Enabling Cross-Platform Comparison of Online Communities Using Content and Opinion Similarity

Prasanna Lakkur Subramanyam Jeng-Yu Chou Kevin K. Nam† Brian N. Levine
University of Massachusetts Amherst
†MIT Lincoln Laboratory
{psubramanyam,jchou,bnl}@umass.edu
kevin.nam@ll.mit.edu

Abstract

With the continuous growth of online communities, understanding their similarities and dissimilarities is more crucial than ever for enhancing digital interactions, maintaining healthy interactions, and improving content recommendation and moderation systems. In this work, we present two novel techniques: *Binary Opinion Transfer Similarity* (BOTS) for finding similarity between online communities based on their opinion, and *Emb-PSR* for finding similarity in the content they post. To facilitate finding the similarity based on opinion, we model the opinions on online communities using upvotes and downvotes as an indicator for community approval. Our results demonstrate that BOTS and Emb-PSR outperform existing techniques at their individual tasks while also being flexible enough to allow for cross-platform comparison of online communities. We demonstrate this novel cross-platform capability by comparing GAB with various subreddits.

1 Introduction

Online communities have become important venues for public discourse and opinion. Platforms like Reddit, with its diverse range of subreddits, offer a unique window into the collective sentiments and interests of different groups. Understanding these sentiments and the similarities between different communities can provide valuable insights for users, researchers, marketers, and social scientists.

While several prior works have explored the task of measuring the similarity between communities (Martin, 2017; Kumar et al., 2018; TeBlunthuis and Hill, 2022; Mok et al., 2023), existing works either measure the community similarity in only a single dimension or use manually annotated seed communities to guide similarity measures across different dimensions. In contrast, our approach divides the community similarity task into two distinct components — content similarity and opinion similarity — and does not require manually identified seed communities. Content similarity measures the similarity in the types of topics being discussed in the two communities regardless of how a community reacts towards a topic. For example, all politics related subreddits will be close in this metric regardless of their political leaning. In contrast, opinion similarity measures the similarity in the way communities react to content posted on the community. For example, left-leaning politics related subreddits will be close to each other but will be further away from right-leaning subreddits.

We obtain content similarity between communities A and B by calculating the ratio of posts in a community B that are similar to posts in A. We use SentenceBERT to obtain sentence embeddings to facilitate the comparison. To enable the calculation of opinion similarity, we model the opinions of a community by training a LLAMA2 (Touvron et al., 2023) model at the task of predicting whether a comment will get heavily upvoted or downvoted. If a model trained on data from Subreddit A can accurately predict the upvote/downvote patterns in Subreddit B, we infer that the two subreddits share similar opinions. We evaluate content similarity and opinion similarity separately and demonstrate that our techniques outperform existing techniques by a significant margin in both the tasks. Our meth-

This material is based upon work supported by the Department of the Air Force under Air Force Contract No. FA8702-15-D-0001. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Department of the Air Force.

Findings of the Association for Computational Linguistics: EACL 2024, pages 10017–10028
November 12-16, 2024

ods, while being more performant also have the unique advantage of allowing us to compare online communities from different social media platforms. We demonstrate this benefit by comparing discussions from the GAB online community with various subreddits.

More concretely, our contributions are as follows.

1. **Opinion modeling (i.e., score classification) on various subreddits.** We leverage upvotes and downvotes as indicators of community approval and disapproval. Our approach models community opinions by training LLAMA2 to predict whether a comment will receive significant upvotes or downvotes. Our results show that parameter-efficient fine-tuning of LLAMA2 with LoRA (Hu et al., 2022) substantially outperforms both the zero-shot and five-shot LLAMA2 classifiers. Several of these fine-tuned per-subreddit models achieve test F_1 scores high enough to effectively predict community reactions to comments.

2. **Methods to calculate the similarity between communities based on opinion (BOTS) and content (Emb-PSR).** We break down the task of calculating similarity between online communities into two axes: opinion similarity and content similarity. To evaluate BOTS, we calculate the similarity between political subreddits. We evaluate Emb-PSR by grouping around 183 subreddits across 16 categories. Our evaluation demonstrates that we outperform existing techniques on both of the sub tasks.

3. **Cross platform community similarity.** BOTS and Emb-PSR do not use platform specific features such as network and graph features. Hence, unlike prior work we can compare communities from different social media platforms using BOTS and Emb-PSR. We demonstrate this benefit by comparing GAB with various subreddits.

2 Related Work

Numerous prior studies have explored the task of forecasting content popularity on social media platforms. The Social Media Prediction (SMP) Challenge (Wu et al., 2019) predicts image popularity on Flickr, leveraging both text and visuals. Wang et al. (2020) predict the popularity of posts by extracting features from textual and visual components, incorporating numerical attributes and follower and following counts, and leveraging a CatBoost model. Other related works make use of neural network architectures (Abousaleh et al., 2021; Lin et al., 2022; Xu et al., 2020). Notably, visual features are commonly derived through convolutional neural networks, while text features frequently employ pretrained embeddings, such as Word2Vec (Mikolov et al., 2013).

Several prior works have explored the task of representing a community. Martin (2017) creates embedding representations for communities by applying the GloVe algorithm (Pennington et al., 2014) on a user-subreddit co-occurrence matrix. They apply t-SNE (van der Maaten and Hinton, 2008) on the resulting embeddings to find clusters of similar subreddits. In a similar vein, Waller and Anderson (2019) create community embeddings by using Word2Vec on data where the communities are treated as words and users are treated as contexts. Waller and Anderson (2021) use the same word vectors and apply agglomerative clustering to group subreddits; they use manually identified seeds to order the communities along various "social dimensions" such as age, partisanship, and affluence. Mok et al. (2023) use the vectors developed by Waller and Anderson (2019) to rank the subreddits on a political scale based on their similarity to `r/Conservative` and `r/progressive` subreddits. Partridge et al. (2024) use an approach similar to Waller and Anderson (2021) to create embeddings for each subreddit. The distance between subreddits is measured by the cosine distance between the L2-norm of any two subreddit embeddings. Ma et al. (2023) use a graph neural network to model communities and perform tasks such as violation detection, sentiment and analysis. In our work, we model the opinions of a community by training a LLAMA2 model to predict whether a comment will get heavily upvoted or downvoted.

Chandrasekharan et al. (2018) cluster subreddits based on whether a model trained on moderation data for each make similar predictions. Kumar et al. (2018) model Reddit conflicts by clustering communities by applying Word2Vec or GloVe to co-occurrences of actively posting users among subreddits. Datta and Adar (2019) identify inter-community conflicts by identifying users who receive many *upvotes* in one subreddit and receive many *downvotes* in another subreddit. TeBlunthuis and Hill (2022) cluster subreddits based on user

overlap, which is high if two subreddits share a lot of users. Phadke et al. (2021) assign subreddits along a "conspiracy scale" by applying PCA on the user-subreddit co-occurrence matrix and manually identifying the axis that ranks known conspiracy and science subreddits on opposite ends. Samory and Mitra (2018), while studying the rise of conspiracy theories in the aftermath of major events, use positive pointwise mutual information and Community2Vec (Martin, 2017) vectors to identify communities similar to `r/science` and `r/conspiracy`. In contrast to past work, our approach to measuring community similarity does not require manually identified seed communities. Once an opinion model is trained on a subreddit, such as `r/science`, we can subsequently determine its similarity to other online communities without any further retraining.

3 Corpus Overview

We use data from two popular online social media platforms, Reddit and GAB, in our experiments. We obtained data from both platforms through Pushshift (Baumgartner et al., 2020), including all posts and comments made on Reddit each month. Our models are trained on Reddit data between January 2018 and April 2023. Our focus is on assessing the approval or disapproval comments receive within specific subreddits, given the context of individual posts.

For each post, we use its title, ID, and selftext (i.e., textual content). For each comment, we extract the comment's text and its score, calculated as the number of *upvotes* minus the number of *downvotes*. Upvotes and downvotes signify community approval or disapproval, respectively, with each account limited to one vote per post and comment. While multiple accounts are permissible (Reddit, 2023), using them for vote manipulation can lead to temporary or permanent banning of a user's Reddit account (Reddit, 2024). Posts and comments with a high score gain more visibility within a community. In contrast with prior work, we intentionally exclude network-based features (e.g., author name and interaction history) to ensure our methods' applicability across platforms like Reddit and GAB.

Pushshift's GAB data, whose post objects more closely resemble Twitter tweet objects (Twitter API, 2024), lacks the post title and score attributes seen in Reddit data collected through Pushshift. To account for this change, we reuse the text body of a GAB post object for both the title and selftext fields when passing GAB data through model trained on Reddit.

3.1 Data preprocessing

We exclude Reddit posts that were *stickied* (i.e. pinned to the top of a subreddit) since that can artificially increase the score of a post or comments in the post. To allow for more straightforward modeling and analysis, we examine only top-level comments — i.e., comments that reply directly to the post content. Since we only utilize large subreddits, there are a sufficient amount of top-level comments to allow us to achieve our objective of understanding the opinions of a community. We exclude any comment with the text "[deleted]" or "[removed]".

Comment scores serve as a proxy for community approval within each subreddit. We place comments into two categories: **Class Low (L)** if the score is less than or equal to -2; **Class High (H)** if comment score greater than 24; and we ignore all other comments. This asymmetrical classification reflects the observed skewness in score distribution, where negative scores are less frequent. The -2 threshold for Class L, while ensuring sufficient data for modeling, implies a minimum of three downvotes, a reasonable indicator of community dislike. This threshold also proved effective in our opinion model, yielding good results in inter-community opinion similarity analysis. Given the large user bases of all the subreddits studied, a static threshold was deemed appropriate for this research. The training and evaluation data includes all data points that were not filtered out by the data preprocessing steps. The median number of data points in the training dataset was approximately 56,500, with the 25th percentile at ~31,220 and the 75th percentile at ~174,400.

4 Predicting Opinion Polarity of Individual Comments

One of our goals is to quantify the similarity of subreddits based on their opinions and not merely the textual content. As a first step towards this goal, in this section we devise a method to model the opinion of distinct communities. For a given comment on a specific post, we treat its score as a self-labelled ground truth data set for that community's opinion. Class L posts indicate diapproval, Class H posts indicate approval; see Section 3.1.

We build the comment classification model using LLAMA2. Our model consists of the LLM followed by a single linear layer and a softmax layer for classification. The last token's vector representation is passed to the linear layer as the sentence embedding. The input to the model consists of the post title, selftext (post text), and the text of the comment being classified. These inputs are passed into the LLM in the following format:

```
<sub>r/subreddit</sub>
<post>post_title</post>
<comment>comment</comment>
<text>self_text</text>
```

Finally, the model predicts either Low or High. We trained LLAMA2 models until convergence or up to 20 epochs. Due to limitations of our computational resources, we experimented with only the 7 billion parameter version of LLAMA2. Fine-tuning was accomplished using 16-bit parameters and the LoRA technique (Hu et al., 2022), a parameter-efficient training strategy. We compare the LORA finetuned model against LLAMA2-7B under 0-shot and 5-shot setting, below. In the 5-shot setting, five random {post title, selftext, comment} pairs from the same subreddit were used. The specific prompt format for the 5-shot setting is detailed in Appendix A. The code used to train and evaluate our models can be found in our Github repository[1].

4.1 Evaluation

To evaluate opinion modeling and similarity, we selected 30 subreddits spanning diverse topics (politics, technology, religion, etc.). Results (Table 1) demonstrate that LLAMA2 fine-tuned with LoRA consistently outperforms both 0-shot and 5-shot base LLAMA2, even exceeding LLAMA3-8B 5-shot (result not shown for brevity). Interestingly, the inclusion of extra examples to learn from in the 5-shot setting did not enhance performance by much compared to the 0-shot setting.

Given the excellent performance of the LLAMA2 LORA models on several subreddits, we expect the model to be useful for predicting whether a comment, given a post, was — or would be — well received within a community. This approach could empower community members to write comments that resonate better within a specific community, or help them avoid those likely to be disapproved. The performance of the models was especially affected by uneven distributions of the H and L class datapoints (with L class data points being much lesser always). Our experiments on additional subreddits, not included in this analysis, reveal that model performance also degrades when the classifier lacks access to approximately 1,000 or more data points from each class.

[1] https://github.com/umass-forensics/community-similarity

Subreddit	L2-7B LoRA	L2-7B 0-shot	L2-7B 5-shot
politics	**0.90**	0.41	0.40
atheism	**0.89**	0.38	0.42
BanVideoGames	**0.88**	0.40	0.40
Bad_Cop_No_Donut	**0.88**	0.44	0.43
Christianity	**0.85**	0.37	0.41
exmuslim	**0.85**	0.43	0.46
conservatives	**0.84**	0.41	0.46
democrats	**0.83**	0.39	0.39
exmormon	**0.80**	0.37	0.40
TheRightCantMeme	**0.80**	0.42	0.41
PoliticalDiscussion	**0.80**	0.35	0.48
neoliberal	**0.79**	0.38	0.41
gunpolitics	**0.79**	0.39	0.42
Catholicism	**0.79**	0.37	0.42
Conservative	**0.78**	0.41	0.45
conspiracy	**0.78**	0.38	0.39
The_Donald	**0.77**	0.43	0.44
Libertarian	**0.77**	0.39	0.41
apple	**0.76**	0.38	0.42
RoastMe	**0.76**	0.41	0.44
AskALiberal	**0.76**	0.36	0.44
Android	**0.75**	0.38	0.44
AskTrumpSupporters	**0.75**	0.36	0.47
gaming	**0.75**	0.40	0.39
TheLeftCantMeme	**0.74**	0.40	0.42
JoeBiden	**0.72**	0.41	0.39
Republican	**0.72**	0.40	0.44
islam	**0.72**	0.41	0.42
AskThe_Donald	**0.71**	0.39	0.39
liberalgunowners	**0.65**	0.38	0.41

Table 1: F_1 score results from training the LLAMA2-7B model on the comment score classification task. Columns show results for the LLAMA2 model fine tuned using LoRA and LLAMA2 0-shot and LLAMA2 5-shot predictions.

5 Community Similarity based on both Content and Opinion

As discussed in Section 2, the results from the existing community similarity metrics focus heavily on content-based similarity, or they rely on manually selected set of seed communities to calculate opinion similarity. Furthermore, none of the existing community similarity metrics allow for cross platform (e.g. Reddit, GAB, or Twitter) comparison of communities.

In reality, subreddits can differ by opinion even if they are similar in terms of a topic. In this section,

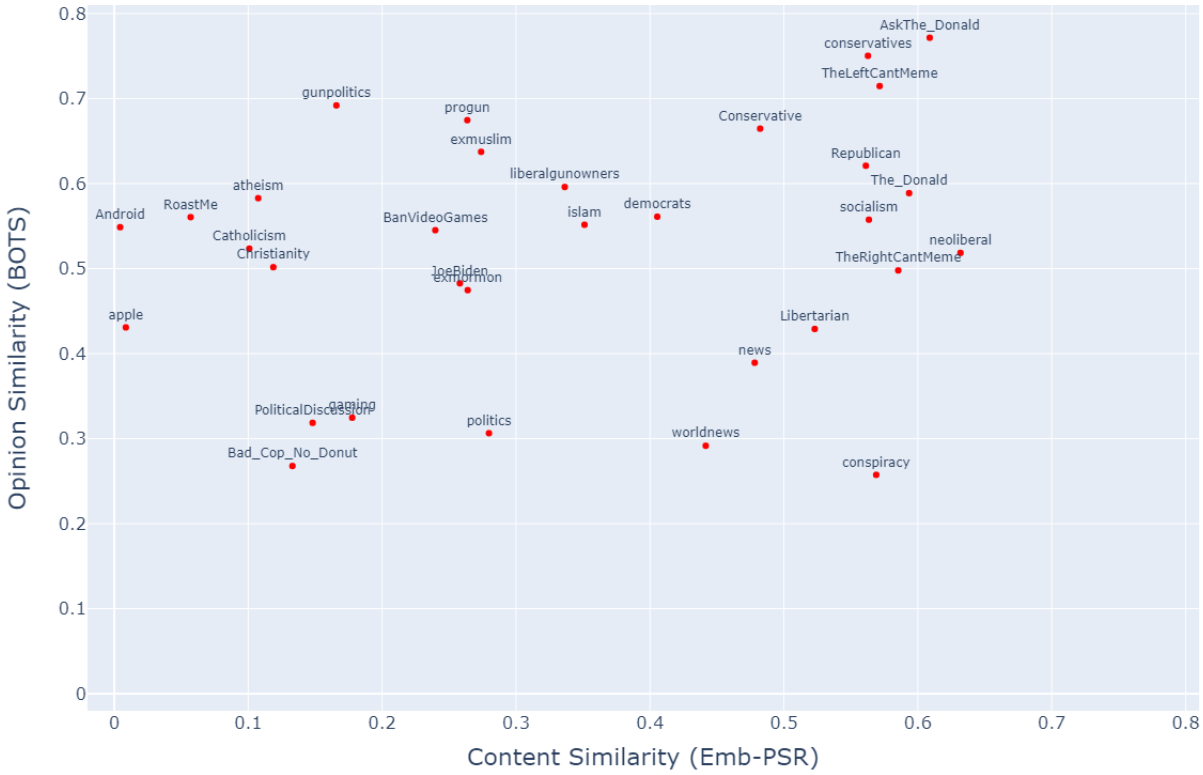

Figure 1: Content- and opinion-based similarity of GAB with subreddits.

we propose to quantify similarity among subreddits (or any collection of posts) according to both content and opinion. Figure 1 shows the results of our approach, for a methodology we detail below. Subreddits that are similar to the GAB in terms of posted content appear closer to 1 on the x-axis, explained in Section 5.1; whereas subreddits that are similar to opinions expressed on GAB appear closer to 1 on the y-axis, explained in Section 5.3.

5.1 Content similarity with Emb-PSR

We propose a novel method for calculating the content similarity of online communities, called Emb-PSR, that outperforms related work.

To calculate the similarity between subreddits, we take several steps. Let A and B represent two subreddits, each with a set of posts $A_1, A_2, \ldots$ and $B_1, B_2, \ldots$, respectively. Overall, we compute the fraction of posts in B that are at least as similar to posts in A as the posts in A are to the set A itself.

We solely consider a post's title and exclude the comments when calculating content similarity. We observed that comments often do not mention discussion topics explicitly and typically react to posts with context-dependent sentences. In contrast, post titles almost always explicitly mention the topic of discussion, making them a more reliable source of text for content similarity calculations.

First, we define a function $f()$ that determines the mean cosine similarity of a single post, p, to the set of all posts in a subreddit X, which is of size $|X|$. Note that we compute cosine similarity on Sentence-BERT embedding vectors (Reimers and Gurevych, 2019). Let $F(A)$ be the set of values from applying $f()$ to each post in A to all other posts in A.

$$f(p, X) = \frac{1}{|X|} \sum_{j \in X} \cos(p, X_j). \quad (1)$$

$$F(A) = \{f(A_i, \{A \setminus A_i\}),\ \forall i \in A\}. \quad (2)$$

Second, we create a summary statistic, $s(A)$, representing the "self-similarity" of A, from the distribution by quantifying the spread from its mean as follows. Let μ_A and σ_A represent the mean and standard deviation of the set $F(A)$, respectively. We define:

$$s(A) = \mu_A - c\sigma_A. \quad (3)$$

We selected constant c empirically as a hyperparamter, based on the performance on a validation set (20% of the data). In our experiments, $c = 0.75$ produced the best results.

Last, we measure how similar the posts in B are to A in the context of the self-similarity of the posts

Category	C2V	Emb-PSR
TV ($n = 11$)	0.44	**0.65**
artists/bands ($n = 7$)	0.67	**1.00**
books/magazines ($n = 9$)	0.42	**0.48**
cities ($n = 23$)	0.38	**0.78**
programming ($n = 8$)	0.62	**0.70**
computers ($n = 10$)	0.38	**0.42**
countries ($n = 32$)	**0.55**	0.53
food/drinks ($n = 7$)	0.57	**0.88**
gender/sexuality ($n = 7$)	**0.81**	0.75
hockey ($n = 10$)	0.89	**1.00**
internet/apps ($n = 8$)	0.20	**0.25**
machines ($n = 8$)	0.29	**0.48**
regions ($n = 11$)	0.27	**0.33**
religion/beliefs ($n = 9$)	0.31	**0.46**
video games ($n = 7$)	0.14	**0.67**
universities ($n = 16$)	0.58	**0.84**
mean	0.47	**0.64**

Table 2: A comparison of Emb-PSR to Community Vectors from Waller and Anderson (2021) for topic-based similarity across various categories. Each results shows, for a given category, the fraction of subreddits in the ranked list of the $n-1$ most-similar subreddits that belong in that category according to ground truth. The performance improvement of Emb-PSR is statistically significant according to a 95% c.i. paired t-test.

in A. Let $|B|$ be the number posts in B. We define a set $G(A, B)$ equal the subset of posts in B that are similar to posts in A in the context of the summary statistic. Finally, we define Emb-PSR(A, B) as the number of posts in B that are similar to posts in A, divided by the total number of posts in B. For computational efficiency, we sample a random set of 2500 posts from A to represent the subreddit.

$$G(A, B) = \{B_i \text{ if } f(B_i, A) > s(A), \forall i \in B\} \tag{4}$$

$$\text{Emb-PSR}(A, B) = \frac{|G(A, B)|}{|B|} \tag{5}$$

5.2 Emb-PSR Evaluation

Emb-PSR is aimed at solving content-based similarity; therefore, we expect subreddits discussing similar topics to be have a high similarity score. For example, we expect subreddits discussing hockey to have high similarity regardless of which team they support. We obtained labeled ground truth for assigning subreddits to categories from Israeli et al. (2022), shown in Table 2. Each category contains n subreddits, and each subreddit appears in one category. We restricted our evaluation to categories with at least 7 subreddits to avoid statistical outliers, leaving 183 subreddits in total.

We compare Emb-PSR against the community vectors developed in Waller and Anderson (2021) and Partridge et al. (2024). Certain subreddits in our analysis do not exist in the community embeddings provided by Waller and Anderson (2021). Hence, we use the code base from Partridge et al. (2024) to retrain the community embeddings on all Reddit data from 2006 to 2019 to mimic the settings used in Waller and Anderson (2021).

To evaluate both methods, we quantify their ability to find subreddits in each category. For each of the 183 subreddits, we find a candidate set of $n-1$ most-similar subreddits according to each method, where n is determined by the subreddit's category. We then determine the fraction of the candidates that are correct according to the ground truth; this metric is called *Hits@*n. For instance, to calculate Hits@n for Subreddit A from the TV category, we determine the similarity between Subreddit A and all other 183 subreddits, exclude Subreddit A itself, and then assess the number of subreddits from the TV category present within the top 10 most similar subreddits (since $n = 11$ for TV). This process is repeated for each subreddit within the category, yielding n results. Table 2 reports the mean of these n fractions for each method and category.

The average difference between the two methods for the 16 experiments is 0.17 in favor of Emb-PSR. A one-sided two-sample t-test confirms that the difference in performance between the methods is statistically significant for a 95% conf. interval.[2]

We hypothesize that Emb-PSR's improvement is due to its focus on content, rather than C2V's focus on user interactions which can introduce errors. For example, users from University X may discuss similar topics as University Y, but not post anything in University Y's subreddit. In such cases, University X and Y's C2V vectors will likely not be similar, whereas Emb-PSR, analyzing the posted content, will be able to assign a high similarity to Universities X and Y. We expect our approach to generalize across other online communities, since our approach relies on analyzing the post title rather than assumptions about user interactions.

5.3 Opinion similarity using BOTS

We introduce a new method to find the similarity between subreddits based on their opinions. In contrast with existing community similarity techniques (Mok et al., 2023; Phadke et al., 2021; Waller and

[2]Using a Shapiro-Wilk normality test, we confirmed that the distribution of differences is normal.

Anderson, 2021; Samory and Mitra, 2018), our technique is completely automated and doesn't require manually identified seed communities.

Our approach is based on several steps. First, we train a model for a specific subreddit, A, according to the method described in Section 4. Next, we avoid making predictions on topics that are not present in A. For example, a model trained on `r/politics` would not perform well predicting opinions about games taken from `r/gaming`. That is, when using an opinion model trained on A, we make a prediction on a post p (or any textual input) only if $f(p, A) > s(A)$ using Equations 1 and 3. This process is illustrated in Figure 2.

Once trained on the opinion similarity classification task within community A, our model can be used to compare any other subreddit or online text-based community to A without further retraining. This adaptability lends itself directly to tasks previously explored in the literature, such as identifying conspiracy related subreddits (Phadke et al., 2021) or politics related subreddits (Mok et al., 2023).

When we model opinions using community A's data and apply that model to predict opinions in community B, we can gauge community similarity based on the model's performance. High accuracy suggests similar opinions, while poor accuracy indicates opposing views. We refer to this approach as Opinion Transfer Similarity. In our current problem, the opinion modeling technique used is as described in Section 4. Since opinion is binary (heavily upvoted or downvoted) in our data, we call our similarity measure Binary Opinion Transfer Similarity (BOTS).

To evaluate BOTS, we use the F_1 score as our performance metric. This metric exhibits directionality, as random predictions on balanced datasets produce an F_1 score of 0.5. Achieving a high F_1 score requires substantial agreement between model predictions and ground truth data. Conversely, a low F_1 score indicates that the model's predictions are frequently the opposite of the correct labels. Thus, a model trained on community A's data will only yield a very low F_1 score on community B's data if the opinions of the two communities are largely opposed across many comments.

5.4 BOTS Evaluation

We evaluate opinion similarity in two different ways: by using a modified Hits@n metric; and by correlating the similarity obtained using our metric to the ordering obtained using media bias ratings from AllSides[3].

Political Bias Rating Correlation. Following previous work (Mok et al., 2023), we use AllSides to obtain a measure of the political leaning of a subreddit. AllSides categorizes news sources into five groups: Left, Lean Left, Center, Lean Right, and Right. Mapping these to numerical values from 1 (Left) to 5 (Right), we derive a subreddit's bias rating by averaging the bias ratings of news media sources shared within it (see Appendix B for details). To rank subreddits by their proximity to subreddit A, we order them based on the absolute difference between their bias rating and that of subreddit A. An ordering of the closest subreddits is also obtained from BOTS, Emb-PSR, Community2Vec (Martin, 2017), and the political partisanship distance from Mok et al. (2023). We then find the mean of the Spearman's rank correlation coefficient between the rankings obtained from media bias ratings and the rankings from the above methods, which is displayed in Table 3. When calculating the mean, we only consider subreddits that are either news or politics related (based on their name and description). We do this to account for the fact that a news organization with a political bias may post non-political articles as well. We also consider only subreddits that have posted at least 100 articles from labeled sources. Our results reveal that BOTS-based rankings exhibit the highest correlation with AllSides rankings, demonstrating its ability to order communities based on political opinion. However, many political subreddits, such as `TheLeftCantMeme`, `TheRightCantMeme`, `AskALiberal`, and `AskTrumpSupporters`, do not primarily share news articles. To include these communities in our evaluation, we have developed an alternative method, detailed below.

Bidirectional Hits@n. We have used Hits@n in our evaluation of Emb-PSR. However, unlike Emb-PSR, the closest and furthest subreddits to a given subreddit may belong to the same category. For example, in political opinions, we expect left-leaning subreddits to be closest to other left-leaning ones, followed by moderately political or non-political subreddits, and finally right-leaning subreddits. Therefore, we have modified the Hits@n metric to assess whether the proximity of both the closest and furthest n subreddits aligns with these expectations.

[3] `https://www.allsides.com/unbiased-balanced-news`

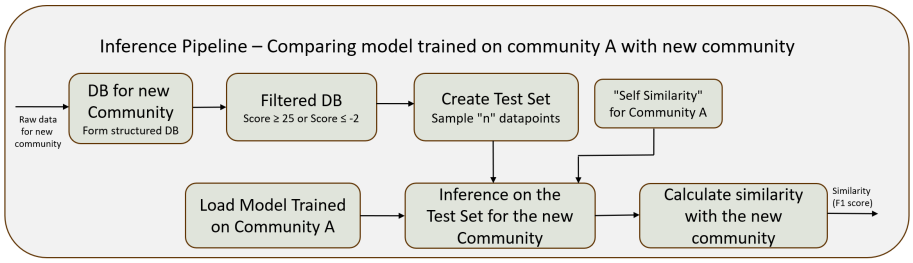

Figure 2: Inference Pipeline used in BOTS to compare Community A to a new community using the opinion model trained on community A.

	Bidirectional Hits@n (% in group)	Political Bias Rating Correlation
C2V	31.48	0.45
Emb-PSR	7.48	−0.09
BOTS	**44.44**	**0.51**
Partisanship-Dist	25.93	0.42

Table 3: Evaluation of opinion-based similarity on politics related communities.

Based on subreddit names, descriptions, and political bias ratings (see above), we have curated groups of left-leaning and right-leaning subreddits (refer to Appendix C for the lists). For each left-leaning subreddit, we expect other left-leaning subreddits to be the closest, while right-leaning subreddits should be the furthest away. This requirement is captured by defining the group cohesion as the difference between the number of subreddits from the same group and those from the opposing group. The Bidirectional Hits@N for a surbeddit S is then defined as the sum of the group cohesion of the N closest subreddits and the N furtherst subreddits divided by $2N - 1$ (since we ignore subreddit S from the calculation).

Table 3 demonstrates BOTS' superior performance in average Bidirectional Hits@n for left and right-leaning subreddits. We observe that Community2Vec tends to group all political subreddits together, regardless of their political leanings (Hits@n Top = 47.92%, Hits@n Bottom = 18.33%). Partisanship-Distance performs better at separating subreddits with opposing views (Hits@n Top = 18.75%, Hits@n Bottom = 31.67%), but fails to accurately capture subreddits with similar opinions. In contrast, BOTS not only excels at identifying subreddits with similar opinions (Hits@n Top = 45.83), but also significantly outperforms existing methods in identifying those with opposing views (Hits@n Bottom = 43.33).

6 Cross Platform Community Similarity

We intentionally omit features derived from social network interactions, such as usernames and user-community engagement patterns, from our community similarity calculations. This enables us to compare communities beyond the constraints of a single social network. To our knowledge, existing community similarity methods do not readily facilitate cross-platform comparisons. We demonstrate the cross-platform capabilities of both BOTS and Emb-PSR by comparing GAB with various subreddits.

To compare GAB with various subreddits, we have two options: train an opinion classification model on GAB data and then predict opinions on subreddit data, or load pre-trained subreddit models and use them to predict opinions on GAB data. We opted for the latter approach to avoid the computationally intensive task of training a new model.

As detailed in Section 3, our GAB dataset, obtained via Pushshift, spans from June 2018 to October 2018. Applying similar preprocessing steps as with the Reddit data, we randomly sampled 10,000 posts with a score of at most −2 and 10,000 posts with a score of at least 25. We then loaded pre-trained subreddit models and had them predict the score classes of these 20,000 posts to obtain the similarity (F_1 score on score classification) between GAB and other subreddits. However, post structure differs between Reddit and GAB. Reddit includes comments within the context of a post and selftext, while GAB posts in our dataset are independent and lack selftext. To address this, we found that repeating the GAB post text as both the selftext and comment was sufficient for comparison with subreddits.

We present the results in a "similarity plot" in Figure 1. The x-axis represents content similarity (Emb-PSR) and the y-axis represents opinion similarity (BOTS). We observe that GAB is similar in content to several politics related subreddits including `r/conservatives`, `r/neoliberal`, and `r/Republican`. Among these subreddits,

however, its opinions appear more similar to right-leaning subreddits (e.g. r/conservatives, r/AskThe_Donald) compared to left-leaning subreddits (e.g. r/neoliberal, r/socialism). The r/TheLeftCantMeme, a right-leaning subreddit, and r/TheRightCantMeme, a left-leaning subreddit, demonstrate the usefulness of a similarity plot — the two subreddits are very close in terms of their content similarity to GAB, but GAB is much more similar to r/TheLeftCantMeme in its opinions compared to r/TheRightCantMeme. BOTS and Emb-PSR, when presented in a similarity plot, allow us to analyze subreddit similarity in a more holistic manner by considering both content and opinion similarity.

7 Conclusion

Our research advances the understanding of online communities and their similarities. By training LLAMA2 models on the task of comment score classification, we demonstrate their impressive performance across diverse subreddits, showcasing their potential to accurately predict community reactions. We decompose the task of calculating community similarity into two fundamental components: opinion similarity and content similarity. We then introduce two novel techniques to quantify these dimensions within online communities. Our evaluation demonstrates that our techniques outperform existing community similarity measures by a considerable margin on their individual tasks. Furthermore, our similarity measures are less dependent on platform-specific features and do not require manually identified seed communities, enabling cross-platform comparison of communities as demonstrated through our comparison of GAB with various subreddits. This research paves the way for a more comprehensive understanding of online communities, with potential applications in a broad range of tasks ranging from content moderation to community recommendation.

8 Limitations

Fine-tuning the LLAMA2 model on the opinion modeling task is computationally expensive and requires a GPU with substantial memory capacity (we used a single NVIDIA A100 GPU with 80GB VRAM). Post-training, inference remains more resource intensive than existing methods and likely necessitates a GPU. We tried training a single model for all the subreddits. However, the performance of this model was significantly worse on several subreddits compared to the use of individual models.

Currently, our opinion models do not process images or videos embedded in Reddit posts. This decision prioritizes text, the most common communication medium on social media platforms, to ensure the widest possible applicability of our comparison techniques. Notably, even in communities with numerous image-based posts, like r/TheRightCantMeme, our text-based opinion model demonstrates strong performance, achieving an encouraging F1 score of 0.8. In the future, we may explore incorporating a multimodal model with optional image/video inputs to further enhance our analysis. Importantly, the BOTS technique remains compatible with any opinion model and would not require modification even if the underlying opinion model is upgraded. However, platforms must support some form of positive and negative feedback on posts/comments to train the opinion model effectively.

9 Ethical Considerations

Our research allows for the comparison of online communities based on their expressed opinions, facilitating the identification of subreddits with similar or differing viewpoints. This could be utilized to curate a more diverse feed encompassing a range of perspectives on a given topic, or conversely, to construct echo chambers composed of communities holding like-minded opinions. However, echo chambers can also be formed through lower-level recommendations, such as those at the post level. While our techniques could be misused to create echo chambers, such chambers are not a new phenomenon and can form through various mechanisms. We also recognize the techniques could be misused by a government or an organization to maliciously target specific online communities that share the same viewpoint. We believe our techniques do not present any elevated risk over existing methods for identifying similar communities. We believe the potential benefits of our findings for community discovery, content moderation, and social media analysis outweigh the risks.

References

Fatma S. Abousaleh, Wen-Huang Cheng, Neng-Hao Yu, and Yu Tsao. 2021. Multimodal Deep Learning Framework for Image Popularity Prediction on

Social Media. *IEEE Trans. on Cognitive and Developmental Systems*, 13(3).

Jason Baumgartner, Savvas Zannettou, Brian Keegan, Megan Squire, and Jeremy Blackburn. 2020. The pushshift reddit dataset. In *Proc. AAAI Intl. conference on web and social media*, volume 14, pages 830–839.

Eshwar Chandrasekharan, Mattia Samory, Shagun Jhaver, Hunter Charvat, Amy Bruckman, Cliff Lampe, Jacob Eisenstein, and Eric Gilbert. 2018. The Internet's Hidden Rules: An Empirical Study of Reddit Norm Violations at Micro, Meso, and Macro Scales. *Proc. ACM Hum.-Comput. Interact.*, 2.

Srayan Datta and Eytan Adar. 2019. Extracting Inter-Community Conflicts in Reddit. *Proc. Intl. AAAI Conf. on Web and Social Media*, 13.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *Proc. Intl. Conf. Learning Representations (ICLR)*.

Abraham Israeli, Shani Cohen, and Oren Tsur. 2022. Unsupervised discovery of non-trivial similarities between online communities. *Expert Systems with Applications*, 206:117900.

Srijan Kumar, William L. Hamilton, Jure Leskovec, and Dan Jurafsky. 2018. Community interaction and conflict on the web. In *Proc. World Wide Web Conference*, pages 933–943.

Hung-Hsiang Lin, Jiun-Da Lin, Jose Jaena Mari Ople, Jun-Cheng Chen, and Kai-Lung Hua. 2022. Social Media Popularity Prediction Based on Multi-Modal Self-Attention Mechanisms. *IEEE Access*, 10:4448–4455.

Ruoxue Ma, Jiarong Xu, Xinnong Zhang, Haozhe Zhang, Zuyu Zhao, Qi Zhang, Xuanjing Huang, and Zhongyu Wei. 2023. One-model-connects-all: A unified graph pre-training model for online community modeling. In *Findings of the Association for Computational Linguistics: EMNLP*, pages 15034–1504.

Trevor Martin. 2017. community2vec: Vector representations of online communities encode semantic relationships. In *Proc. ACL Workshop on NLP and Computational Social Science*, pages 27–31.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proc. Intl. Conf. Neural Information Processing Systems*, volume 26, page 3111–3119.

Lillio Mok, Michael Inzlicht, and Ashton Anderson. 2023. Echo tunnels: Polarized news sharing online runs narrow but deep. *Proc. AAAI Intl. Conf. on Web and Social Media*, 17.

Virginia Partridge, Jasmine Mangat, Rebecca Curran, Ryan Mcgrady, and Ethan Zuckerman. 2024. Here be livestreams: Trade-offs in creating temporal maps of reddit. In *Proc. ACM Web Science Conference*, pages 81–91.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Shruti Phadke, Mattia Samory, and Tanushree Mitra. 2021. What makes people join conspiracy communities?: Role of social factors in conspiracy engagement. *Proc. ACM on Human-Computer Interaction*, 4(CSCW3).

Reddit. 2023. Is it ok to create multiple accounts?

Reddit. 2024. What constitutes vote cheating or vote manipulation?

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Mattia Samory and Tanushree Mitra. 2018. Conspiracies online: User discussions in a conspiracy community following dramatic events. *Proc. International AAAI Conference on Web and Social Media*, 12(1).

Nathan TeBlunthuis and Benjamin Mako Hill. 2022. Identifying competition and mutualism between online groups. *Proc. Intl. AAAI Conference on Web and Social Media*, 16.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Data dictionary: Standard v1.1 Twitter API. 2024. Tweet object.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(86):2579–2605.

Isaac Waller and Ashton Anderson. 2019. Generalists and specialists: Using community embeddings to quantify activity diversity in online platforms. In *The World Wide Web Conference*. ACM.

Isaac Waller and Ashton Anderson. 2021. Quantifying social organization and political polarization in online platforms. *Nature*, 600(7888).

Kai Wang, Penghui Wang, Xin Chen, Qiushi Huang, Zhendong Mao, and Yongdong Zhang. 2020. A feature generalization framework for social media popularity prediction. In *Proc. ACM Intl. Conf. Multimedia*. ACM.

Bo Wu, Wen-Huang Cheng, Peiye Liu, Bei Liu, Zhaoyang Zeng, and Jiebo Luo. 2019. Smp challenge: An overview of social media prediction challenge 2019. In *Proc. ACM Intl. Conf. Multimedia*.

Kele Xu, Zhimin Lin, Jianqiao Zhao, Peicang Shi, Wei Deng, and Huaimin Wang. 2020. Multimodal deep learning for social media popularity prediction with attention mechanism. In *Proc. ACM Intl. Conf. Multimedia*. ACM.

A Opinion Modeling: 5-shot prompt

Presented below is the prompt format for a 5-shot prompt when trying to make a prediction on a comment from the subreddit "AskThe_Donald"

```
[INST] <<SYS>>
You are a machine learning model trained to predict whether a comment on a Reddit post in the subreddit r/AskThe_Donald will receive a lot of upvotes or downvotes based on the POST TITLE, SELF TEXT, and the COMMENT text itself.

Consider factors such as the tone, language used, relevance of the comment to the original post when making your prediction, and community norms in r/AskThe_Donald. In your analysis, try to consider how users on r/AskThe_Donald might typically react to a comment in the context of the specified post.

Here are some examples:

POST TITLE: Post Title 1
COMMENT: Comment 1
SELF TEXT: Self Text 1
PREDICTION: UPVOTE

POST TITLE: Post Title 2
COMMENT: Comment 2
SELF TEXT: Self Text 2
PREDICTION: DOWNVOTE

...

POST TITLE: Post Title 5
COMMENT: Comment 5
SELF TEXT: Self Text 5
PREDICTION: DOWNVOTE

Output either "UPVOTE" or "DOWNVOTE" to indicate your prediction for the following comment. No explanation needed. Just output "UPVOTE" or "DOWNVOTE".

<</SYS>>
POST TITLE: Post Title
COMMENT: Comment
SELF TEXT: Self Text
PREDICTION:
[/INST]
```

Subreddit	Bias Rating
socialism	2.04
democrats	2.07
politics	2.14
JoeBiden	2.19
neoliberal	2.21
islam	2.31
news	2.32
apple	2.35
exmormon	2.35
atheism	2.36
Bad_Cop_No_Donut	2.38
Christianity	2.43
gaming	2.45
Android	2.49
worldnews	2.53
liberalgunowners	2.54
Libertarian	2.85
exmuslim	2.86
Catholicism	3.00
conspiracy	3.03
gunpolitics	3.37
progun	3.75
The_Donald	4.26
Republican	4.34
AskThe_Donald	4.40
Conservative	4.41
conservatives	4.44

Table 4: Subreddit Bias Ratings calculated using the labels from AllSides.com

B Political Bias Ratings based on Allsides.com

The political bias ratings calculated using the labels from AllSides is as seen in Table 4. Only the subreddits used in our opinion similarity study that have at least a 100 articles from sources labeled in allsides.com are present in this table.

C Right and Left leaning subreddits

We label the following subreddits as right-leaning and left-leaning in our analysis based on the subreddit title, description, and the bias ratings from Table 4.

Right Leaning : Conservative, conservatives, The_Donald, AskThe_Donald, Republican, TheLeftCantMeme

Left Leaning : democrats, JoeBiden, neoliberal, politics, TheRightCantMeme, worldnews