of content-based recommender systems. Dealing with the linguistic polarization and homogenization effect is still an open research question. Choosing other recommender models, such as collaborative filtering, may avoid those biases. However, using collaborative filtering still has its own biases, such as 'subsumed' effect, as we discovered in the previous chapter.

5.5 Summary

This chapter conducts content analysis into the articles we collected on certain topics. Firstly, we examine the most informative words on a specific topic. Then we propose two different methods to analyze the language used on different political topics and stances. In the end, we propose the concept of juxtaposition words that pair the most closed words but in the opposite partisan sources.

For the single topic analysis, we found that the polarized words on the liberal and conservative sources always get higher weights, e.g., 'anti' vs. 'pro' in abortion. In the cross-topic analysis, there are over-lapped usage for certain languages in different topics and different source stances. It is easier to explain the most informative words and languages, but most of them need to be dug further within the context of usage. The juxtaposition approach is to be found more informative since the pre-condition is to fix the context by using the embedding method.

There are still some common limitations in the methods described in this chapter. 1) the scalability would be one of the concerns, especially for cross-topic analysis when involving more topic pairs; 2) we still need to cherry-pick most of the informative examples in our analysis. In the meantime, we are still seeing some redundant information in our ranked list. Therefore, one major future work would be how to process the analysis automatically and pick up the most informative language usage without eyeballing.

CHAPTER 6

USER STUDIES FOR FILTER BUBBLES IN NEWS RECOMMENDER SYSTEMS

The results of our simulation study show that filter bubbles vary significantly by different typologies and different recommender systems. In this chapter, I discuss the user study that we conducted to study the formation and evolution of filter bubbles. Furthermore, I would like to find a way to combat and alleviate the filter bubbles in recommender systems. Prior work to mitigate filter bubbles usually consider automated approaches to increase diversity or randomness to nudge users out of the bubbles. I adopted a similar idea in our simulation study by injecting randomness into a personalized model. However, this approach would assume users are passive to the reaction of the filter bubbles.

In this study, I assume that users can be empowered with greater transparency and with the autonomy to actively engage with and modify the filtering system. Transparency and interaction may be the solutions based on the previous works to combat filter bubbles but in different domains [102]. To investigate the effect of transparency and interaction on filter bubbles, I first build a content-based news recommender website, which has the options to provide the transparency and interaction of filter bubbles for different sessions. I then recruit over 800 U.S. participants into our preliminary demographic study. Finally, there are 102 U.S. participants, out of these 800 users, who participate in the full scale study. These users are assigned either to the control group or to treatment group randomly. The users in the treatment group can view their profiles described by the system anytime and adjust their political leaning and interests on various topics during the recommendation process. I investigate the following three research questions in this chapter.

- **How does transparency and interaction effect user engagement with the system?**

  I found that giving users more control in the recommendation would result in more engagement and likes by the users. Comparing the result between the control group and treatment group, it shows that users are more likely to agree and up-vote recommended articles in the treatment group, which results in a higher click-through rate comparing with the users in the control group.

- **How does transparency and interaction effect the filter bubbles?**

  Our analysis showed that, for the treatment group, the users on the extremes moved closer to the center by using the interaction tools, whereas the users in the center used the tools to move away from the center slightly. The users in the control group did not make big jumps one way or the other in the extremeness scale. We also observed that the variance of the change was larger for the treatment group, suggesting that while some users used to the interaction tool to receive less extreme content, others used it to consume more extreme content.

- **Does transparency lead to more awareness of filter bubbles?**

  Via the analysis of the post-questionnaires, I found that providing transparency (e.g., showing filter bubble metrics and status) during or after the recommending procedure helped the users realize that they might be trapped in filter bubbles. Our study showed that simple statistical graphs about users' recommender history would raise the awareness of filter bubbles. Further, the results showed that users in the treatment group felt more informed, compared to the control group, about how news recommender systems worked.

6.1 Dataset

I used the same dataset that I described in Chapter 3. In the original version of the data, we identified 14 political topics. For this chapter, I removed three topics. I removed *US 2020 election* and *Trump impeachment* because they were not recent events, and I removed *technology* because it did not have enough articles. The final list of topics we used are: *abortion, environment, foreign policy, gun control, healthcare, immigration, LGBTQIA, racism, taxes, trade*, and *welfare*.

I sampled 8,000 articles from each of the five political stances (i.e., -2, -1, etc.), which resulted in a total of 40,000 articles. The number of articles for each topic is summarized in Table 6.1. This data is used in two separate stages. The first stage is to bootstrap the personalized model for each user before the recommendation. The second stage is the pool of articles from which the recommender system could pick its recommendations. We sample 5,000 articles for bootstrap from all 40,000 articles. The remaining 35,000 articles are then used as candidates for recommendation. There are 44,033 topic labels in 40,000 articles. Note that an article can have multiple labels, such as both 'immigration' and 'racism.'

6.2 Demographic survey

Amazon Mechanical Turk (AMT)[2] is one of the largest crowd-sourcing websites which hires online workers to complete different tasks, such as labeling, survey, user study, etc. Therefore, we choose to use AMT to invite participants to our user study. Our user study consists of two steps: 1) demographic survey; 2) recommender system user study (more details are provided in Section 6.3). The purpose of the demographic survey is to: a) ensure there are no selective biases in our study, such as gender, income, education, race, etc.; b) we want to invite only the qualified U.S. Mturkers

[2]https://www.mturk.com/

Table 6.1. Topic distribution for the articles that are used in the user study.

Topic	**# articles**	**Topic**	**# articles**
abortion	1988	environment	2854
foreign policy	5759	guns	2781
healthcare	5999	immigration	5771
LGBTQIA	1611	racism	5550
taxes	4639	trade	3794
welfare	3287		
# articles	40000		
# labels	44033		

into our study. The demographic survey asks six questions: gender, age, race, self-identified political stance, education, and income. The questions and the options are as follows:

1. What is your gender?

   • Male • Female • Other

2. How old are you?

   • Under 20 • 20-29 • 30-39 • 40-49

   • 50-59 • 60-69 • Over 70

3. Would you describe yourself as (check all that apply)?

- American Indian/Native American
- Asian
- Black/African American
- Hispanic/Latino
- White/Caucasian
- Pacific Islander
- Other

4. Generally speaking, do you consider yourself to be a(n)
   - Democrat
   - Republican
   - Independent
   - Other
   - Don't know/Undecided
5. What is the highest level of education you have completed?
   - Elementary school
   - Middle school
   - High school
   - Some college
   - Bachelor's degree
   - Some graduate work
   - Completed Masters or professional degree
   - Advanced graduate work or PhD
6. What do you estimate your 2020 household income was?
   - Under $25,000
   - $25,000-$49,999
   - $50,000-$74,999
   - $75,000-$99,999
   - $100,000-$124,999
   - $125,000-$149,999
   - Over $150,000

Secondly, we also collect the answers about how the U.S. residents receive the political news and information, and how often they ever use some fact-checking websites. We ask the following three questions:

1. Where do you get most of your information about current news events (check all that apply)?

    - Print newspapers
    - Online newspapers
    - Print magazines
    - Online magazines
    - Other places on the Internet
    - TV
    - Radio
    - Facebook/Twitter/other social media
    - Family/friends/colleagues
    - Other source

2. How often do you read or watch news about U.S. politics, policies, or the economy?

    • Never • Rarely • Sometimes • Often • Always

3. How often do you use fact-checking websites (for example: PolitiFact, Snopes, FactCheck, etc.)?

    • Never • Rarely • Sometimes • Often • Always

Finally, we ask three questions that are related to the basic knowledge of U.S. politics. The questions are easy to answer if the AMT Turkers have the basic knowledge of U.S. politics. These questions are as follows:

1. Which of the following is the most conservative news source?
   - MSNBC
   - New York Times
   - Fox News
   - The Guardian
2. Among the following, who is the most liberal politician?
   - Ted Cruz
   - Bernie Sanders
   - Donald Trump
   - Lindsey Graham
3. Which state among the following recently enacted a restrictive abortion law?
   - Texas
   - Massachusetts
   - New York
   - California

We gradually released small batches of tasks and collect the responses on the AMT website. We require AMT annotators to be located in the USA and must have voted in the 2020 election. Overall, we collected over 850 responses from the initial survey stage. Our analysis shows that the distribution from AMT is almost uniform in each dimension. Our data shows that there are 51% females and 49% males. 38% of participants are self-identified as *Republican*; 41% are self-identified as *Democrat*; the rest 21% are self-reported as *Independent*. In terms of education level, over 47% of the participants have obtained bachelor's degrees; the rest reported education as high school, college-equivalent, and master's degrees.

For the question of *where to get most of the information about current news events*, 40% of people said they use online newspapers to receive political news information. The second significant source is print newspapers, resulting in 31%; the other major sources are the internet, T.V., and others. For the question of frequency on reading U.S. politics news, there are over 90% users are above the level of *sometimes*. Finally, for the question of fact-checking, 12% people never use it at all and 23% people rarely use it; the rest use fact-checking pages sometimes or more.

For the political literacy questions, around 54% people could answer three questions correctly; 16% people could answer two out of three correctly. The difficulties of the three questions are pretty similar to each other. The accuracies for the three questions are 68%, 69%, and 75%. We also find that master workers always have much better quality and accuracy than regular workers in the AMT. A qualified Mturker would at least answer 2 out of 3 political literacy questions. We select these qualified Mturkers for our next stage: the full-scale recommendation study.

6.3 Recommendation study

The selected users from Section 6.2 would use our designed recommender system hosted on a standalone website. Each user is provided with a unique username and password to log into our website. They first need to answer eleven questions to describe their ideologies and interests on different U.S. political topics. Then they would up-vote or down-vote at least thirty articles (we also provided *skip* option, but those skipped articles do not count toward the thirty article requirement.). In the end, the user will complete the required post-questionnaire and provide optional comments concluding the study.

6.3.1 Control group versus Treatment group. The selected participants use our designed website to proceed to the news recommender system. Since we want to evaluate how the transparency and interaction would affect the filter bubbles in the news recommender, we assign the users uniformly randomly into either a control group or a treatment group.

6.3.1.1 Control group. The participants assigned to the control group receive recommended news articles one by one. To receive the next recommendation, the user must click one of *up-vote, down-vote*, or *skip* buttons. The recommender is retrained after each click to re-rank articles and provide a new recommendation. There are no

transparency and interaction tools provided during the recommendation process.

6.3.1.2 Treatment group. The treatment we provided for the treatment group is the access to user-friendly interaction tools that the users may use to adjust their interests and ideologies during the recommendations. Specifically, the user can change their interests and switch to other ideologies or tell the system to receive more/fewer articles on specific topics based on the interaction.

One major component in the treatment group is that users can adjust their interests and political stances during the recommendations. Firstly, users in the treatment group can enter the interaction page anytime. The URL to the interaction page is on the top of every page. Secondly, users in the treatment group are automatically entered into the interaction page every five sequential articles or every three sequential down-voting in a row.

The header on the interaction pages would illustrate the purpose of tools and how to interact and adjust. Figure 6.1 is the interaction tools for the users in the treatment group. Both the 'Political stance' slider and 'Interest' slider represent the recommender system's current status. For the 'Political stance,' we choose to aggregate the top-ranked 100 articles and calculate the average partisan score for each topic; for the 'Interest,' we calculate the topic ratio in the top-ranked 1,000 articles. If the user is not satisfied with the status of the recommender, they can tune it via the slider bars.

We also provide the instructions on the page, such as the user may move the 'Political stance' slider to adjust the number of articles on a topic; meanwhile, the user could move the 'Interest' slider to adjust the ratio of articles on a topic. We also provide the function that users could revert to their original preferences based on their survey responses.

Once the user interacts with the sliders to adjust their preference, the system would use a binary search algorithm to update user's profile to meet the adjustment. Users may exit the interaction page and proceed to the recommendation page at any time.

6.3.2 Recommender system. We build a personalized news recommender system on our host server. The selected AMT is given a username and password to log into our website and is expected to finish the whole recommendation user study. Each user is randomly assigned to either the control group or the treatment group. To complete the profile, they first answer eleven topic questions as shown in Table 6.2. The responses range from "strongly agree" to "strongly disagree" in five discrete scales. There is one additional question where they are asked to indicate their interests (from a scale of 1 to 5) on each topic. Our system processes these responses and builds a personalized bootstrap model for each user. Then, the user is asked to up-vote/down-vote 30 sequential articles to finish the user study. In the end, the user needs to answer a short post-survey and leave any comments if desired. Next, I provide the details for the main components of the system.

6.3.2.1 Profile questions. To solve the cold-start problem, we ask each participant to answer their profile questions for each topic. The questions are originally from Pew surveys of U.S. political typologies [96]. The survey aims to identify different political typologies that may differ due to nuanced ideologies in different topics. We use these questions for the same purpose and profile each user in our recommender system. The questions are summarized in the Table 6.2. The question is to ask the participant as *To what extent do you agree or disagree with the following statement?* when providing the statement for each topic. User must select one of the options, which are *Strongly agree, Somewhat agree, Neither agree nor disagree, Somewhat disagree, Strongly disagree*. We then match the answers to these questions into the same scale

range from -2 to $+2$. After completing eleven questions, the user would also provide their interest on each topic. The scale of interests is from 1 to 5. Therefore, each user would have two profile vectors representing their stances on each topic u_s and their interests on each topic u_i.

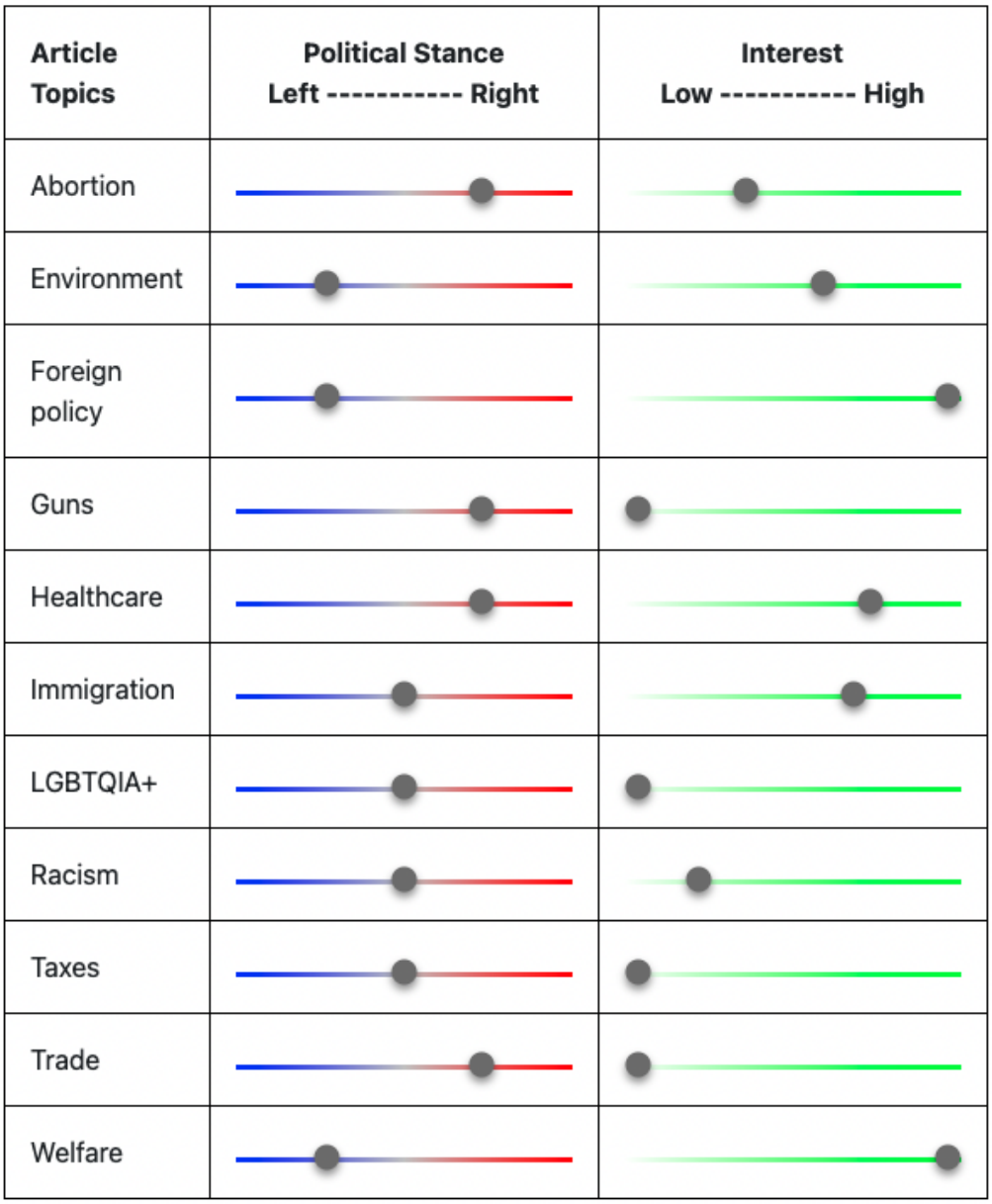

Figure 6.1. The interaction and transparency tools available only to the treatment group.

6.3.2.2 Bootstrap algorithm. Since we are building individual a content-based

Table 6.2. Topic profile questions.

Profile Questionnaires	
abortion	Abortion should be legal in most cases.
environment	Stricter environmental regulations and laws are worth the costs.
foreign policy	Good diplomacy is the best way for the U.S. to ensure peace.
guns	Gun laws should be stricter than they are today.
healthcare	Providing healthcare to Americans is the federal government's responsibility.
immigration	Immigrants strengthen the United States in many different ways.
LGBTQIA	Members of the LGBTIA+ community should have the right to marry.
racism	Changes are needed in American society to improve racial equality.
taxes	The U.S. economic system unfairly favors powerful interests.
trade	U.S. involvement in the global economy is good for the country.
welfare	Poor people have hard lives because government programs do not do enough for them.

model for each user, we need to bootstrap some training instances to train the initial model for each user. As we mentioned in the Section 6.1, we reserve 5,000 articles for bootstrap. All the individual models will draw the training samples from this pool.

The algorithm 2 illustrates how to build a bootstrap model for each user. The idea is to define the positive instances as the selected stance for each article, and vice versa. For example, if a user u choose -1 in abortion, the algorithm would draw 25 articles from topic abortion with partisan score -1 as positive, then draw 25 articles from topic abortion with partisan score $+2$ as negative. If a chosen stance is 0, the algorithm would draw the equal number of negative examples from both $+2$ and -2, but assign them instance weights of 0.5.

Algorithm 2 The pseudo-code of the bootstrap model.

Input: u_s – the user stance vector; U – 5,000 articles

Output: M – a trained bootstrap model

```
Denote D as an empty training set
for each topic t do
    Get stance s_t from u_s
    Draw N samples in topic t with stance s_t from U into D as positive instance
    Draw N samples in topic t with the most opposite of stance s_t from U into D as
    negative instance
end for
Train a SGD model M on D
return M
```

6.3.2.3 Ranking model. A personalized model should be able to rank a list of articles based on their profile and historical actions. Therefore, we chose to use a two-stage ranking method that can provide personalized ranking and adapt itself to the changes entered in the interaction page.

The first stage is to build a content-based model to deliver personalized recommendations for each user. Each article is transformed into a vector with 3,000 dimensions using Tf-idf. The content-based recommender is an Stochastic Gradient Descent (SGD) model that can be updated incrementally after each up-vote/down-vote interaction.

The second stage ranking is defined as follows. Based on the profile questions, each user has a stance vector u_s, and an interest vector u_i. The SGD model M first predicts the score of each article s_r. Let us denote that each article a has a topic vector a_t, a binary indicator for each topic, and a stance vector a_s, which has the same input range with u_s. We would like to use a second-ranking mechanism that is defined as

$$s = \lambda s_r + (1 - \lambda)(Sim(u_i, a_t) + Sim(u_s, a_s))/2 \tag{6.1}$$

We set the $\lambda = 0.4$ as we have observed in our preliminary studies that it would balance both classifier preference and user's profile. The benefit of such a two-staged ranking is to provide the ability for our interaction tools to affect the final ranking. The algorithm would first rank the articles and choose a random article from top-K articles as its recommendation. We used top-K instead of top-1 to provide more diversity during the recommendation process.

6.3.3 Sanity checking and attention system. Previous study [103] shows that good survey and experimental research requires more attention to questions and treatments. We also enable the attentional articles that are blended in our recommendation process. We select ten scientific articles which are irrelevant to U.S. politics. After 3 or 4 sentences, we insert an action sentence to ask the user to up-vote, down-vote, or skip that article. If the user failed on attention articles five times,

the user-study would end, and our analysis would not account for the response. As a result, we filtered out around 10% of responses that failed on the attentional checks.

6.3.4 Post-questionnaires and feedback. After finishing at least 30 recommended articles, users in both the control group and treatment group are asked to complete a post-questionnaire at the end. The design of post-questionnaires on two groups is a little different from each other as shown in Table 6.3.

Table 6.3. Post-questionnaires for the control and treatment groups.

Shared questions
Qa: The system presented articles to me that I enjoyed reading.
Qb: I was exposed to news articles that presented diverse political perspectives.
Qd: Having participated in this study, I feel more informed about how news recommender systems like this work.
Control group only
Qc: Based on the data information about the news I read, I was exposed to news articles with diverse political perspectives.
Treatment group only
Qe: I had more control of the news I read by making adjustments to my preferences (using the sliders on the "Preferences Page").
Qf: The news I read was more reliable because of the adjustments I made to my preferences (using the sliders on the "Preferences Page").

The users in the control group will answer a set of questions (Qa, Qb, Qc,

and Qe) sequentially. Specifically, after answering the Qb, we will show each user a histogram of the content of the news articles that were presented to the them. An example is shown in Figure 6.2. After they are presented the statistics for the articles that the recommender systems showed to them, they are asked to answer Qc (control group), which is similar to Qb but worded differently. This design aims to study whether the transparency would raise awareness about filter bubbles.

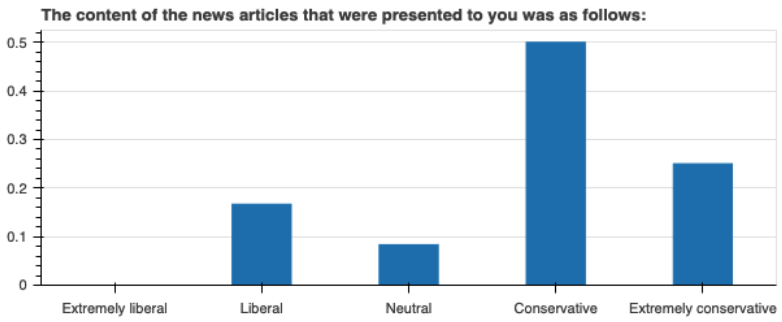

Figure 6.2. An example transparency figure available to the control group only after they answer Qb and before they answer Qc. This is specialized to each user based on what they are actually presented.

For the treatment group, we ask the same questions Qa, Qb, and Qd. Qc is different for the treatment group, asking whether they felt they had more control, and there was an addtional question, Qf, measuring if they felt the interaction tools helped with the reliability of the news they read. The user would give a scale from five to one to express how they agree with each statement.

6.4 User study results

We first discuss the research questions we proposed at the beginning of this chapter. Specifically, we would like to collect and analyze the data from our user study from three aspects:

- The full history of the recommended articles and the user's actions

- The status of personalized model M over time
- The answers to post-questionnaires

6.4.1 Problem formulation and metrics. Let U be a group of users. Each user u belongs to either the control group or the treatment group. Each user would be recommended N articles, one at a time. Each recommended article i would receive a response r_i, where $r_i = 1$ means the user up-votes the article, $r_i = 0$ means the user down-votes the article. The metrics would not consider the skipped articles[3]. We propose the following metrics to study the filter bubble metrics and the differences between the control and treatment groups.

6.4.1.1 Click-through rate. The click-through rate (CTR) is a common metric to measure engagement in a recommender system. The CTR is defined as the number of clicks on the recommended items divided by the number of recommended items that are shown to the users. The calculation of CTR is:

$$CTR = \frac{\sum(r_i)}{N}, 1 \leq i \leq N \tag{6.2}$$

6.4.1.2 Recommender system stance score. The recommender system stance score (RS-score) reflects the position of the recommender system at any time, as an average political stance of the articles it recommends. A recommender model R first ranks all the candidate articles. We pick the top K articles and take the average stance of these top K articles. Let $s(a_j) \in \{-2, -1, 0, +1, +2\}$ be the stance for article a_j. RS-score is calculated for every user at every step of the recommendation process. The recommender system score is defined as follows:

[3]We also considered using skipped articles as slightly-down-vote but left it as future work.

$$\overline{s} = \frac{\sum s(a_j)}{K}, 1 \leq i \leq K \tag{6.3}$$

6.4.1.3 Extremeness score. The extremeness score measures how extreme the recommendations to the user are. The extremeness score is a modified version of the RS-score that would take the absolute value of the partisan score. Therefore, the extremeness range should be from $[0, +2]$. There are two possible definitions to define extremeness. They are defined as:

$$s_{e1} = \frac{\sum |s(a_j)|}{K}, 1 \leq i \leq K \tag{6.4}$$

or

$$s_{e2} = \frac{|\sum s(a_j)|}{K}, 1 \leq i \leq K \tag{6.5}$$

The difference is s_{e1} is taking the absolute value the stance of the article first, and then taking the average, whereas s_{e2} is to take the average first, and then take the absolute value. Both metrics should have similar behaviors, but mathematically s_{e1} is no less than s_{e2}. We denote them as RS-Extreme-v1 and RS-Extreme-v2 respectively.

6.4.1.4 Normalized stance entropy. The stance spectrum in our study is from -2 to $+2$. Another measure of filter bubbles, in addition to the scores described above, is the entropy of the stances of the articles. Let p_i be the fraction of articles with stance i in the top K articles ranked by recommender model R at any time. Normalized stance entropy is the entropy of the stance distribution, normalized by $\log m$ so that the maximum value is 1. The value $m = 5$ represents the values of the five discrete stances we defined in this paper. The entropy is defined as:

$$entropy = \frac{-\sum_{i=1}^{m} p_i \log p_i}{\log m} \tag{6.6}$$

6.4.1.5 Normalized topic entropy. Another dimension of filter bubbles is with respect to the topics of the articles. Similar to normalized stance entropy, we also measure the diversity of topics. The metrics definition is similar with Equation 4.3, but $m = 11$ since there are eleven different U.S. topics. A high value of topic entropy would indicate a smaller topic filter bubble and vice versa.

6.4.1.6 Difference during recommendation. For all the metrics we introduced earlier, we would like to calculate how these metrics change over time, from the beginning of the recommendation process to the end of it, as the user interacts with the system. For example, CTR may be increasing in the control group but decreasing in the treatment group. However, each user would have a different CTR at the very beginning. It is not fair to only compare their initial CTR or ending CTR. It would make more sense to compare the change. Therefore, we would like to calculate each metric's difference during the recommendation. For CTR, we take the difference between the CTR of the last ten articles and the CTR of the first ten articles. For other metrics, we would like to calculate the difference of metrics between after the bootstrap (the model trained on bootstrap articles only) and after the final interaction (the final status of the model)

6.4.2 User stats. Eventually, 102 distinct users participated in the full scale recommender system study. The basic statistics for these users are presented in Table 6.4. There are 51 people in the control group and 51 people in the treatment group. Of these 102 users, 56 identified as male and 46 identified as female. The stats show that there are more self-identified Republicans than Democrats in our recommender system study, and the percentage of self-identified Independent is around 20%.

Table 6.4. User statistics in the control and treatment groups.

	Overall	**Control**	**Treatment**
male	56	26	30
female	46	25	21
Democrat	35	17	18
Republican	47	23	24
Independent	19	10	9
# users	102	51	51

We also calculated the user activities in the treatment group. There are mainly two statistics we want to calculate: 1) how many interactions did the user make on the interaction page? 2) how long did the user spend on the interaction page? We plot the histograms to answer these two questions in Fig. 6.3. First, we observe

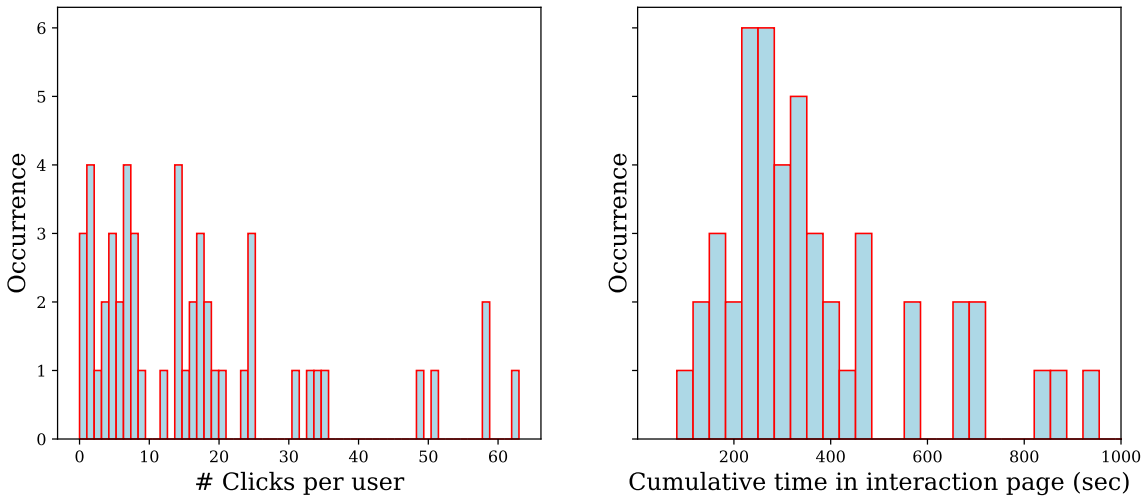

Figure 6.3. User activity in the treatment group.

that most people would use with the slider bar 10 to 20 times. There are only two out of fifty-one in the treatment group who did not use the slider bars at all. The second plot calculates the average time that a user spent on the interaction page. Most people spent about four minutes on the interaction page, and some users spent significantly more time.

6.4.3 User study results. We wanted to compare users in the treatment group to other users in the control group who had similar "beginning" scores. For example, to measure if the CTR increased or decreased for the treatment versus control group, we create groups of people in the treatment and the control group that have similar starting CTR values, and compare their metrics. To create such groups, we first rank all the users in the treatment group based on their initial CTR values. Using a sliding window of 10, we create several groups of 10 users. We create groups for the control group the same way. Then, we match a group in the treatment to the closest group in the control, and compare their relevant metrics. If a group of 10 users in the treatment cannot be matched to another group of 10 users in the control (or vice versa), that group is dropped from the comparison. Similar matching processes are carried out for the other metrics, such as RS-Score, extremeness measure, and so on. Algorithm 3 describes this process.

For each of the metrics, we create on figure with four subplots.The top of the plot shows the beginning score for each group. The next one shows the difference between the beginning score and the end score, as a box plot. The third subplot shows the median of the boxplots as lines. The final and the fourth subplot shows the variance of these metrics.

We first present the CTR metrics in Figure 6.4. There are 30 pairs found in this metric. From left to right, we group the people from lower CTR to higher CTR in both the control and treatment groups. The difference of CTR is calculated as the

Algorithm 3 The pseudo-code of grouping users into pairs.

Input: G_1 – the metric m result from group 1; G_2 – the metric m result from group 2; w – sliding window size

Output: P – a list of tuples (s_i, r_j)

```
G1 = sort(G1); G2 = sort(G2)
Slide G1 with size w resulted in G1* – a list of subsets {s1, s2, ..., sm}
Slide G2 with size w resulted in G2* – a list of subsets {r1, r2, ..., rn}
for each element si in G1* do
    Find the closed subset rj in G2*
    if rj's closed subset in G1* is si then
        P.add(si, rj)
    end if
end for
return P
```

CTR of the last ten recommended articles minus the CTR of the first ten articles for each separate user. The third subplot shows that for most pairs, the CTR in the treatment group is always getting higher than the users in the control group. The curves in the last several pairs are mixed together since their beginning CTR is already higher enough. This result shows that the interaction tools help the users find articles that are more likely to up-vote.

We next present the RS-score results in Figure 6.5. Again, there are 30 pairs between the control group and treatment group. The first subplot shows the average partisan score of the top 200 articles ranked by the recommender model before up-vote/down-vote actions. The y-axis indicates that the recommender system identifies the users from extreme liberal (around -2.0) to extreme conservative (around -1.2). The second subplot shows the difference of recommender system score, which is the average partisan score after recommendation minus the average partisan score before

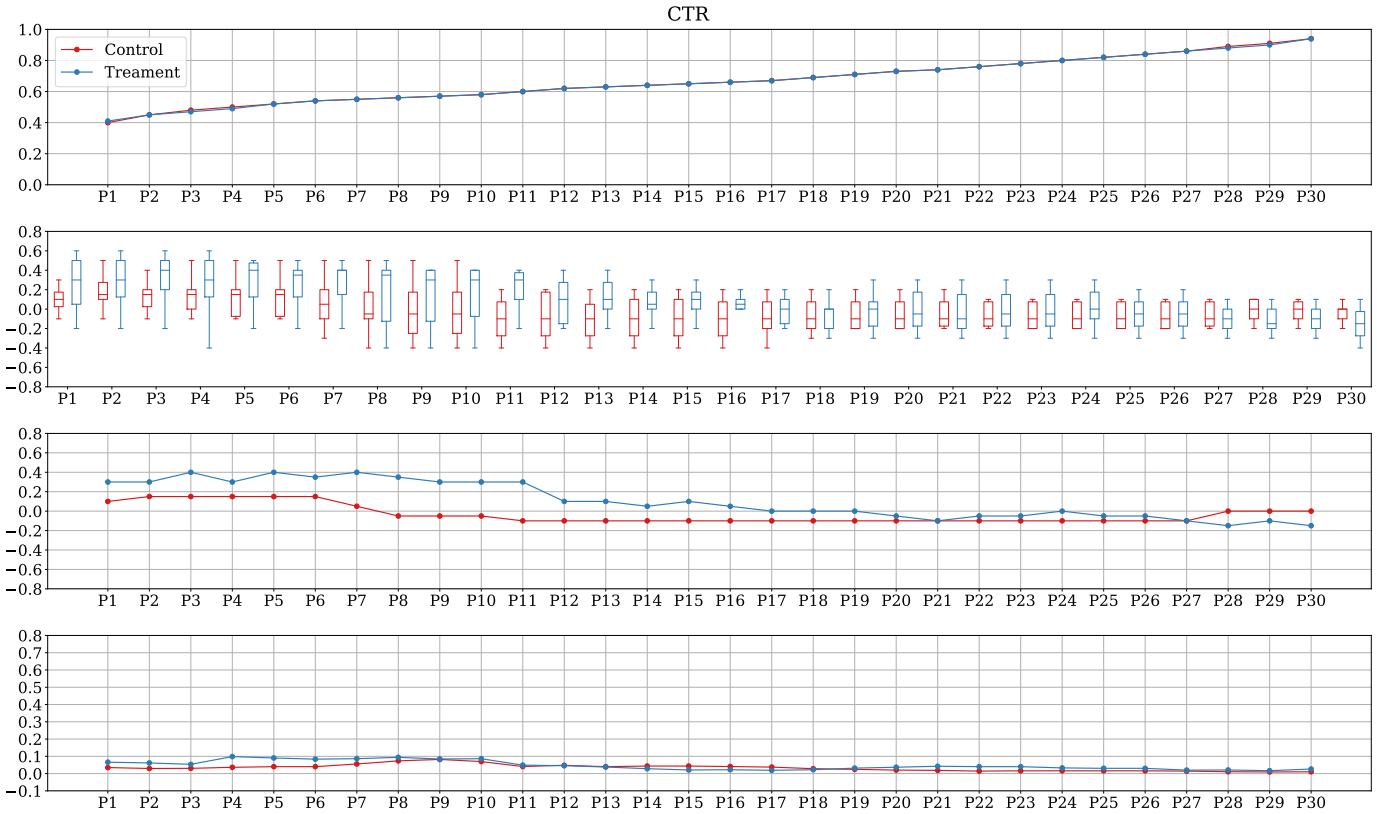

Figure 6.4. Click-through rate comparison between control and treatment groups. From top to bottom, the corresponding figure measures the following of each closest user subgroup pair: CTR in the first ten recommended articles; distributions of CTR difference during recommendation; median of CTR difference during recommendation; variance of CTR difference during recommendation.

the recommendation (the score in subplot 1). If the difference is negative, the recommender system pushes the user to the liberal side and vice versa. The third subplot is the line plot that connects the median value of the second subplot. Before the P22 pair, the difference in the treatment group is almost always higher than the difference in the control group. It should be noticed that before the P22 pair, the users are mostly liberals. This shows that the liberal users are moving to the conservative/-center side using the interaction tools. The same trend also shows the opposite after the P22 pair, which means the conservative users are using the interaction tools to move to the liberal/center side. The last subplot shows the variance in the treatment group is much higher than the variance in the control group. This is expected since the users are given more control in the treatment group during the recommendation procedure.

We also calculate and plot the similar figures for extremeness measure, topic entropy measure, and stance entropy measure. Figures 6.6 and 6.7 shows the similar trends except the range of y-axis for the starting value is different for both as expected. If the extremeness difference is higher or positive, it means the recommender system pushes the user to experience more extreme articles in the end. In both figures, the results show that the least extreme users use the interaction tools to move to slightly more extreme (a positive change of 0.5), whereas the change in the control group is close to zero for these pairs. For people who are in the middle spectrum of the extreme, the change is close to zero for the treatment group, whereas the change is negative (becoming less extreme) for the control group. For the most extreme group, the change is close to zero for both treatment and control.Finally, the variance of extremeness difference in the treatment group is much larger than the users in the control group. This indicates that, while the results are averaged out over groups, some users in the treatment group are taking the model to less extreme while others are taking to more extreme.

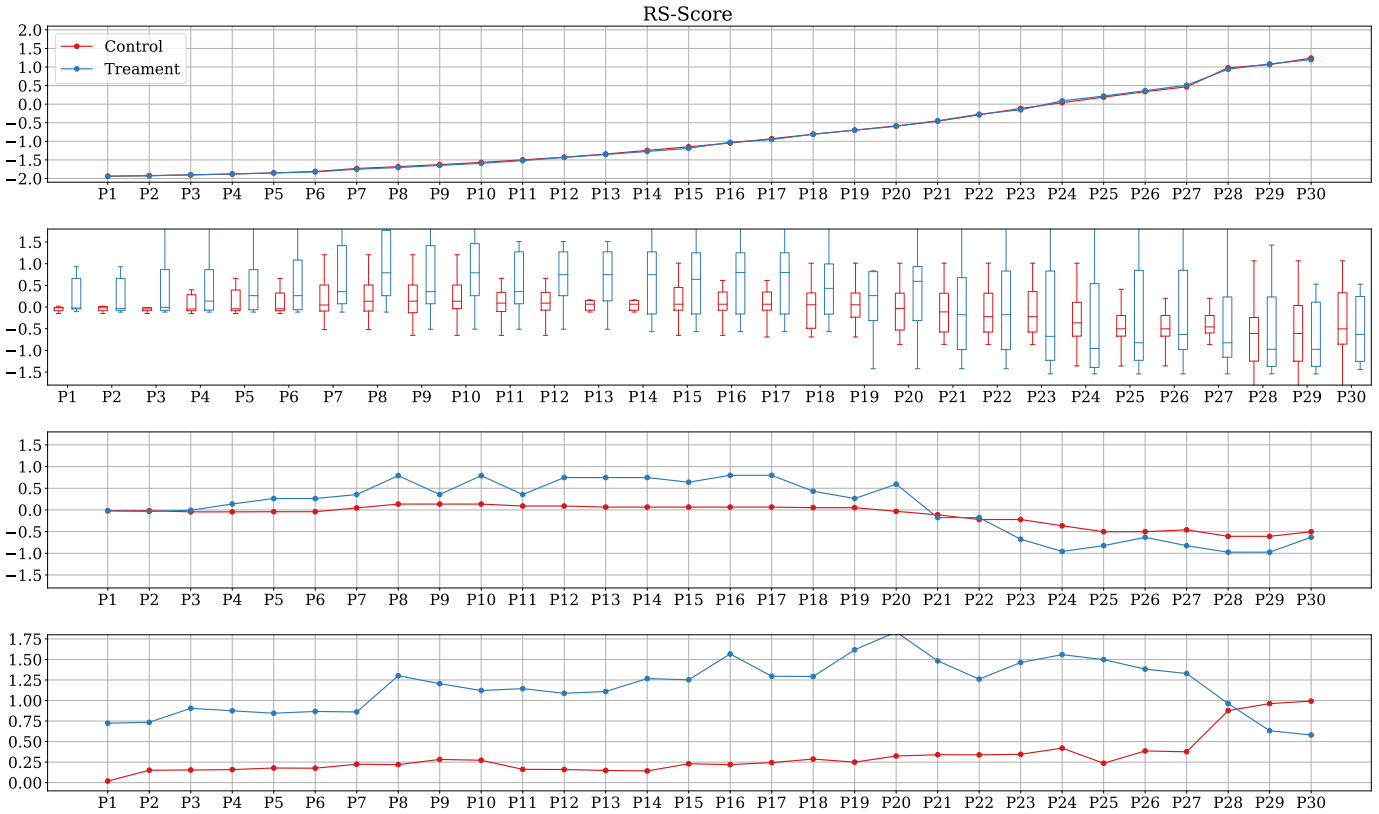

Figure 6.5. Recommender system score comparison between control and treatment groups. From top to bottom, the corresponding figure measures the following of each closest user subgroup pair: RS-score right after bootstrap; distributions of RS-score difference during recommendation; median of RS-score difference during recommendation; variance of RS-score difference during recommendation.

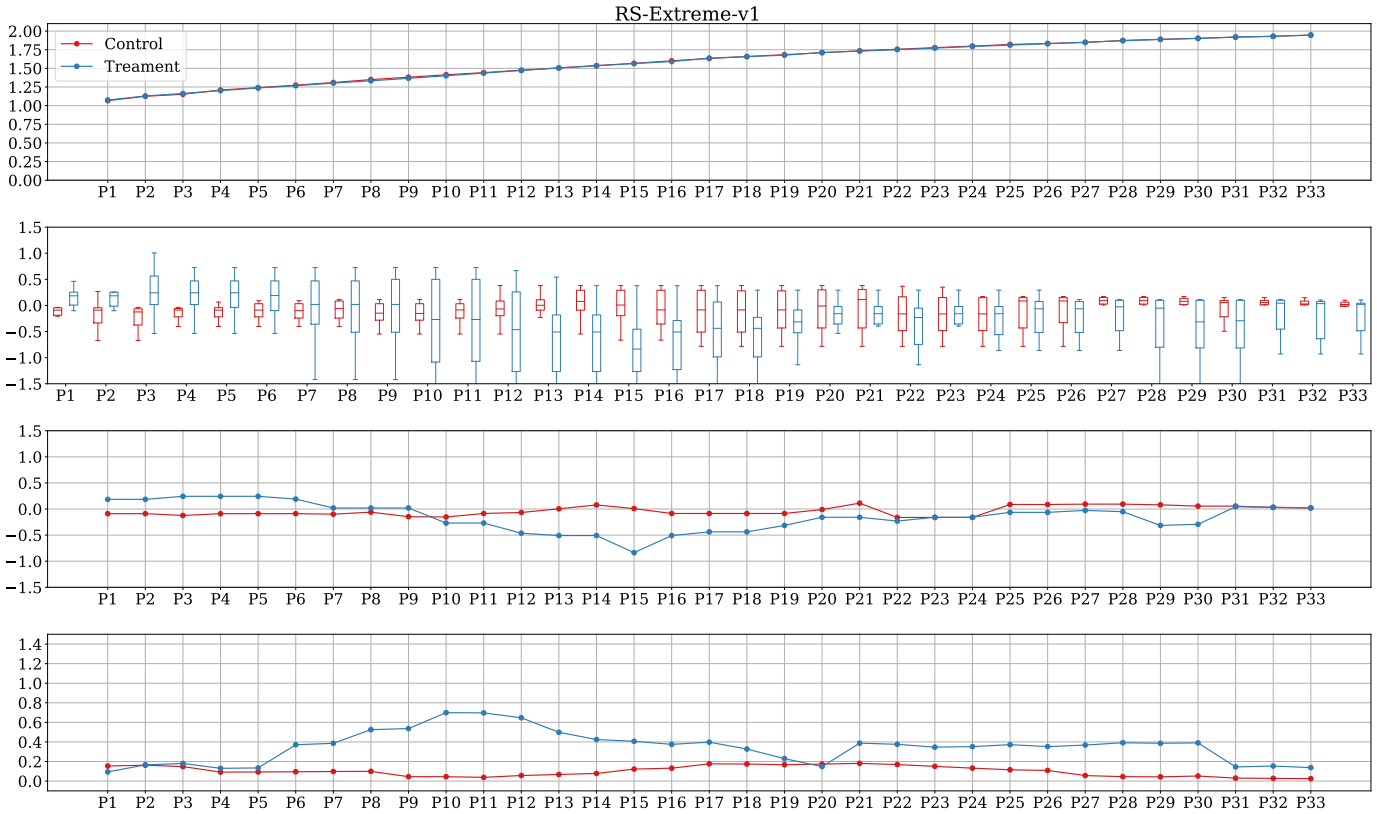

Figure 6.6. RS-Extreme-v1 comparison between treatment and control groups. From top to bottom, the corresponding figure measures the following of each closest user subgroup pair: RS-Extreme-v1 right after bootstrap; distributions of RS-Extreme-v1 difference during recommendation; median of RS-Extreme-v1 difference during recommendation; variance of RS-Extreme-v1 difference during recommendation.

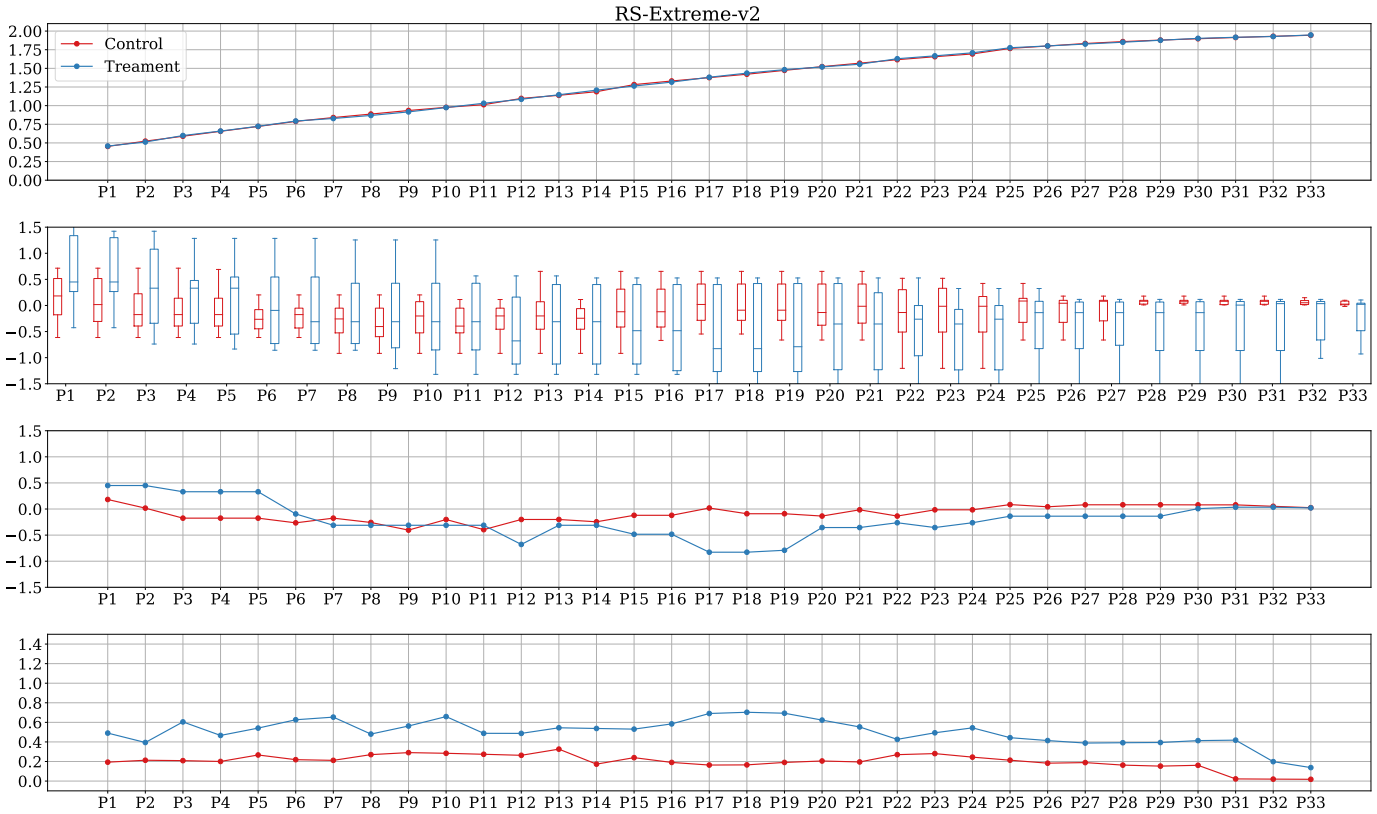

Figure 6.7. RS-Extreme-v2 comparison between treatment and control groups. From top to bottom, the corresponding figure measures the following of each closest user subgroup pair: RS-Extreme-v2 right after bootstrap; distributions of RS-Extreme-v2 difference during recommendation; median of RS-Extreme-v2 difference during recommendation; variance of RS-Extreme-v2 difference during recommendation.

Finally, Figures 6.8 and 6.9 present the results for the topic entropy and stance entropy respectively. The results are pretty similar for these measures. The changes are largely negative for both the treatment and the control group, showing that the users are taking the models to less extreme. The variances are also similar for both groups.

6.4.4 Post questionnaire analysis. After the recommendation, each user was asked to answer a set of questions, described earlier in Table 6.3. This section compares the results of shared questions between the control and the treatment groups. We also compare the responses to questions Qb before transparency graphs and to Qc after the transparency graphs in the control group. Finally, we discuss the results of Qe AND Qf for the treatment group.

Figure 6.10 shows the results for "Qa: The system presented articles to me that I enjoyed reading." for both the control and the treatment groups. The box plots on the left show that both the median and the average of the responses are extremely close for both groups. However, diving deeper into the data through histograms tell us a different story. The control group responded with "strongly agree" (5) more than the treatment group, and the treatment group responded with "agree" (4) more than the control group. The variance for the treatment group is also high. Note that the CTR increased more for the treatment group than for the control group compared to the beginning of the recommender system. Hence, this is a bit surprising result. One possible explanation is that giving finer grained controls to the treatment group allowed them to find more articles that they liked, but this did not necessarily increase their satisfaction with the system.

Figure 6.11 shows the results for "Qb: I was exposed to news articles that presented diverse political perspectives." for both the control and the treatment groups. Again, the mean and median responses are pretty similar for both groups.

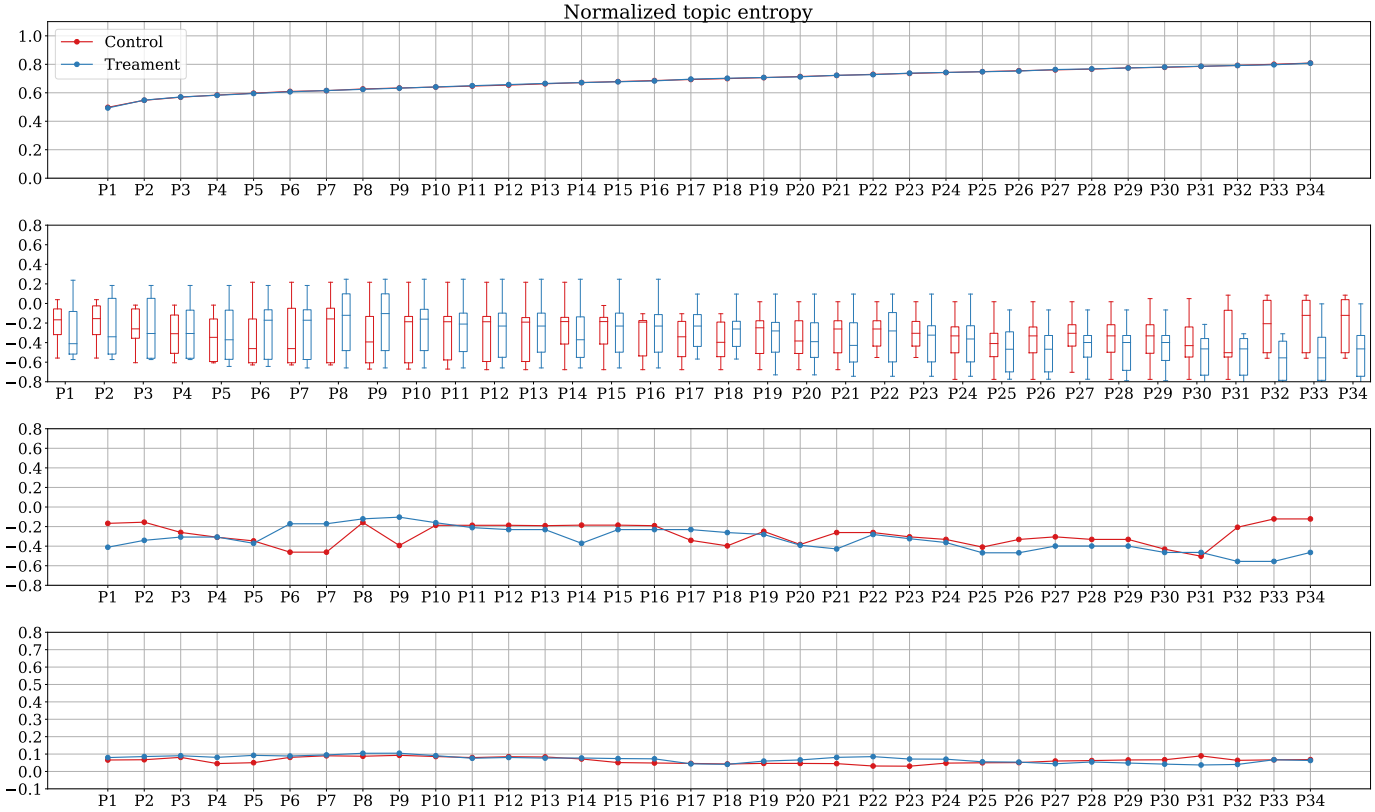

Figure 6.8. Normalized topic entropy comparison between treatment and control groups. From top to bottom, the corresponding figure measures the following of each closest user subgroup pair: Normalized topic entropy right after bootstrap; distributions of Normalized topic entropy difference during recommendation; median of Normalized topic entropy difference during recommendation; variance of Normalized topic entropy difference during recommendation.

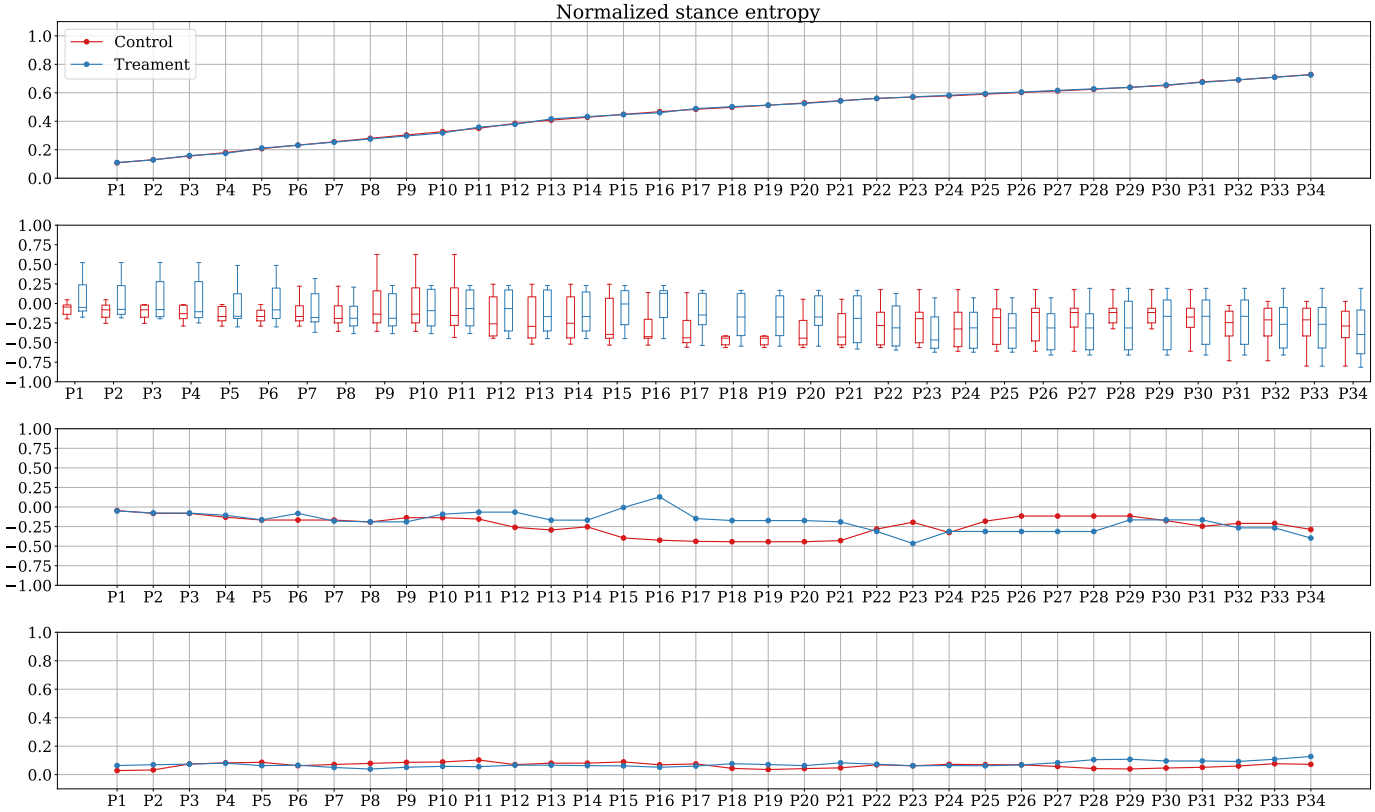

Figure 6.9. Normalized stance entropy comparison between treatment and control groups. From top to bottom, the corresponding figure measures the following of each closest user subgroup pair: Normalized stance entropy right after bootstrap; distributions of Normalized stance entropy difference during recommendation; median of Normalized stance entropy difference during recommendation; variance of Normalized stance entropy difference during recommendation.

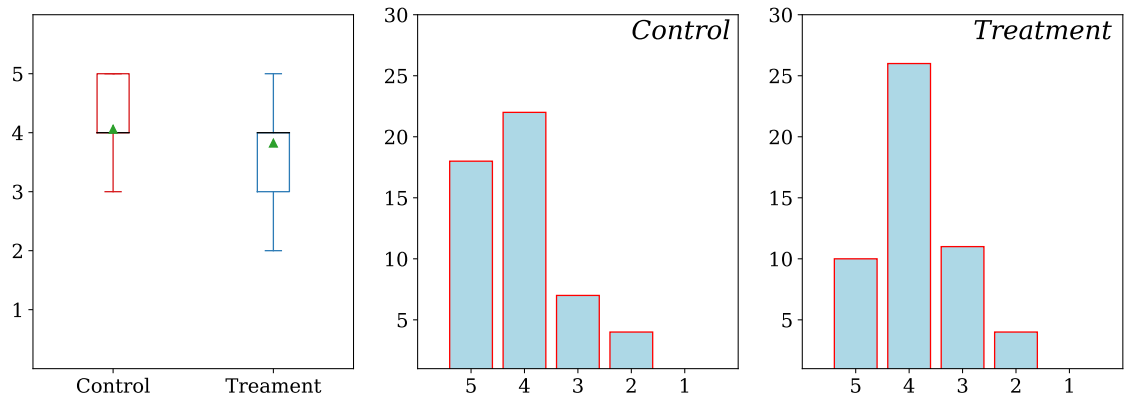

Figure 6.10. Qa: The system presented articles to me that I enjoyed reading. 5: Strongly agree, 1: strongly disagree.

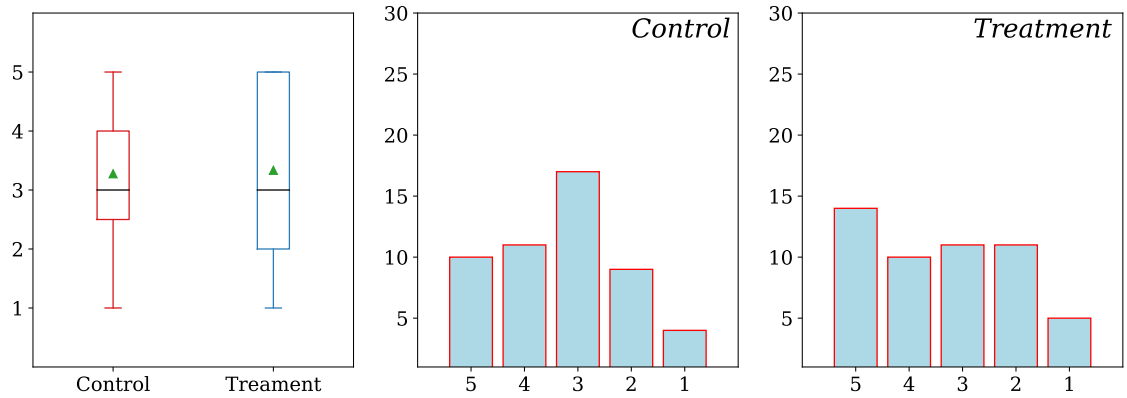

Figure 6.11. Qb: I was exposed to news articles that presented diverse political perspectives. 5: strongly agree, 1: strongly disagree.

The distributions are slightly different; the treatment group has slightly more people who strongly agree with the provided statement, but overall has an even distribution, whereas for the treatment group, the mass is centered in neither agree nor disagree response.

Figure 6.12 shows the results for "Qd: Having participated in this study, I feel more informed about how news recommender systems like this work." for both the control and the treatment groups. The results shows that the users in the treatment have a much larger share of users who strongly agree with this statement. Also, the variance in the treatment group is smaller than the variance in the control group.

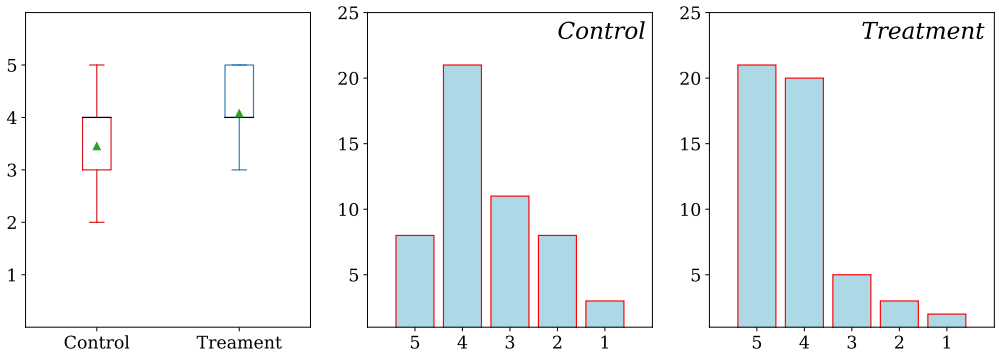

Figure 6.12. Qd: Having participated in this study, I feel more informed about how news recommender systems like this work. 5: strongly agree, 1: strongly disagree.

This result indicates that the users will likely understand how a recommender system works if we could provide more controls and transparency.

Figure 6.13 compares, for the control group, the results of Qb "I was exposed to news articles that presented diverse political perspectives." before they are shown the statistics about the news articles that they were presented (e.g., Figure 6.2), and Qc "Based on the data information about the news I read, I was exposed to news articles with diverse political perspectives." after they are shown the transparency graph. For

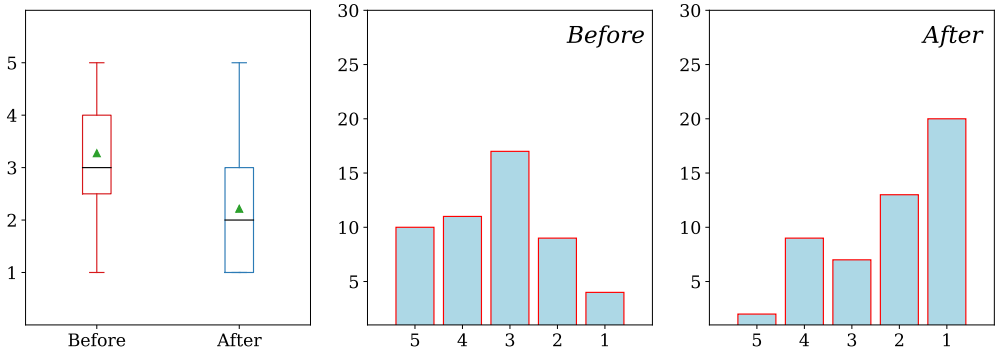

Figure 6.13. Qc: Based on the data information about the news I read, I was exposed to news articles that presented diverse political perspectives. 5: strongly agree, 1: strongly disagree.

this comparison, two samples are from the same people in the control group. The users thought, before they are shown the transparency graph, that they were exposed to diverse articles agreeing with the statement of Qb. However, after they are presented

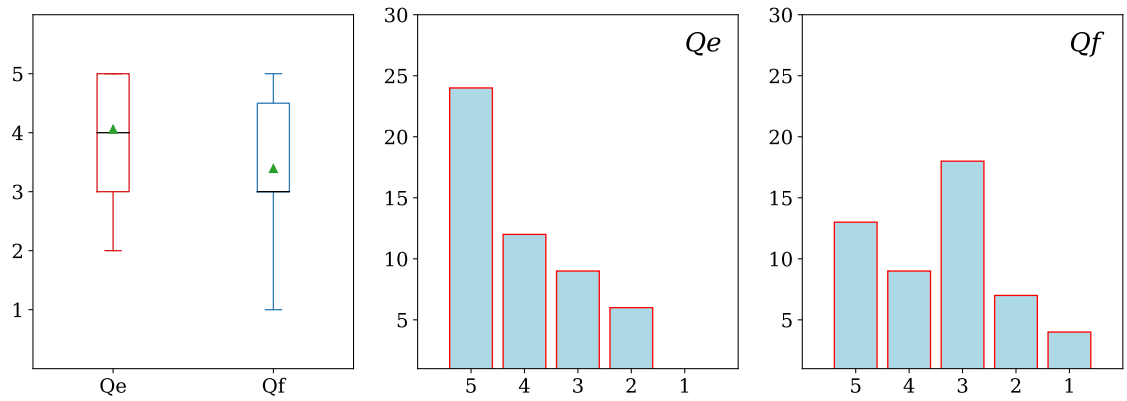

Figure 6.14. Qe: I had more control of the news I read by making adjustments to my preferences (using the sliders on the Preferences Page). Qf: The news I read was more reliable because of the adjustments I made to my preferences (using the sliders on the Preferences Page). 5: strongly agree, 1: strongly disagree.

with the transparency graph, their views have changed and they disagreed, most of the times strongly, with the statement of Qc. This result shows that the transparency can help significantly for helping people realize that the recommender system created a filter bubble for them.

Figure 6.14 summarizes the responses to Qe "I had more control of the news I read by making adjustments to my preferences (using the sliders on the Preferences Page" and to Qf "The news I read was more reliable because of the adjustments I made to my preferences (using the sliders on the Preferences Page)" for users in the treatment group. The results show that users largely agree with the statement of Qe, acknowledging the power and control the interaction tools provided to them. The responses to Qf are mixed, however; it is possible that either the tools were not sufficient to take them out of filter bubbles, or they did not use them to get out of filter bubbles.

6.5 Summary

In this chapter, I extended our work from simulations to user studies for a news recommender system. I developed a news recommender website where users

could participate in the user study. I proposed the combination of transparency and interactions as a possible treatment to alleviate filter bubbles. The comparison of filter bubble metrics and the responses to the post-questionnaires showed that a) transparency and interaction increased user engagement, resulting increased up-voting of articles for the treatment group; b) the interaction tools had the potential to alleviate filter bubbles for the people in the extremes, but also allowed the people in the center to move to more extremes, and c) transparency raised awareness of the filter bubbles.

One of the limitations of this work is our recruited participants did not cover all the political spectrum from liberals to conservatives. The users we recruited ended up being more on the liberal side than on the conservative side, based on their responses to the 11 political questions that were posed to them at the beginning of the study. However, this distribution did not align with the distribution of self-identification responses provided by the users in the demographic survey. Therefore, we have more results for the liberal spectrum than the conservative spectrum.

CHAPTER 7

CONCLUSIONS

In this thesis, I studied filter bubbles in political news recommender systems. First, I collected over 900K news, of which more than 100K were labeled with one or more of 14 political topics. Then, I analyzed the formation and evolution of filter bubbles in news recommendation through extensive simulations. Based on the findings of these simulations, I conducted a content analysis of the news articles. Finally, I proposed to provide transparency and interaction as a possible mechanism to combat filter bubbles and I studied its effects through a user study.

In Chapter 4, I have presented several simulation models to understand the relationship between political typology and news recommendation algorithms. I find that users with more extreme views tend to be easier for recommendation systems to model and thus enjoy higher click-through rates. However, this is only possible with less diverse recommendations regarding political views and topics. Furthermore, I find that two common classes of recommendation algorithms, content-based and collaborative filtering, can result in filter bubbles, though of different types and for different reasons. Finally, I find that users with heterogeneous preferences tend to be recommended articles that reflect more homogeneous viewpoints. These results suggest that future work in news article recommendations should consider a wider range of metrics when measuring diversity and a wider range of user preferences.

In Chapter 5, I have presented an analysis to understand how the political articles are constructed on different topics with different political stances. I conducted single topic analyses, cross-topic analyses, and juxtaposition of similar words that are used by opposing political views. This chapter highlighted the potential dangers of

using off-the-shelf content-based recommenders that can pick up on topic-irrelevant signals to make incorrect recommendations and lead to homogenization of recommendations where people with mixed views are pulled either to the left or to the right.

In Chapter 6, I designed a news article website that provides a content-based news recommender system. I recruited over 800 people into our demographic survey and invited 102 U.S. participants to the news recommender website. The users are split into control and treatment groups, which could interact with internal tools to adjust their ideology and interests in the recommended articles. Our analysis shows that users would engage more if they were provided with more user-controlled tools during the recommendation. The engagement is not necessarily equivalent to receiving more articles from extreme sources. As expected, while some users used the tools to get out of filter bubbles, others used the tools to move to extremes. Responses to post-questionnaires showed that transparency would raise the awareness about filter bubbles that the users did not know that they were in. Finally, users who were exposes to transparency and interaction tools expressed a higher understanding of how these models work.

7.1 Future work

Our work can be extended in at least two aspects:

- **Diversity of recommendation systems in the user study:** Our simulation study compared the collaborative filtering and content-based models with the oracle model to analyze the interaction between political types and filter bubbles. In our user study, I deployed only a content-based system on our website. User studies with different kinds of recommenders, such as collaborative filtering, can potentially result in different results, as the simulations have

shown.

- **Transparency effects in article level:** In our user study, I found that transparency is helpful to help users to understand how the system works and it also raises awareness about filter bubbles. The transparency that I provided was an aggregate statistics of what the user has been presented so far. Providing more fine-grained transparency and perhaps for each recommendation explaining why that article was recommended can further increase awareness about filter bubbles.

7.2 Conclusion

This thesis investigates the filter bubbles in the political news recommender system in different aspects. I designed simulations under different recommender models to analyze the interactions between political ideologies and filter bubbles. I analyzed the content and language usage in our collected labeled dataset across different topics from different political leaning sources. Finally, I designed a content-based news recommendation website that enables interaction functions, which involved over 100 U.S. participants. Our research shows that transparency and interaction would help users to a) understand the existence of filter bubbles b) improve engagement and alleviate filter bubbles.

Filter bubbles isolate users' online experience and the information they receive. Getting stuck in their own bubbles may amplify the negative impact and cognitive biases. With the rise in the use of AI-enabled recommendations in human life, it is crucial to understand the formation of filter bubbles in different domains. More importantly, finding effective treatments to combat the filter bubbles in the real world is even more challenging. In this thesis, I take several primary steps that are proven to lead towards more understanding of filter bubbles in the area of political news. Firstly,

our study provides an initial approach to measuring and identifying filter bubbles in the political news recommending system. Secondly, I found that randomness is one of the easiest and the most effective ways to increase the diversity of news. Further, I showed that providing transparency and interaction with the users is a more proactive approach. Granting sufficient control to users would increase the engagement and raise the awareness of the bubble formation. Finally, I believe the solutions adopted in our studies are not limited to the news articles domain, and can be easily extended to quantify and tackle the filter bubbles in many other fields, such as information retrieval systems.

BIBLIOGRAPHY

[1] P. Sen, G. Namata, M. Bilgic, L. Getoor, B. Galligher, and T. Eliassi-Rad, "Collective classification in network data," *AI magazine*, vol. 29, no. 3, pp. 93–93, 2008.

[2] A. Culotta and J. Cutler, "Mining brand perceptions from twitter social networks," *Marketing science*, vol. 35, no. 3, pp. 343–362, 2016.

[3] P. Liu, J. Guberman, L. Hemphill, and A. Culotta, "Forecasting the presence and intensity of hostility on instagram using linguistic and social features," in *Proceedings of the 12th International Conference on Web and Social Media*, 2018.

[4] B. H. Juang and L. R. Rabiner, "Hidden markov models for speech recognition," *Technometrics*, vol. 33, no. 3, pp. 251–272, 1991.

[5] A. Graves, A. R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *2013 IEEE international conference on acoustics, speech and signal processing*, pp. 6645–6649, IEEE, 2013.

[6] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 20, no. 1, pp. 23–38, 1998.

[7] H. Jiang and E. Learned-Miller, "Face detection with the faster R-CNN," in *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pp. 650–657, IEEE, 2017.

[8] H.-T. Cheng, L. Koc, J. Harmsen, T. Shaked, T. Chandra, H. Aradhye, G. Anderson, G. Corrado, W. Chai, M. Ispir, *et al.*, "Wide & deep learning for recommender systems," in *Proceedings of the 1st workshop on deep learning for recommender systems*, pp. 7–10, 2016.

[9] H.-J. Xue, X. Dai, J. Zhang, S. Huang, and J. Chen, "Deep matrix factorization models for recommender systems.," in *IJCAI*, vol. 17, pp. 3203–3209, Melbourne, Australia, 2017.

[10] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Computer networks and ISDN systems*, vol. 30, no. 1-7, pp. 107–117, 1998.

[11] T. Graepel, J. Q. Candela, T. Borchert, and R. Herbrich, "Web-scale bayesian click-through rate prediction for sponsored search advertising in microsoft's bing search engine," in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, pp. 13–20, 2010.

[12] I. Kononenko, "Machine learning for medical diagnosis: history, state of the art and perspective," *Artificial Intelligence in medicine*, vol. 23, no. 1, pp. 89–109, 2001.

[13] M. F. Akay, "Support vector machines combined with feature selection for breast cancer diagnosis," *Expert systems with applications*, vol. 36, no. 2, pp. 3240–3247, 2009.

[14] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3354–3361, IEEE, 2012.

[15] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, "Deepdriving: Learning affordance for direct perception in autonomous driving," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2722–2730, 2015.

[16] S. Stolfo, D. W. Fan, W. Lee, A. Prodromidis, and P. Chan, "Credit card fraud detection using meta-learning: Issues and initial results," in *AAAI-97 Workshop on Fraud Detection and Risk Management*, pp. 83–90, 1997.

[17] S. Panigrahi, A. Kundu, S. Sural, and A. K. Majumdar, "Credit card fraud detection: A fusion approach using dempster–shafer theory and bayesian learning," *Information Fusion*, vol. 10, no. 4, pp. 354–363, 2009.

[18] E. Pariser, *The filter bubble: What the Internet is hiding from you.* Penguin UK, 2011.

[19] G. S. Sanders and B. Mullen, "Accuracy in perceptions of consensus: Differential tendencies of people with majority and minority positions," *European journal of social psychology*, vol. 13, no. 1, pp. 57–70, 1983.

[20] R. Epstein and R. E. Robertson, "The search engine manipulation effect (seme) and its possible impact on the outcomes of elections," *Proceedings of the National Academy of Sciences*, vol. 112, no. 33, pp. E4512–E4521, 2015.

[21] D. Baldassarri and A. Gelman, "Partisans without constraint: Political polarization and trends in american public opinion," *American Journal of Sociology*, vol. 114, no. 2, pp. 408–446, 2008.

[22] J. N. Druckman and A. Lupia, "Using frames to make scientific communication more effective," *The Oxford handbook of the science of science communication*, pp. 243–252, 2017.

[23] A. M. McCright and R. E. Dunlap, "The politicization of climate change and polarization in the american public's views of global warming, 2001–2010," *The Sociological Quarterly*, vol. 52, no. 2, pp. 155–194, 2011.

[24] C. G. Rodriguez, J. P. Moskowitz, R. M. Salem, and P. H. Ditto, "Partisan selective exposure: The role of party, ideology and ideological extremity over time.," *Translational Issues in Psychological Science*, vol. 3, no. 3, p. 254, 2017.

[25] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *Acm transactions on interactive intelligent systems (tiis)*, vol. 5, no. 4, pp. 1–19, 2015.

[26] X. He and T.-S. Chua, "Neural factorization machines for sparse predictive analytics," in *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*, pp. 355–364, 2017.

[27] A. S. Lampropoulos, P. S. Lampropoulou, and G. A. Tsihrintzis, "A cascade-hybrid music recommender system for mobile services based on musical genre classification and personality diagnosis," *Multimedia Tools and Applications*, vol. 59, no. 1, pp. 241–258, 2012.

[28] F. Wu, Y. Qiao, J.-H. Chen, C. Wu, T. Qi, J. Lian, D. Liu, X. Xie, J. Gao, W. Wu, *et al.*, "Mind: A large-scale dataset for news recommendation," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 3597–3606, 2020.

[29] L. Gou, F. You, J. Guo, L. Wu, and X. Zhang, "Sfviz: interest-based friends exploration and recommendation in social networks," in *Proceedings of the 2011 Visual Information Communication-International Symposium*, pp. 1–10, 2011.

[30] M. J. Pazzani and D. Billsus, "Content-based recommendation systems," in *The adaptive web*, pp. 325–341, Springer, 2007.

[31] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proceedings of the 10th international conference on World Wide Web*, pp. 285–295, 2001.

[32] Z. Zhang, Y. Kudo, and T. Murai, "Neighbor selection for user-based collaborative filtering using covering-based rough sets," *Annals of Operations Research*, vol. 256, no. 2, pp. 359–374, 2017.

[33] A. Cichocki and A.-H. Phan, "Fast local algorithms for large scale nonnegative matrix and tensor factorizations," *IEICE transactions on fundamentals of electronics, communications and computer sciences*, vol. 92, no. 3, pp. 708–721, 2009.

[34] H. Abdollahpouri, R. Burke, and B. Mobasher, "Managing popularity bias in recommender systems with personalized re-ranking," *arXiv preprint arXiv:1901.07555*, 2019.

[35] H. Abdollahpouri, R. Burke, and B. Mobasher, "Controlling popularity bias in learning-to-rank recommendation," in *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pp. 42–46, 2017.

[36] H. Abdollahpouri, "Popularity bias in ranking and recommendation," in *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 529–530, 2019.

[37] G. Adomavicius, J. Bockstedt, S. Curley, and J. Zhang, "De-biasing user preference ratings in recommender systems," in *RecSys 2014 Workshop on Interfaces and Human Decision Making for Recommender Systems (IntRS 2014)*, pp. 2–9, Citeseer, 2014.

[38] G. Adomavicius, J. C. Bockstedt, S. P. Curley, and J. Zhang, "Do recommender systems manipulate consumer preferences? a study of anchoring effects," *Information Systems Research*, vol. 24, no. 4, pp. 956–975, 2013.

[39] S. Dumais, T. Joachims, K. Bharat, and A. Weigend, "Sigir 2003 workshop report: implicit measures of user interests and preferences," in *ACM SIGIR Forum*, vol. 37, pp. 50–54, ACM New York, NY, USA, 2003.

[40] W. Chu and S.-T. Park, "Personalized recommendation on dynamic content using predictive bilinear models," in *Proceedings of the 18th international conference on World wide web*, pp. 691–700, 2009.

[41] C. C. Johnson, "Logistic matrix factorization for implicit feedback data," *Advances in Neural Information Processing Systems*, vol. 27, p. 78, 2014.

[42] J. L. Freedman and D. O. Sears, "Selective exposure," in *Advances in experimental social psychology*, vol. 2, pp. 57–97, Elsevier, 1965.

[43] Z. Kunda, "The case for motivated reasoning.," *Psychological bulletin*, vol. 108, no. 3, p. 480, 1990.

[44] F. Bacon, "Three steps toward a theory of motivated political reasoning," *Elements of reason: Cognition, choice, and the bounds of rationality*, vol. 183, 2000.

[45] D. M. Kahan, "The politically motivated reasoning paradigm, part 1: What politically motivated reasoning is and how to measure it," *Emerging trends in the social and behavioral sciences: An interdisciplinary, searchable, and linkable resource*, pp. 1–16, 2015.

[46] T. Bolsen, J. N. Druckman, and F. L. Cook, "The influence of partisan motivated reasoning on public opinion," *Political Behavior*, vol. 36, no. 2, pp. 235–262, 2014.

[47] S. Mukerjee and T. Yang, "Choosing to avoid? a conjoint experimental study to understand selective exposure and avoidance on social media," *Political Communication*, vol. 0, no. 0, pp. 1–19, 2020.

[48] P. Barberá, J. T. Jost, J. Nagler, J. A. Tucker, and R. Bonneau, "Tweeting from left to right: Is online political communication more than an echo chamber?," *Psychological science*, vol. 26, no. 10, pp. 1531–1542, 2015.

[49] E. Bakshy, S. Messing, and L. A. Adamic, "Exposure to ideologically diverse news and opinion on facebook," *Science*, vol. 348, no. 6239, pp. 1130–1132, 2015.

[50] G. Eady, J. Nagler, A. Guess, J. Zilinsky, and J. A. Tucker, "How many people live in political bubbles on social media? evidence from linked survey and twitter data," *SAGE Open*, vol. 9, no. 1, p. 2158244019832705, 2019.

[51] S. J. Min and D. Y. Wohn, "All the news that you don't like: Cross-cutting exposure and political participation in the age of social media," *Computers in Human Behavior*, vol. 83, pp. 24 – 31, 2018.

[52] A. Guess, "(almost) everything in moderation: New evidence on americans' online media diets.," 2018.

[53] A. Bessi, F. Zollo, M. Del Vicario, M. Puliga, A. Scala, G. Caldarelli, B. Uzzi, and W. Quattrociocchi, "Users polarization on facebook and youtube," *PloS one*, vol. 11, no. 8, p. e0159641, 2016.

[54] M. A. Shapiro and H. W. Park, "More than entertainment: Youtube and public responses to the science of global warming and climate change," *Social Science Information*, vol. 54, no. 1, pp. 115–145, 2015.

[55] M. A. Shapiro and H. W. Park, "Climate change and youtube: Deliberation potential in post-video discussions," *Environmental Communication*, vol. 12, no. 1, pp. 115–131, 2018.

[56] M. Jurkowitz and A. Mitchell, "About one-fifth of democrats and republicans get political news in a kind of media buble," *Pew Research Center*, 2020.

[57] R. Epstein, R. E. Robertson, D. Lazer, and C. Wilson, "Suppressing the search engine manipulation effect (seme)," *Proceedings of the ACM on Human-Computer Interaction*, vol. 1, no. CSCW, pp. 1–22, 2017.

[58] Y. Shmargad and S. Klar, "Sorting the news: How ranking by popularity polarizes our politics," *Political Communication*, vol. 37, no. 3, pp. 423–446, 2020.

[59] D. Geschke, J. Lorenz, and P. Holtz, "The triple-filter bubble: Using agent-based modelling to test a meta-theoretical framework for the emergence of filter bubbles and echo chambers," *British Journal of Social Psychology*, vol. 58, no. 1, pp. 129–149, 2019.

[60] A. J. Chaney, B. M. Stewart, and B. E. Engelhardt, "How algorithmic confounding in recommendation systems increases homogeneity and decreases utility," in *Proceedings of the 12th ACM Conference on Recommender Systems*, pp. 224–232, 2018.

[61] R. Jiang, S. Chiappa, T. Lattimore, A. György, and P. Kohli, "Degenerate feedback loops in recommender systems," in *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 383–390, 2019.

[62] P. Briñol, D. Rucker, Z. Tormala, and R. Petty, *Individual differences in resistance to persuasion: The role of beliefs and meta-beliefs*, pp. 83–104. Routledge Taylor & Francis Group, Dec. 2003.

[63] M. J. Salganik and D. J. Watts, "Leading the herd astray: An experimental study of self-fulfilling prophecies in an artificial cultural market," *Social Psychology Quarterly*, vol. 71, no. 4, pp. 338–355, 2008.

[64] F. Masrour, T. Wilson, H. Yan, P.-N. Tan, and A. Esfahanian, "Bursting the filter bubble: Fairness-aware network link prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, pp. 841–848, 2020.

[65] R. Bhargava, A. Chung, N. S. Gaikwad, A. Hope, D. Jen, J. Rubinovitz, B. Saldías-Fuentes, and E. Zuckerman, "Gobo: A system for exploring user control of invisible algorithms in social media," in *Conference Companion Publication of the 2019 on Computer Supported Cooperative Work and Social Computing*, pp. 151–155, 2019.

[66] G. Pennycook, J. McPhetres, Y. Zhang, J. G. Lu, and D. G. Rand, "Fighting covid-19 misinformation on social media: Experimental evidence for a scalable accuracy-nudge intervention," *Psychological science*, vol. 31, no. 7, pp. 770–780, 2020.

[67] J. N. Druckman, "Communicating policy-relevant science," *PS: Political Science & Politics*, vol. 48, no. S1, p. 58–69, 2015.

[68] N. Dias, G. Pennycook, and D. G. Rand, "Emphasizing publishers does not effectively reduce susceptibility to misinformation on social media," *Harvard Kennedy School Misinformation Review*, vol. 1, no. 1, 2020.

[69] L. A. Brannon, M. J. Tagler, and A. H. Eagly, "The moderating role of attitude strength in selective exposure to information," *Journal of Experimental Social Psychology*, vol. 43, no. 4, pp. 611–617, 2007.

[70] S. Stier, N. Kirkizh, C. Froio, and R. Schroeder, "Populist attitudes and selective exposure to online news: A cross-country analysis combining web tracking and surveys," *The International Journal of Press/Politics*, vol. 25, no. 3, pp. 426–446, 2020.

[71] G. Eady, J. Nagler, A. Guess, J. Zilinsky, and J. A. Tucker, "How many people live in political bubbles on social media? evidence from linked survey and twitter data," *Sage Open*, vol. 9, no. 1, p. 2158244019832705, 2019.

[72] A. E. Boydstun, D. Card, J. Gross, P. Resnick, and N. A. Smith, "Tracking the Development of Media Frames within and across Policy Issues," 8 2014.

[73] D. Card, A. Boydstun, J. H. Gross, P. Resnik, and N. A. Smith, "The media frames corpus: Annotations of frames across issues," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pp. 438–444, 2015.

[74] N. Naderi and G. Hirst, "Classifying frames at the sentence level in news articles," *Policy*, vol. 9, pp. 4–233, 2017.

[75] K. Johnson, I.-T. Lee, and D. Goldwasser, "Ideological phrase indicators for classification of political discourse framing on twitter," in *Proceedings of the Second Workshop on NLP and Computational Social Science*, pp. 90–99, 2017.

[76] F. Morstatter, L. Wu, U. Yavanoglu, S. R. Corman, and H. Liu, "Identifying framing bias in online news," *ACM Transactions on Social Computing*, vol. 1, no. 2, pp. 1–18, 2018.

[77] D. A. Crow and A. Lawlor, "Media in the policy process: Using framing and narratives to understand policy influences," *Review of Policy Research*, vol. 33, no. 5, pp. 472–491, 2016.

[78] J. Merolla, S. K. Ramakrishnan, and C. Haynes, "" illegal,"" undocumented," or" unauthorized": Equivalency frames, issue frames, and public opinion on immigration," *Perspectives on Politics*, pp. 789–807, 2013.

[79] A. Lawlor and E. Tolley, "Deciding who's legitimate: News media framing of immigrants and refugees," *International Journal of Communication*, vol. 11, p. 25, 2017.

[80] M. K. Merry, "Narrative strategies in the gun policy debate: Exploring proximity and social construction," *Policy Studies Journal*, vol. 46, no. 4, pp. 747–770, 2018.

[81] S. Jiangli, "European & american think tanks and the reality of us-china trade war: An npf application," *Journal of Social and Political Sciences*, vol. 3, no. 2, 2020.

[82] S. Soroka and C. Wlezien, "Tracking the coverage of public policy in mass media," *Policy studies journal*, vol. 47, no. 2, pp. 471–491, 2019.

[83] S. Müller, "Media coverage of campaign promises throughout the electoral cycle," *Political Communication*, pp. 1–23, 2020.

[84] A. P. Kirilenko and S. O. Stepchenkova, "Climate change discourse in mass media: application of computer-assisted content analysis," *Journal of Environmental Studies and Sciences*, vol. 2, no. 2, pp. 178–191, 2012.

[85] K. Coe, P. J. Kuttner, M. Pokharel, D. Park-Ozee, and M. McKasy, "The "discourse of derision" in news coverage of education: A mixed methods analysis of an emerging frame," *American Journal of Education*, vol. 126, no. 3, pp. 423–445, 2020.

[86] P. Pu, L. Chen, and R. Hu, "A user-centric evaluation framework for recommender systems," in *Proceedings of the fifth ACM conference on Recommender systems*, pp. 157–164, 2011.

[87] J. W. Payne, J. W. Payne, J. R. Bettman, and E. J. Johnson, *The adaptive decision maker*. Cambridge university press, 1993.

[88] B. P. Knijnenburg, M. C. Willemsen, and A. Kobsa, "A pragmatic procedure to support the user-centric evaluation of recommender systems," in *Proceedings of the fifth ACM conference on Recommender systems*, pp. 321–324, 2011.

[89] F. Garcin, B. Faltings, O. Donatsch, A. Alazzawi, C. Bruttin, and A. Huber, "Offline and online evaluation of news recommender systems at swissinfo.ch," in *Proceedings of the 8th ACM Conference on Recommender systems*, pp. 169–176, 2014.

[90] P. Gaspar, M. Kompan, J. Simko, and M. Bielikova, "Analysis of user behavior in interfaces with recommended items: An eye-tracking study.," in *IntRS@ RecSys*, pp. 32–36, 2018.

[91] M. F. Ghori, A. Dehpanah, J. Gemmell, H. Q. Saremi, and B. Mobasher, "Does the user have a theory of the recommender? a pilot study.," in *IntRS@ RecSys*, pp. 77–85, 2019.

[92] C. Musto, G. Rossiello, M. de Gemmis, P. Lops, and G. Semeraro, "Combining text summarization and aspect-based sentiment analysis of users' reviews to justify recommendations," in *Proceedings of the 13th ACM conference on recommender systems*, pp. 383–387, 2019.

[93] S. Ganguly, J. Kulshrestha, J. An, and H. Kwak, "Empirical evaluation of three common assumptions in building political media bias datasets," in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 14, pp. 939–943, 2020.

[94] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[95] P. Liu, K. Shivaram, A. Culotta, M. A. Shapiro, and M. Bilgic, "The interaction between political typology and filter bubbles in news recommendation algorithms," in *Proceedings of the Web Conference 2021*, WWW '21, (New York, NY, USA), p. 3791–3801, Association for Computing Machinery, 2021.

[96] C. Doherty, J. Kiley, and B. Johnson, "Political typology reveals deep fissures on the right and left: Conservative republican groups divided on immigration,'openness'," *Pew Research Center*, 2017.

[97] S. Schmit and C. Riquelme, "Human interaction with recommendation systems," in *International Conference on Artificial Intelligence and Statistics*, pp. 862–870, 2018.

[98] E. Ie, C.-w. Hsu, M. Mladenov, V. Jain, S. Narvekar, J. Wang, R. Wu, and C. Boutilier, "Recsim: A configurable simulation platform for recommender systems," 2019.

[99] C. Févotte and J. Idier, "Algorithms for nonnegative matrix factorization with the β-divergence," *Neural computation*, vol. 23, no. 9, pp. 2421–2456, 2011.

[100] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and G. Varoquaux, "API design for machine learning software: experiences from the scikit-learn project," in *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pp. 108–122, 2013.

[101] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in neural information processing systems*, vol. 26, 2013.

[102] S. A. Munson, S. Y. Lee, and P. Resnick, "Encouraging reading of diverse political viewpoints with a browser widget," in *Seventh international aaai conference on weblogs and social media*, 2013.

[103] A. J. Berinsky, M. F. Margolis, and M. W. Sances, "Separating the shirkers from the workers? making sure respondents pay attention on self-administered surveys," *American Journal of Political Science*, vol. 58, no. 3, pp. 739–753, 2014.