Constructed Political Coordinates: Aggregating Over the Opposition in News Recommendation

Eamon Earl
eamon.earl@torontomu.ca
Toronto Metropolitan University
Toronto, Ontario, Canada

ABSTRACT

In the past two decades, open access to information has increased rapidly, empowering education and freedom of thought in the public court of opinion. To help facilitate the consumption of content and mediate the effects of information overload, recommendation systems (RS) have been widely adopted to effectively filter information for users. In doing so, RSs aim to predict what content the user would choose to engage with if they had perfect knowledge of all available content. This is impossible to gauge with complete accuracy, and must be estimated either on the basis of past user interactions or using samples from multiple users over a subset of all content items. True user interests can be overpowered by incorrectly leveraging this interaction data. This phenomena can homogenize recommendations over different content types and in turn induce the homogenization of a user's underlying interests. In the context of news recommendation systems (NRS), the formation of *filter bubbles* can push political partisanship towards the extremes. Considering these behaviors in addition to the political bias embedded within the news articles themselves, it becomes clear that NRSs can dangerously enhance political polarization in the public domain.

CCS CONCEPTS

• **Information Systems → Recommendation Systems; Unbiased Recommendation**.

KEYWORDS

news recommendation, unbiased recommendation, recommendation diversity, user behavior modeling, content-based recommendation, collaborative filtering

1 INTRODUCTION

Recommendation system utility has two main measurements of value: users seeing content that they engage positively with, and the content providers maximizing engagement with their content or platform. While the two are evidently correlated (i.e. a user who is not properly catered to will likely cease to use the platform), the latter provides motivation for recommendation algorithms to shift a user's preferences to make them easier to cater to, resulting in higher expectations of long-term engagement [3].

Within the context of News Recommendation Systems (NRSs) this effect can be particularly harmful to civil discourse, especially in political landscapes that are largely polarized. Research into the relationship between RSs and political typology suggests that users with more extreme political preferences are shown a greater degree of articles which exhibit a singular and identical political bias, and users with divergent views on different topics often have their preferences homogenized via recommendation systems [6]. In this way, NRSs push their users towards *filter bubbles*, wherein users are exposed solely to beliefs that conform with their own.

This loss of political diversity in NRSs is ultimately a loss for civil discourse at large. Within the setting of social networks there is a causal relationship between exposure to conflicting viewpoints on prominent issues and a general increase in political tolerance [10]. In particular, this exposure can further an individual's understanding of opposing viewpoints and empower them to derive richer reasoning for their initial stance.

Thus, Unbiased News Recommendation Systems (UNRSs) should respect a user's diverse political alignments across distinct topics, while also providing the opportunity to interact with opposing viewpoints. In this paper we propose a hybrid Unbiased News Recommendation System which derives a latent understanding of a user's topical preferences in a manner that is agnostic to their partisanship over that topic. The system then recommends articles sourced from politically diverse users over these topics of interest, providing our user with opposing perspectives on personally engaging subjects. This system can then be integrated with any existing NRS, and its diverse recommendations can be queried proportional to the user's desire for diversity.

2 PROBLEM DEFINITION

While traditional NRSs are partly responsible for the propagation of polarization in their user base, the political bias lies in the content itself. A majority of American people believe that news organizations often favour one side of the political spectrum, with many believing that organizations who favour the opposing side are less accurate in reporting the news [5]. Media bias can take many forms: content in favor of the incumbent government, in favor of a particular political party, ideologically liberal or conservative, in favor of industries or companies that advertise heavily in the outlet or that own the outlet, or in favor of audiences that are more valuable to advertisers [7]. It follows that individual articles in their perspective, language and headlines often perpetuate this bias, and leverage it to increase engagement from their desired audience. For the purposes of this paper we will focus on media bias towards ideologically liberal or conservative stances.

An analysis of content-based NRSs showed that they regularly homogenize user partisanship across topics, failing to capture the complexity of a user's political alignments and pushing their preferences towards more extreme ideologies [6]. Content-based *Unbiased*

Author's address: Eamon Earl, eamon.earl@torontomu.ca, Toronto Metropolitan University, Toronto, Ontario, Canada.

News Recommendation Systems must be able to capture a user's heterogeneous partisan leanings over different cross-cutting topics (i.e. liberal on foreign trade and conservative on immigration), which requires learning relationships between content bias and underlying topics. News topics regularly emerge and change over time, and even under static-topic conditions topic classification can be difficult. As such, modern content-based UNRSs have aimed to implicitly model topic preferences. The authors in [8] aimed to leverage linguistic bias by penalizing attention on phrases within articles that reliably predicted the partisan lean, thus focusing more on topic-specific language. Another approach was proposed using disentangled latent embedding spaces to represent a user's topic preferences while encoding minimal bias-predictive information [9]. These approaches shift recommendations to cater to heterogeneous interests, but do not provide insight into opposing viewpoints over these topics: they are geared solely towards recommendation accuracy.

Collaborative filtering (CF) methods have been proposed to manage an optimal trade-off between diversity and accuracy by leveraging recommendations sourced from Furthest-Neighbor (FN) and Nearest-Neighbor graphs using graph convolutions [4]. The degree to which diversity and utility were mixed was controlled via a regularization parameter, and thus could not be dynamically controlled (without retraining) while maintaining the measured performance guarantees. Despite the benefits to diversity, there have been notably less *direct applications* of neighborhood methods for unbiased news recommendation, as they generally leverage popularity bias to make recommendations and cannot directly address the underlying bias in the content itself.

The phenomena of *filter bubbles* is considered an *unintended effect* of NRSs. The mitigation strategies that have been proposed thus far can be considered as *intentional* alterations to recommendation policies which aim to control the often radicalizing effects of recommended content on a user's political preferences. While well-intended and likely beneficial, this can still be seen as unwanted or unknown manipulation of a user's political alignments, and in turn could be argued as non-democratic. As per our knowledge, there is no existing model for unbiased news recommendation which can be leveraged to dynamically manage partisan diversity relative to user desire.

We believe that proposing adequate solutions to the issue of *filter bubble* formation via RSs in democratic spaces requires:

(1) Capturing the user's heterogeneous partisanship over topics.
(2) Recommending articles with diverse partisanship bias over topics of interest.
(3) Allowing the degree to which bias mitigation strategies are applied to be directly controlled by the user.

In the following paper, we propose and analyze such a model, in the hopes of furthering the field of unbiased news recommendation. In Section 3 we cover the high-level behavior of the model, comprised of three distinct modules. In Section 4 we cover the specifics of our implementation in greater detail. In Section 5 we outline our experiments, and analyze the results. Finally, we futher discuss the implications of our system in Section 6.

3 METHODOLOGY

In this Section we introduce the behavior of our system, which can be categorized into three main modules: the Bias Disentangling Module, the User-Embedding Generator, and the Furthest-Neighbor Graph Convolutional Filter.

3.1 Bias Disentangling

The application of disentangling via adversarial autoencoder networks was recently applied to news recommendation, where the framework was leveraged to produce embeddings devoid of both biased information and information with low veracity [9]. These embeddings were then processed via attention layers alongside additional context to form the lower-level of a hierarchical interest model.

We follow this research by similarly applying adversarial autoencoders to disentangle and isolate a representation of underlying interests devoid of political bias. Unlike previous work, we leverage both the polarity-free embeddings as well as the polarized embeddings to produce recommended articles which align with user interest, but are diverse in their partisan lean.

In Figure 1, we show a high-level representation of inference over the bias disentangling module. Firstly, a pre-trained BERT model is applied to generate embeddings over article titles and descriptions, which are combined and compressed via an attention layer [1]. From there, an encoder produces a latent representation which is fed to two separate decoders, our polarity-free decoder and our polarized decoder. During training a polarity classifier is used, from which the former decoder learns to discard polarity-predictive information, while the latter aims to retain and emphasize it. The specifics of this process will be further discussed in Section 4.

3.2 User Embedding Generation

The paper *The Interaction between Political Typology and Filter Bubbles in News Recommendation Algorithms* [6] worked to diagnose how *filter bubbles* form in NRSs, and identified the various kinds of bias which contribute to this phenomenon. This analysis required analyzing how heterogeneous partisanship over topics was homogenized by both CF and content-based algorithms.

To predict article bias, 900k articles were scraped and collected, with their partisan score being annotated using the estimated bias of their source, relative to *www.allsides.com*. This work, like many others, uses a partisan score rating system of { -2, -1, 0, +1, +2 }, from very liberal (-2) to very conservative (+2), to classify bias found in the news source.

The articles were then further annotated over relevant topics. The co-authors identified what they believed to be the most complete and distinct 14 topics in 2020 American politics, and used expert annotators from Amazon Mechanical Turk to identify and label these topics within approximately 2100 articles. A suite of classifiers was then trained to propagate these labels to all articles within the 900k database, predicting for each article whether any of the 14 given topics were covered. This set was then sampled to retain even partisanship over all topics, resulting in a dataset of 40k articles. More details on the dataset will be provided in Section 4.

User preferences were modeled via a 14x5 matrix of utility values u_{ij}, where $u_{ij} \in [0, 1]$ indicates the user's utility for reading

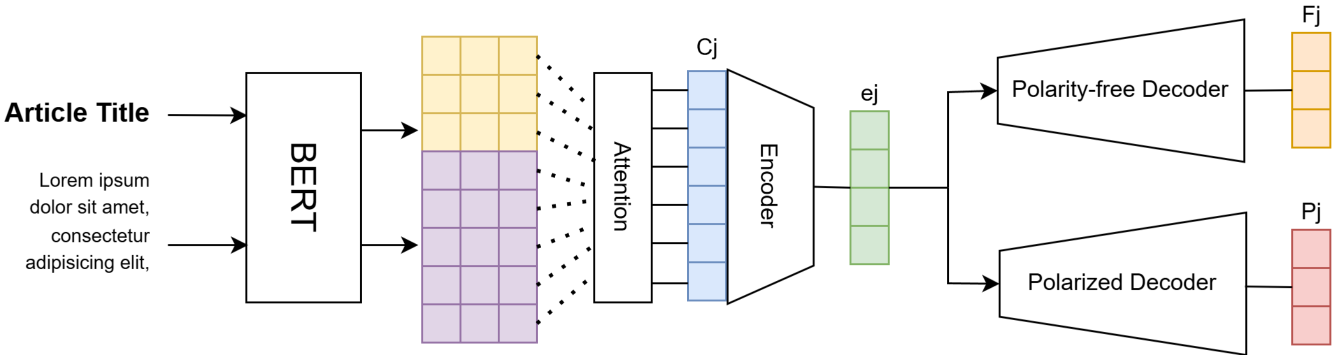


Figure 1: The Bias Disentangling module.

The flattened embedding C_j is used as a target for reconstruction loss relative to the concatenated outputs $[f_j; p_j]$ of the two decoders. The polarity-free decoder is taught to produce embeddings f_j which poorly predict the polarity bias of the article, while the opposite is true for the polarized decoder embeddings p_j.

an article on a topic i with political stance j. For the purposes of accurate-to-world simulation, exemplar user utility matrices were generated with respect to a Pew survey which aimed to identify nuanced political ideologies [2]. Their work defined 9 political classes: bystanders, core conservatives, country first conservatives, devout and diverse, disaffected democrats, market skeptic republicans, new era enterprisers, opportunity democrats, and solid liberals. To generate exemplar utility entry a_{ij} for political class A, they derived the most dominant partisan score $s_A \in \{-2, -1, 0, +1, +2\}$ from survey questions about topic i, and the dominant percent p of responses which aligned with this opinion. The utility a_{ij} was then sampled from a beta distribution $Beta(p, s_A)$, with the resulting value being decayed to other partisan scores as a function of the standard deviation in the survey responses. Article matrices of dimensions 14x5 were similarly modeled with respect to their classified topic and their source partisan score. User-item interaction probability r_{uv} with a given item is then calculated as a dot product between the flattened partisan-topic user and item vectors. In this manner, *exemplar user-choice models* can be constructed for each class, and can be use to generate simulated interaction histories for users in that class.

We apply these user choice simulations to generate user-interest models in the embedding space of our decoders. Given these 9 exemplar user-choice models, we sample our pool of items N times per exemplar, and create a weighted-average exemplar model of the content sampled. Specifically, we compute the polarized embedding:

$$c_{pol}(u_a) = \sum_{i<N} \frac{r_{u_a v_i} * p(v_i)}{r_{u_a v_i}} \quad (1)$$

The polarized embedding $p(v_i)$ of each item is weighted relative to their respective user-item interaction probability r_{uv}. With an adequate number of samples, these 9 user interest models should define the centers of clusters in our user-embedding space.

To calculate the user embedding for a regular, non-exemplar user we first calculate their polarized embedding $c_{pol}(u_T)$. From there, we calculate the pairwise distance between $c_{pol}(u_T)$ and $c_{pol}(u_a)$ for each of the exemplar users $u_a \in L$. These distances are also normalized by the maximum distance in each dimension among our exemplar users to ensure that embeddings remain centered on the exemplar-defined space. Our final embedding for arbitrary user u_T is the 9-dimensional pairwise distance from every exemplar user, w.r.t their polarized embeddings. In this way, we construct a high-dimensional estimation of the abstract political spectrum over a subset of the population. It is important to note that while comparison to our exemplar users occurs in the higher-dimensional embedding space output by our polarized decoder, user-space is represented by only these 9-dimensions. We call this user-space *Constructed Political Coordinates (CPC).*

3.3 Furthest Neighbor Aggregation

Previous work has aimed to leverage Graph Convolutions over Furthest-Neighbor graphs to increase recommendation diversity without notably degrading accuracy [4]. We follow this work to create a Furthest-Neighbor graph according to our *Constructed Political Coordinates*.

Given our M users and their *CPC*, we calculate a correlation matrix B' between our users using the Pearson correlation coefficient. We use this metric as it groups users with similar *patterns* of distance to the exemplar users, prioritizing grouping by relative position as opposed to grouping by the magnitude of distance. As distances were normalized and centered on a space that we assume to be enclosed by our exemplar users, we do not expect to lose significant information in this manner. We then perform a simple conversion to create an inverse correlation matrix (least-correlated matrix) $B_{x,y} = 1 - abs(B'_{x,y})$ for all user pairs $(x, y) \in B'$. We also considered performing pure negation to construct B from B', but found that this method resulted in many users (30-40%) being excluded from the GCF aggregation (they were not chosen as a furthest-neighbor by any other user). We believe this is caused by a majority of left-leaning users targeting only far right-leaning users as their furthest-neighbors and vice-versa, resulting in users with more central ideologies being ignored. We consider this phenomenon to be counterintuitive to the goals of our system and will discuss its relative implications in Section 5.2.

To formalize the graph-convolution problem, we introduce the notation $B \in R^{UxU}$, a symmetric user-correlation matrix over a set of users U. Alternatively, B can be seen as the adjacency matrix of a user-correlation graph $G_U = (U, E_U)$, where U is the user set and the edge set E_U contains an edge $(u, v) \in E_U$ only if $B_{uv} \neq 0$. A rating vector for item i can be seen as a signal on the vertices of G_U. An item-specific graph can then be constructed from this by considering the signals $[x^i]_u$ (equal to rating-matrix entry X_{ui}) for every user-node. Specifically, the correlation graph B is transformed into the item-specific graph B_i as follows: each correlated edge becomes two directed edges, edges from users who have not rated the item are removed, and the strongest k incoming edges to each node are kept. Predicting ratings for users who have not yet rated item i means simply shifting available (normalized) ratings to neighboring users in graph B_i. Specifically, the shifted ratings to immediate neighbors can be written as: $\hat{x}^i = B_i x^i$. The above can be considered as a single-graph convolution which directly mimics the behavior of vanilla Nearest-Neighbor implementations.

We now consider multiple convolutional layers. Let us consider the generic graph adjacency matrix variable S, the generic rating signal x, and the estimated rating signal $\hat{x}$. We then have $\hat{x} = Sx$. Additionally, we can consider the two-step neighbours via the second-order shift $S^2x = S(Sx)$. Generally, neighbours up to k-steps away can be considered via $S^k = S(S^{k-1}x)$. Finally, a set of weight parameters $h = [h_0, ..., h_k]^T$ are introduced to balance the information coming from the different resolutions $Sx, S^2x, ..., S^kx$.

When applying the graph convolutional filter (GCF) transformation to our FN graph in our CPC user-space, we leverage the simplicity of the system to reasonably train a single filter for every individual user. After aggregating over our existing ratings, we select the M items with the highest predicted interaction score and compute their polarity-free embeddings. These embeddings are combined similarly to the polarized embeddings, specified by:

$$c_f(u, V_M) = \sum_{i<M} \frac{\hat{r}_{uv_i} * f(v_i)}{\hat{r}_{uv_i}}, v_i \in V_M \tag{2}$$

The weight parameters h are trained to propagate rating signals such that the top-M polarity-free embedding $c_f(u, V_M)$ is as close as possible to $c_f(u, V_T)$, where V_T is the historical set of prior items and interaction scores for our user u. In essence, we train our weights h to aggregate ratings from users with alternate political views such that the topics of articles chosen are similar to the underlying partisan-agnostic topic preferences of the user.

4 MODEL DETAILS

In this section we define the specifics of our implementation, data, and training process.

4.1 Disentangling Model Structure

For generating our initial embeddings we used the pretrained BERT base model, which has an embedding dimension of 768 and 110 million parameters [1]. Text and title embeddings (256 and 64 tokens respectively) were then concatenated and fed into an attention layer with 8 attention heads and a dropout of 0.1. The attention layer casts the concatenated embeddings into a single vector which is then passed through a sigmoid activation function, producing embedding c_j of dimension 256. As per [9], the encoder is a dense network with a single hidden layer employing both residual connections and LeakyReLU activation functions. The encoder takes in vector c_j as input and produces a latent representation of dimension 128. The two decoders are similarly constructed, though their outputs are vectors of dimension 128, half the dimensionality of our input vector c_j, and are concatenated for calculating the reconstruction loss.

For training purposes, a *polarity classifier* is instantiated, which is simultaneously trained to predict the polarity of an article given the outputs of either decoder. This model is also a dense network with a single hidden layer and a softmax activation function, which outputs predictions over the polarity classes. In accordance with other work, we simplify the classification task by collapsing partisan classes into $\{-1, 0, 1\}$.

The training objective for the bias-disentangling module is formalized by:

$$Loss = L_{class} + L_{conf} + L_r \tag{3}$$

L_{class} is the cross-entropy classification error of our polarity classifier over the polarized embedding:

$$L_{class} = -y^p log(\hat{y}^p) \tag{4}$$

In Equation 4, y^p is the true polarity label and $\hat{y}^p$ is the predicted polarity label given the polarized embedding p. We also force the polarity-free decoder to produce embeddings devoid of polarity-predictive information through our confusion loss:

$$L_{conf} = -\frac{1}{y^p log(\hat{y}^f)} \tag{5}$$

In Equation 5, $\hat{y}^f$ is the predicted polarity label given our polarity-free embedding f. Finally, we force both of our decoded embeddings towards the input to our autoencoders, c_j, via reconstruction loss:

$$L_r = \frac{1}{2}(c_j - [f_j; p_j])^2 \tag{6}$$

4.2 Data Preparation

All training and evaluation was done using the dataset of 40k news articles from 2020, classified on the basis of expertly-labeled data into 14 topics, and labeled with partisan bias relative to their respective media outlets [6]. This data was filtered to have a near perfect distribution of partisan bias, but had uneven distributions over topics. This allows training and evaluation to be more reflective of real-world scenarios, wherein certain topics have increased prevalence at given points in time.

We used an 8:1:1 split for training, testing, and validation data, stratified primarily over partisan distribution and secondarily over topics. While a given article can have multiple topics, only one was selected at random for this stratification process.

For the sake of efficiency in testing our Furthest Neighbor Aggregation module, we took only the first 1000 items from the training set to form our item pool. We take this as a reasonable simplification, seeing that there is only ever a small subset of current and relevant candidate news items available for recommendation. The statistics of this data are provided in Table 1.

Additionally, we simulate 1000 users in the manner discussed in Section 3. These users will populate the user-embedding space and be aggregated over in the Furthest-Neighbor graph convolutions to form predicted user utility scores. True to life, this set is not an even distribution, and our most dominant political classes are the core conservatives and the solid liberals (15.3% and 19.3% respectively).

4.3 Model Training

4.3.1 Pre-training. During early training attempts, the *Bias Disentangling Module* showed negligible learning across epochs, presumably stuck in a local minima across the multi-objective training function. This resulted in minimal diversity among the embeddings $c_{pol}(u_a)$ for exemplar users $u_a \in L$. To address this, we introduced a pre-training phase for the multi-head attention layer and the encoder. Specifically, we trained our encoder for 3 epochs over our first 1000 points in the training set, optimizing to increase the orthogonality of our embeddings c across different partisan labels y^p within each batch of size 50.

$$L_o = \sum_{c_i} \sum_{c_j \in b_i} (\dot{c}_i \, \dot{c}_j^T)^2, \qquad b_i = \{c_q \mid y_q^p \neq y_i^p\} \tag{7}$$

Where $\dot{c_i}$ represents the unit vector of our embedding c_i. In this manner, we ensure our target-embedding c for our decoders is initially capturing polarity-predictive information, thus allowing disentangling behavior to be learned from the beginning of training. For the remainder of the training process the multi-head attention layer is frozen, to dissuade the system from collapsing into this local minima once more. For pre-training we applied the Adam optimizer with a learning rate of 0.001.

4.3.2 Disentangling. The architecture of our disentangling module was covered in Section 4.1. The model was implemented in PyTorch. To train this model we ran 32 epochs, each over a different 1000 samples from our training set, with batches of size 50. This was done as a means to limit the memory complexity of training on the full set for every epoch. We used the Adam optimizer with a learning rate of 0.001, and a scheduler with linear warm-up for the first 6 epochs and exponential degradation for the remaining 26 epochs with a decay rate of 0.95. To ensure training was not beginning in a local minima, we randomly initialized the weights of our decoder layers relative to a uniform distribution between -2 and 2.

Training was done on an NVIDIA GeForce GTX 1650, with 4GB of VRAM and 12GB total memory.

The validation set was used to confirm that there was no significant overfitting, and none of our hyperparameters were tuned during, in-between, or after training.

4.4 Filter Training

After the disentangling model was trained, we generated our exemplar users and encoded each of our 1000 simulated users into their 9-dimensional *Constructed Political Coordinates*, as detailed in Section 3.2.

The user-item matrix was generated over 1000 simulated users and the first 1000 items of the test set. The statistics of these user and item subsets are described in Table 1. A user-item matrix was simulated via the user-choice model previously discussed, with 10 interactions being logged per user. As such, our user-item matrix is 1% filled, with even distributions of ratings among users. This can be seen as only permanently holding the last 10 ratings of every user. The ratings are modeled as the interaction score generated by the user-choice model, $r_{ui} \in [0, 1]$, and are not only highly-rated interactions. For generating the FN correlation matrix, 30 furthest neighbors were retained for each user, allowing the model ample choice of users over which to focus rating aggregation.

Initially, we attempted to train a single graph convolutional filter over our furthest-neighbor graph that would generalize over all users. As expected, the model performed poorly in this use-case, regularly re-fitting towards the user immediately being evaluated, thus losing much of the information gathered in previous learning steps. The resulting loss graph can be seen in our Appendix. Note how mass reductions to the loss happen at an arbitrary epoch, but *patterns* of loss remain the same per-user. This was somewhat expected behavior: multiple GCFs are generally used in combination to implement Graph-Neural-Networks (GNNs), which would exhibit better behavior in generalizing across users. The benefit of singular GCFs for aggregation are their lightweight nature; it is not inconceivable to train a singular filter for each user.

For each of our users, we train a 5-step filter over 40 epochs using Stochastic Gradient Decsent with a learning rate of 0.05, warming up for the first 8 epochs followed by 32 epochs with an exponential decay of 0.98. It is worth noting that many users were continuing to see reductions to the loss function up to epoch 40, though we stopped at 40 epochs for the sake of more efficient computation over the 1000 users and 1000 items.

Each epoch required a recomputation of the predicted ratings through the GCF with respect to the adjusted $[h_1, ..., h_5]$ weights. Loss was calculated as the Mean-Squared Error between the averaged polarity-free embedding $c_f(u, S)$ over the top-10 recommended articles in slate S and the embedding of our previous interaction data $c_f(u, V_T)$.

5 EXPERIMENT

The behavior we are aiming to test is twofold: firstly, we wish to recommend articles whose topics match the topics over which a user has shown previous interaction with, and secondly we aim to recommend articles with diverse partisan bias over these *topics-of-interest* relative to the partisan bias of the previously-interacted articles. If we consider our articles in terms of their 2-dimensional partisan score, topic representation (s, T), we consider the problem of targeting accuracy over feature T, while targeting diversity over feature s. As disentangled representations are relatively new and diversity is usually not the primary goal of recommendation stakeholders, there are no true baselines to compare against for this method, especially among work targeting news recommendation.

It is worth noting that while partisan scores s were collapsed into only three classes for training the disentangling module, we refer to the original set of {-2, -1, 0, 1 , 2} (from far-left to far-right) for the purposes of evaluation.

Topics	**% Articles**		
	Training	Test	GCF Items
abortion	2.8	2.8	2.4
environment	3.5	3.5	3.5
guns	3.7	3.9	4.2
health care	10.9	10.9	10.7
immigration	9.8	9.8	**10.8**
LGBTQ	2.5	2.5	1.9
racism	8.2	8.2	8.5
taxes	5.7	5.7	4.9
technology	2.7	2.7	2.9
trade	4.8	4.7	4.5
trump impeachment	11.8	11.8	11.0
us military	15.3	15.3	15.9
us 2020 election	14.5	14.4	**15.6**
welfare	3.8	3.8	3.2

Political Typology	**% Users**
bystanders	4.9
core conserv	15.3
country first conserv	6.3
devout and diverse	6.9
disaffected democrats	10.8
market skeptic repub	11.6
new era enterprisers	11.6
oppty democrats	13.3
solid liberals	19.3

Table 1: Dataset Statistics

On the left we have the distribution of article topics over the training and test sets as well as the test subset used for FN aggregation. Bolded topics are furthest from the distribution in the total dataset. On the right we show the distribution of political classes within the 1000 users over which our aggregation is performed.

Political Typology	**Avg % TC**	**Std % TC**	**Avg % Div**	**Std % Div**
bystanders	0.65	0.17	0.78	0.13
core conserv	0.70	0.18	0.78	0.16
country first conserv	0.67	0.19	0.79	0.14
devout and diverse	0.70	0.16	0.70	0.16
disaffected democrats	**0.74**	0.16	**0.82**	0.16
market skeptic repub	0.70	0.19	0.81	0.14
new era enterprisers	0.70	0.17	0.79	0.15
oppty democrats	**0.71**	0.16	**0.83**	0.13
solid liberals	**0.73**	0.18	**0.83**	0.15

Table 2: Experiment Results

Percent Topic Coverage and Diversity over topics-of-interest, along with their standard deviation.

5.1 Metrics

5.1.1 Topic Coverage. Our first metric to assess the quality of our system is *topic coverage*. After training the h weights as outlined in Section 4.4, we apply them in a GCF and sample the top 10 predicted interaction scores for each user respectively.

$$\% \ Topic \ Coverage = \frac{\sum_{k \in S} \delta(T_k \cap T_u)}{|S|} \tag{8}$$

Equation 8 shows the applied metric for topic coverage, where S is the recommended slate, T_k is the set of topics covered in each recommended article k, and T_u is the topics-of-interest set for user u. The δ function output 1 when the internal intersection is non-empty, and 0 otherwise. Articles can have more than a single topic covered, though most have only one. We optimistically consider a positive result when any topic of a recommended article matches any topic-of-interest for our user.

5.1.2 Diversity. Our second metric for evaluation is the diversity of partisan bias in recommendations over topics-of-interest. We focus our diversity metric specifically on the recommendations which accurately target topics-of-interest for two reasons. Firstly, without previous interaction over the topic of a recommended article, we would not have any basis for assessing whether or not the recommended article was politically diverse from what the user would generally read regarding that topic. Secondly, while we want to promote diverse viewpoints, we must also respect a user's underlying, unbiased interest, else we may adversely affect user satisfaction.

$$\% \ Diversity = \frac{\sum_{k \in S} max_{t_k \in T_c} \left[\delta(t_k \cap T_u) \cdot (1 - \bar{U}_{t_k, s_k})\right]}{|T_c|} \tag{9}$$

In Equation 9, we evaluate diversity of items $k \in S$ over topics-of-interest according to the topic $t_k \in T_C$ with maximum partisan diversity. Partisan diversity scores for an article k recommended to user u are calculated from the 14x5 topic-normalized user-interaction matrix $\bar{U}$. We reward recommendations relative to how likely a user was to click them based on their previous interaction scores. If the recommended article had a partisan bias over which the user had not previously interacted with, a maximum score of 1 would be rewarded for that article. These scores are averaged over the set of articles which included topics-of-interest, T_c. For articles with multiple relevant topics, we take the maximum diversity score from among them, as we consider strong diversity in a single underlying article topic to be sufficient.

5.2 Evaluation and Analysis

The results of our experiment per political type are summarized in Table 2. Relative to both statistics we observe a fairly even distribution of performance.

Bystanders and country-first conservatives see the worst performance with respect to topic coverage, possibly implying that their political types on average have more interest in certain topics which are less catered-to by the system. Performance over individual topics can be reflective of our disentangling robustness, wherein certain more abstract topics may be less-effectively understood in the latent space. Alternatively, the imbalanced distribution of items in the set could cause this as well.

The devout and diverse class had notably the worst recommended diversity over topics-of-interest, likely due to the specific heterogeneity of their interests. This class is generally right-leaning, but left-leaning with respect to issues of social welfare and discrimination. It is possible that the average placement of devout and diverse users within the *CPC* space made this distinction relatively more difficult to capture than other heterogeneous interest classes. Alternatively, performance could also degrade if a class is less extreme in their partisanship over certain topics: if initial user interest is already diversified over a topic, the recommended article is less likely to receive a perfect diversity score.

The alternative can be said for the more left-leaning classes. Solid liberals, opportunity democrats and disaffected democrats comprise the top 3 classes for both metrics. These users make up a majority of the users over which aggregation was performed. Intuitively one would think that users within the dominant-bias of a political space would have more trouble in being paired with politically diverse Furthest-Neighbors than smaller political types with many alternatives to choose from. We see that this is not the case. One possible reason for this is that left-leaning users may on average be more left-leaning than right-leaning users are right-leaning. Considering our manner of converting the correlation matrix into an inverse-correlation matrix $B_{x,y} = 1 - abs(B'_{x,y})$, we consider that for users with more extreme political alignment, this should target users within a more central political typology as Furthest-Neighbors. We consider that left-leaning users may generally be further politically from the central political types than the right-leaning types. This would make catering diverse viewpoints to left-leaning users from articles sourced from more central-users an easier task.

Ultimately, other than devout and diverse users, the average % diversity scores were very similar across classes, and relatively high. The implication is that our system is managing to cater diverse recommendations across topics of interest relatively well to most users, regardless of their pre-existing interests.

Some example heatmaps comparing pre-existing interests of random users and their associated recommendations from the system are shown in the Appendix. Our Disaffected Democrat had topic interests which were abundant in the item set, and the system consequently had many options to find diverse recommendations across these topics and achieved high coverage and diversity. This behavior is similar for our Devout and Diverse user, though much of their interest was not over popular topics. Despite articles on the topic of abortion and the environment being two of the lowest represented topics, the interest was captured and catered to. Note that the right-leaning article on the topic of racism had a relatively low-response from the original user, and the system recommended two articles on this topic with a left-leaning bias. While this is well diversified, we do not know if the low original interaction was due to the bias of this article, or simply that the user is uninterested in the topic. In the future, filtering out low-responses from the user-item matrix may be preferable to avoid this manner of recommendation.

For the Market Skeptic Republican, we have perfect diversity across topics-of-interest, but only 50% topic coverage. This user has a particular interest in health care, and while our diverse recommendation on the topic of health-care was well done, the recommendations were very scattered over many topics. This suggests that our system may have difficulty in catering to users with very specific topical interests.

Finally, our New-Era Enterpriser represents an example of generally poor performance, wherein we struggled to capture interest over certain topics and also failed to fully diversify. This user's interest represent a fairly central right-leaning partisanship. Not only does this make them generally harder to cater to by sampling from one side or the other, but it also places them quite centrally in our *CPC* and thus makes them liable to identify Furthest Neighbors on both the left and the right. Combined with interests that are less represented, this likely caused the lower-fidelity performance. It is worth noting that on an individual basis, more focused training could result in better performance.

6 DISCUSSION

Our modern political landscape can be difficult to navigate. An increase of virtual spaces which cater to specific points of view has created environments where confirmation bias is abundant and expected. Studies have shown that an increase in diverse viewpoints benefits individuals by providing a stronger understanding of opposing partisanship and more robust reasoning for one's own. Recommendation systems, particularly those that deal with heavily politicized content are a large part of this ecosystem. We have seen how susceptible news recommendation is to this phenomenon, resulting in the formation of filter bubbles. Research has also shown that recommendation systems have the capacity to influence the underlying preferences of a user [3]. It becomes clear in light of this that unbiased news recommendation is necessary. Despite this, unbiased news recommendation is an abstract goal, as news media is influenced by multiple sources of bias, and user preferences must be respected while access to diverse viewpoints is maintained. For this reason, we have developed a system that can isolate the topics that a user values and provide opposing viewpoints over said topics with strong average performance.

Being provided only noisy partisan labels based off of the perceived bias of the news source and no information about underlying topics, this system is able to achieve what we consider a strong step towards the abstract goal of democratic news recommendation. Notably, the lightweight schematics of the model allow the system to be mixed-in dynamically with another policy focused on user utility. This allows users to opt-in to diversity as much as they are willing. In this manner, the naturally developing partisanship of a user will not be ignored as it might in other systems which aim to implicitly de-bias user models while simultaneously catering to user utility.

It is worth noting that the only section in the proposed model which benefits from some sense of feature encoding is the choice of our exemplar users. These were created from data which was sourced and annotated from the year 2020, and while some topics covered are relevant, clearly some (i.e. trump impeachment), are less applicable today. This could be amended through other manners of simulating exemplars or clustering techniques, or by simply calculating correlation directly over the decoder output, and omitting the pairwise-distance compression step. We consider addressing this embedding in a more general manner as future work.

Another manner in which this system could be made more general would be through the application of a single generalized Graph-Neural-Network, as opposed to individual GCFs per-user. We consider this a possible avenue for future work as well.

Regarding the results, we do not believe that the system focusing on abundant topics within the item-set is strictly negative, as this matches the nature and the temporal-dependency of news recommendation: whatever is being reported most heavily at a given time should be recommended in a proportionately greater measure. We think it necessary to outline that topic-coverage in this sense should not be considered synonymous to recommendation accuracy.

While the method for simulating users derived in [6] is robust, it can never be fully representative of real-world dynamics. We hope to extend this approach to a modern dataset in the future, and to gather direct user-feedback on the system.

7 CONCLUSION

Encouraging political thought and interaction with opposing viewpoints is necessary to the heart of democratic mobilization in modern society. While the new age of information has vastly increased our ability to interact and exchange opinions, the nature of recommendation algorithms has contributed to political discourse being relegated to filter bubbles, where individuals have their pre-existing opinions re-affirmed, and hardly challenged. We believe it is the responsibility of organizations distributing political information and those holding them accountable to provide users with diverse viewpoints while respecting their underlying interests and motivations. We have provided a lightweight, hybrid recommendation system that can achieve this behavior, and we hope that it can lay the foundation for further research into this avenue of democratic news recommendation.

REFERENCES

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Jill Burstein, Christy Doran, and Thamar Solorio (Eds.). Association for Computational Linguistics, Minneapolis, Minnesota, 4171–4186. https://doi.org/10.18653/v1/N19-1423

[2] Carroll Doherty, Jocelyn Kiley, and Bridget Johnson. 2017. *Political Typology Reveals Deep Fissures on the Right and Left.* Technical Report. Pew Research Centre.

[3] Carroll et al. 2021. Estimating and Penalizing Induced Preference Shifts in Recommender Systems. In *RecSys '21*. ACM. https://doi.org/10.1145/3460231.3478849

[4] Isufi et al. 2020. Accuracy-diversity trade-off in recommender systems via graph convolutions. In *Information Processing and Management*. Elsevier. https://doi.org/10.1016/j.ipm.2020.102459

[5] Kohut et al. 2011. *Press Widely Criticized, but Trusted More than Other Information Sources.* Technical Report. Pew Research Centre.

[6] Liu et al. 2021. The Interaction between Political Typology and Filter Bubbles in News Recommendation Algorithms. In *Proceedings of the Web Conference 2021*. ACM. https://doi.org/10.1145/3442381.3450113

[7] Puglisi et al. 2015. Empirical Studies of Media Bias. In *Handbook of Media Economics*. Elsevier. https://doi.org/10.1016/B978-0-444-63685-0.00015-2

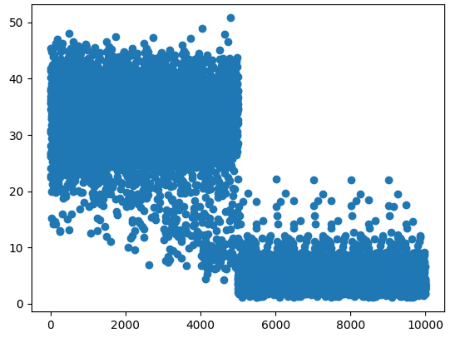
[8] Shivaram et al. 2022. Reducing Cross-Topic Political Homogenization in Content-Based News Recommendation. In *RecSys '22*. ACM. https://doi.org/10.1145/3523227.3546782

[9] Wang et al. 2024. A Hierarchical and Disentangling Interest Learning Framework for Unbiased and True News Recommendation. In *Proceedings of the 30th Conference on Knowledge Discovery and Data Mining*. ACM. https://doi.org/10.1145/3637528.3671944

[10] Diana C. Mutz. 2002. Cross-Cutting Social Networks: Testing Democratic Theory in Practice. In *The American Political Science Review*. American Political Science Association.

8 APPENDIX

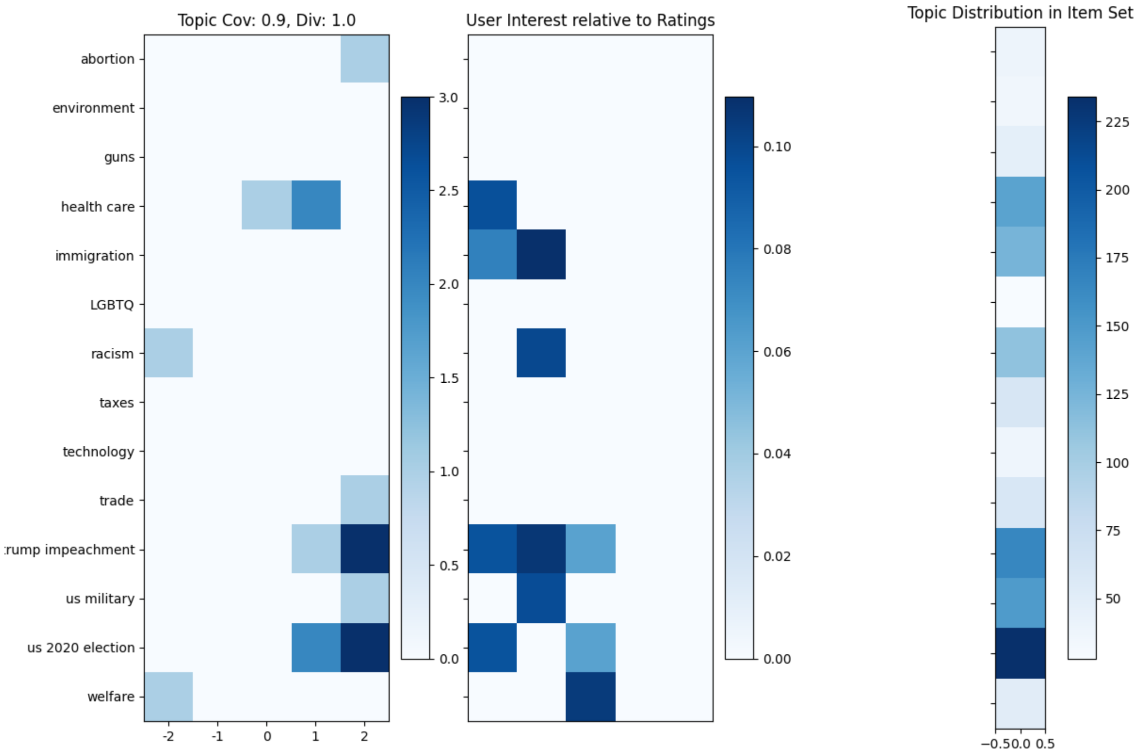
8.1 Generalized GCF



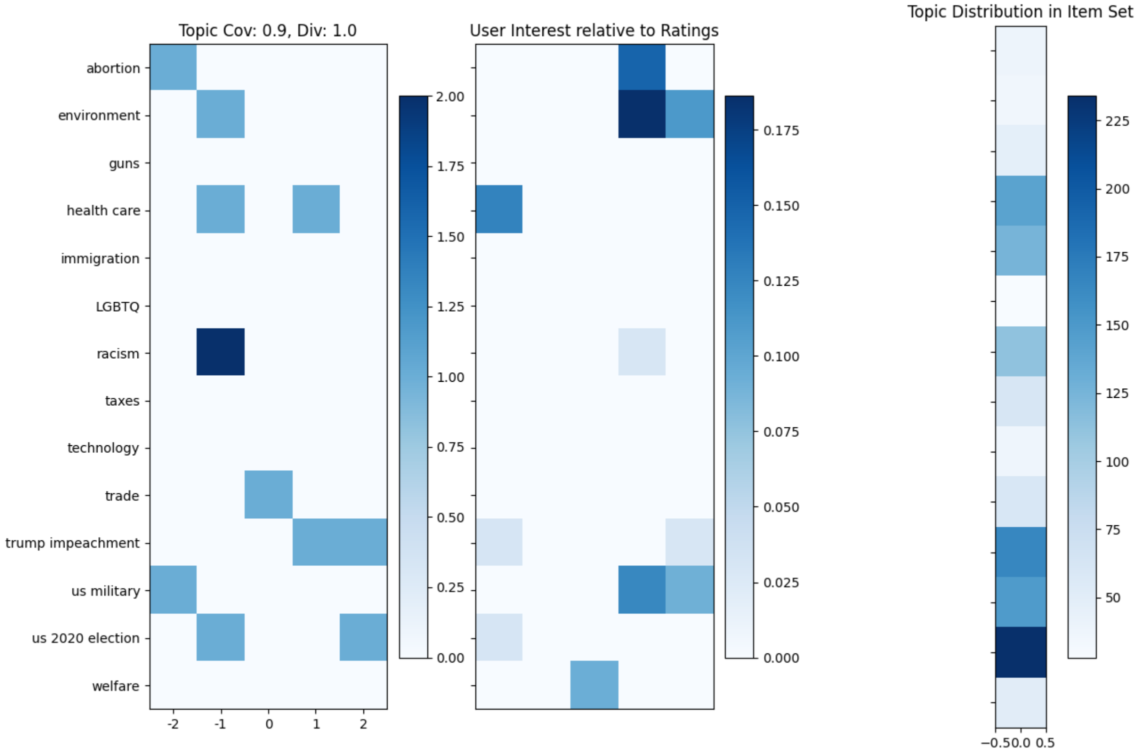
A 7-step GCF learning to capture user interest devoid of partisan-bias over 1000 users simultaneously. While performance was not ideal, and training was highly unstable, it is interesting how around 5000 epochs it was able to greatly reduce average loss. A GNN would be better equipped for this use-case.

8.2 User Examples

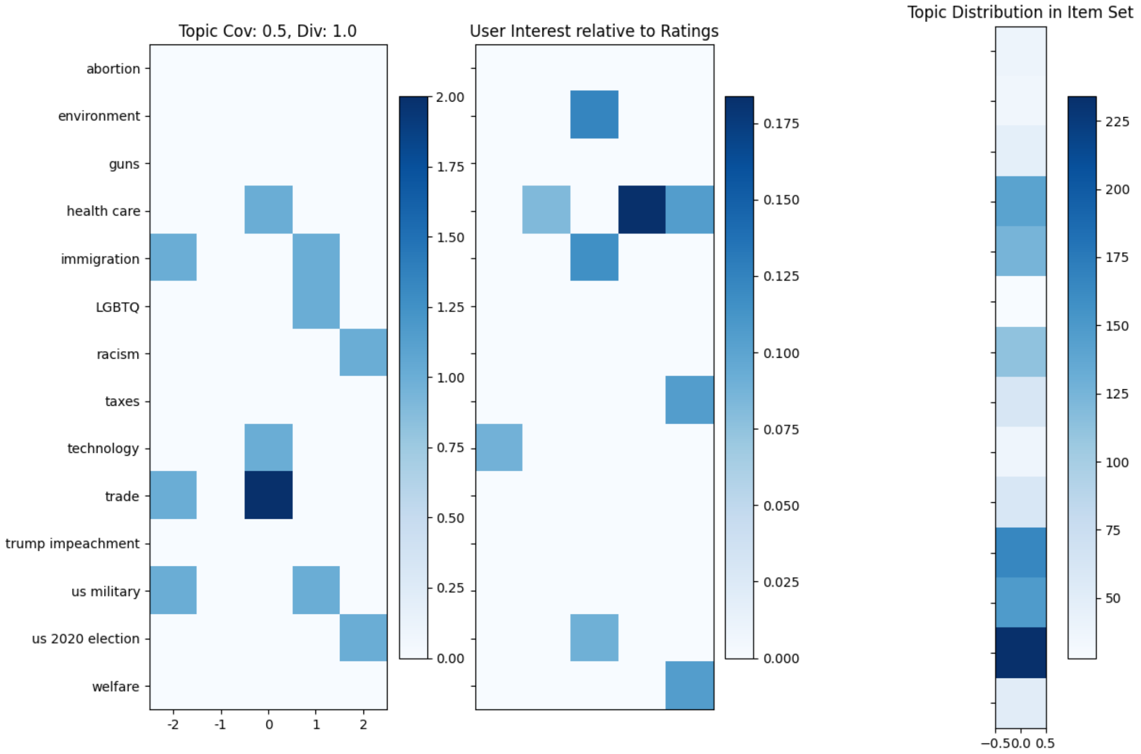
8.2.1 *Disaffected Democrat.*



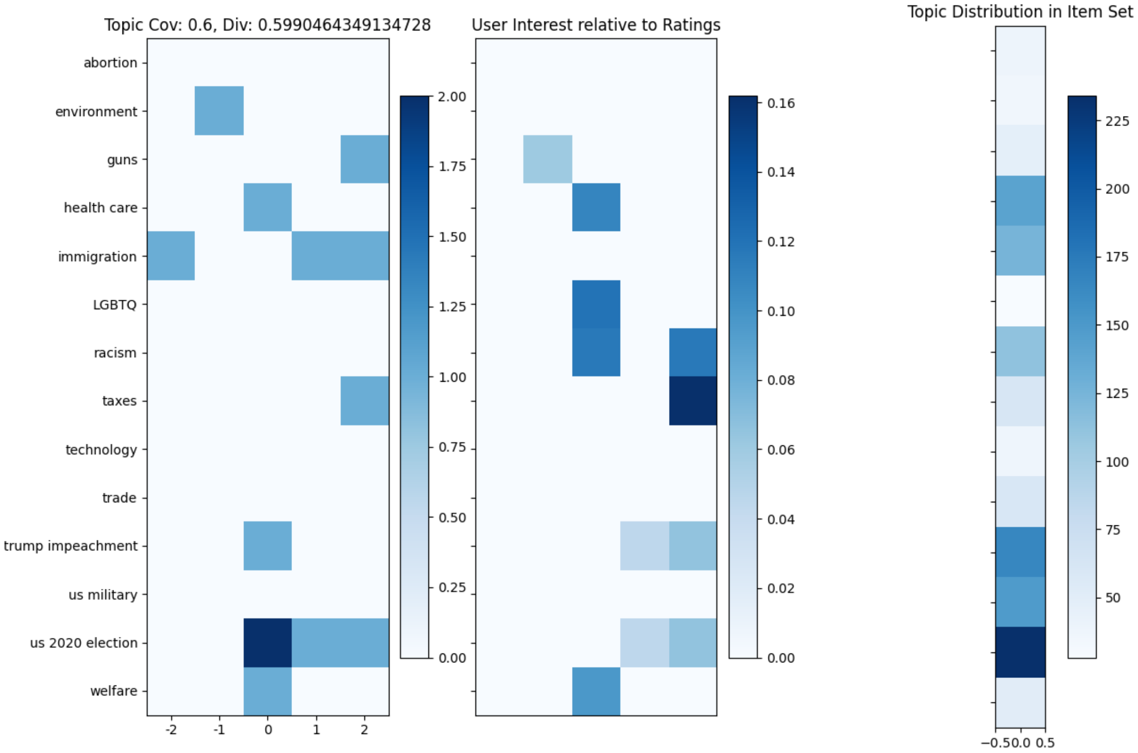
8.2.2 *Devout and Diverse.*



8.2.3 *Market Skeptic Republican.*



8.2.4 *New Era Enterpriser.*


Topic Cov: 0.6, Div: 0.5990464349134728
abortion
environment
guns
health care
immigration
LGBTQ
racism
taxes
technology
trade
trump impeachment
us military
us 2020 election
welfare
-2
-1
0
1
2
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
User Interest relative to Ratings
0.16
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
Topic Distribution in Item Set
225
200
175
150
125
100
75
50
−0.50.0 0.5