Predicting Subreddit Toxicity Using Network Properties

Kevin Chang, Eric Matsumoto, Anna Mitchell

I. Abstract

There is a widespread perception that dialogue has deteriorated on the Internet. Social networks promising utopia instead spread vitriol. Emerging research on social networks has attempted to characterize and explain dialogue, largely focused on Facebook and Twitter. Reddit, however, has now surpassed Facebook to become the third-most-popular site on the Internet. Organized around interest-defined subreddits, it is a promising platform for studying Internet dialogue. We address whether social network structure affects the characteristics of dialogue. Specifically, we examine whether there is a correlation between network structure and the toxicity of dialogue on a subreddit. We build an embedding for each subreddit based on the internal characteristics of its graph and relationships with other subreddits to predict the toxicity of a subreddit. We implement an iterative node classification algorithm using support vector machines on a large dataset of 40,000,000 comments over 2,000 subreddits to predict subreddit toxicity. Our novel contribution to the fields of social network analysis and natural language processing is predicting content characteristics based on network features.

II. Introduction

At its inception, many optimistic commentators saw the Internet as an unprecedented platform for free expression. One judge in the 1996 *Reno vs ACLU* case gushed that ”[i]t is no exaggeration to conclude that the Internet has achieved, and continues to achieve, the most participatory marketplace of mass speech that this country – and indeed the world – has yet seen.”[1] Even more recently, Supreme Court Justice Kennedy called the Internet ”essential venues for public gatherings to celebrate some views, to protest others, or simply to learn and inquire.[2] Unfortunately, the last few years suggested that this vision was flawed. News stories now describe bots spreading propaganda,[3] trolls doxxing unlucky targets,[4] and Twitter and Facebook users retreating into echo chambers that confirm or even radicalize their beliefs.[5]

Emerging political research has tried to gauge the prevalence and characteristics of hate speech on the Internet. Siegel et. al employ multiple methods of text classification to characterize 2016 U.S. election-related hate speech over more than 1 billion tweets.[6] While much of the speech examined here occurred on social networks, little research connects network characteristics and the characteristics of speech on those networks. Our research begins to fill that gap.

Social network analysis, while still new, is a far older field than the study of Internet speech. Many existing network research papers have tried to characterize the structure of social media networks. Other research has begun to draw connections between content characteristics and network structure. For example, one study found that anger spread faster than any other emotion on social media networks.[7] Again, as in political science, little research has tried to determine whether the structure of social media networks can predict anything about the characteristics of dialogue on those networks.

We specifically wanted to analyze whether there is any relationship between network structure and the constructiveness of dialogue on those networks. This problem matters because platforms must balance providing an open forum that encourages user participation, with discouraging profanity or ad hominems that drive users away. To tackle this problem, we needed a ground truth source of data for constructiveness in dialogue, along with a large and diverse dataset of Internet comments. Several projects have attempted to use machine learning to

label hate speech. Gitari et. al trained a classifier relying on a lexicon to categorize hate speech about race, nationality, and religion using polarity scores and semantic labels, with significant success.[10] However, we wanted to use an out-of-the-box solution that, with some modification, would allow us to quickly generate a ground truth data set. We settled on using Google's Perspective API, which labels any sample of text with a "toxicity" probability, or likelihood of making participants leave the conversation.

We chose to study the toxicity of comments on Reddit, one of the most popular social media networks on the Internet, with 330 million monthly active users.[14] It is organized around subreddits catering to interests like r/politics or r/malefashionadvice. Like Facebook and Twitter, however, it has not escaped pitfalls of anonymous platforms. Some subreddits have harbored so much harsh language that they have been banned: for example, r/incels, a community for involuntarily celibate men who blame the sex of women for their romantic failures. Reddit is a powerful platform for discussion and debate, but does not come without its cesspools.

On Reddit, all posts are public, making it a fitting source for social science research. Reddit encompasses many smaller well-defined communities in subreddits, whose characteristics we could easily study and compare. Therefore it seemed like an ideal dataset for studying the relationship between network structure and toxicity.

After examining several methods for prediction, we settled on node classification because it allowed us to incorporate information about the network characteristics of a subreddit along with labels of known neighbors. We constructed a subreddit graph with edges between subreddits with common users. Using the graph, we extracted unique embeddings for each node, based on the subreddit's network attributes - from both its own graph of users and posts and in its relationships with other subreddits - and any known labels of neighboring subreddits. We then trained simple and iterative classifiers using SVMs and other methods using these embeddings to classify the unknown nodes.

III. Related Work

A. Analysis of Reddit

There is a large body of existing work around social networks in general and Reddit in particular. Less formally, Trevor Martin of 538 performed Reddit "addition" and "subtraction" of subreddits.[11] Addition of two subreddits A and B was defined as the union $S_A \cup S_B$, where S_A and S_B are the sets of users in A and B, respectively. Subtraction of B from A was defined as $S_A \backslash S_B$. These methods allowed Martin to discover surprising similarities between users. For example, subtracting the users of r/Politics from those of r/TheDonald produces a group of users which most closely overlaps with the notorious subreddits r/fatpeoplehate and r/TheRedPill. This research provided fascinating insight into the relationships between subreddits. However, we wanted to analyze relationships in more depth by considering similarities in network structure, not only common users.

More formal research has analyzed the graph structure of Reddit. Olson and Neal, like Martin, try to model overlapping user interests, but rely more heavily on network analysis.[9] They create a graph where nodes are subreddits and edges exist between nodes when a significant number of users post in both subreddits. Their work provides insight into the structure of Reddit. They found that Reddit, like many other social networks, follows a small-world network structure. Perhaps surprisingly, a full 80% of subreddits do not contain edges to other subreddits, because they do not share a significant number of shared posters. This suggested to us that studying the characteristics of the subreddit's own network, its "intra-subreddit" characteristics, might be more useful than the "inter-subreddit" characteristics of the subreddit's neighborhood. If a subreddit's neighborhood is defined as the subreddits with whom it shares common users, then a small subreddit neighborhood might not be as useful as examining the graph structures of the subreddit itself. Olson and Neal also found the network to be scale-free, with a power-law degree distribution. Finally, Reddit is modular. Olson and Neal detected around 59 distinct clusters in the network, representing meta-communities grouped under common interests.

Some work has also been done at the subreddit user level. Buntain and Golbeck investigate roles within Reddit communities.[13] They specifically sought to identify ”answer-people,” or users whose main interactions were answering other users' questions. They used a directed graph, in which nodes are users and edges are replies to comments. Buntain and Golbeck hand-labeled roles and calculated network metrics to create features for nodes. Using Scikit's decision tree algorithm, they were able to identify the ”answer-person” on average 80% of the time. Finally, they found that such ”answer-person” individuals often participated within a community but not across communities. Similar to other research efforts, they found that structural rather than content features could determine roles within a network. In our work, we focus on toxicity, not roles of individual nodes, which is admittedly farther removed from network analysis than roles. However, we hope to similarly avoid content-based approaches and instead use network features .

Olson and Neal's work focused more broadly on the subreddit graph, where nodes are subreddits and edges are based on common users. Buntain and Golbeck's research, on the other hand, examines the network properties of the subreddits themselves, creating graphs based on users and posts. Our research combines these techniques with two levels of granularity. First, we classify subreddits' overall toxicity and consider the toxicity levels of its neighbors and position in the broader subreddit graph. However, we also consider attributes of the subreddit itself. For this, we rely on network rather than content features: for example, examining patterns in the clustering of users who commented on related posts. We improve on existing models of Reddit by creating a mirrored graph structure in which we examine the relationship between subreddit nodes, but also examine the structure of nodes within each subreddit.

B. Node Classification

The classic problem in supervised machine learning is predicting labels of unseen data given labeled training data. One version is the graph labeling problem: given a social network with some labeled nodes, how can we accurately label the rest of the nodes? In our project, we classify subreddits as toxic or nontoxic given a few labeled, ground-truth subreddits that we know to be toxic or non-toxic. Some reddits are obviously harmless or harmful: the misogynist r/TheRedPill is clearly toxic, unlike r/animalsbeingbros. But what can we conclude about subreddits whose properties are not obvious, like r/politics?

Node classification techniques allow rich prediction by incorporating into the feature vectors both neighborhood labels and attributes of the node itself. This allows us to consider network characteristics of the subreddit graph along with any known or tentative labels of its neighbor nodes.

Neville and Jensen describe several techniques for node classification.[15] Relational classification, the simplest, generates labels for a node based only on the labels of its neighbors, without considering any attributes of the node. It generates a conditional probability of a label for each node as follows:

Repeat for each node i and label c:

$$P(Y_i = c) = \frac{1}{|N_i|} \sum_{(i,j) \in E} W(i,j) P(Y_j = c)$$

W(i,j) is the edge strength from i to j
$|N_i|$ is the number of neighbors of I

Labeling continues until hopeful convergence. However, this model has severe limitations. First, it is not guaranteed to converge. Second, it does not consider node attributes. In classifying each subreddit, along with considering its neighbors' labels, we also wanted to build a rich feature vector that represented the characteristics of its own network. For example, was a high number of communities in a subreddit correlated with toxicity? Because we planned to examine relationships between network structure and content, this model was too limiting.

An iterative node classification model, as summarized by Bhagat and Neville, was better-suited for our purposes. Iterative classification has been successfully used in a variety of applications, such as computer vision in Hummel Zucker (1983) and loopy belief propagation in Murphy Weiss (1999).[16] Iterative node classification requires extracting a feature vector for each node using its attributes and any known neighbor

attributes, then repeatedly training a classifier on the feature vector to predict a label for the node. Relational classification considers only the labels of neighboring nodes but not the attributes of the node itself. Iterative classification is therefore a more flexible model.

Neville and Jensen's original paper describes an iterative classification algorithm in which the model is trained on a fully labeled training set, then applied to a test set of N instances. At each iteration, the dynamic relational features are recalculated, and then node labels re-predicted by sorting the inferences by probability, and accepting k class labels. After m iterations, the algorithm outputs the top-ranked final labels.

Several criteria are required for iterative classification to be useful.[15] If static node attributes alone could successfully classify, then an iterative approach is unnecessary. This is because dynamic attributes that change in every iteration do not significantly improve the prediction. Also, the graph must be sufficiently connected: if connections between nodes are sparse, then node relationships are less powerful predictors. The required degree of linkage is unclear. These stipulations could limit the usefulness of iterative node classification on our dataset. However, the graph of subreddits was well-connected. Therefore, it seemed plausible that dynamic neighborhood attributes could improve prediction.

Existing implementations of iterative classification relied primarily on dynamic attributes. If static attributes alone can predict a label, then iteration was unnecessary. Instead, existing techniques relied heavily on a node's neighborhood features based on the labels of its neighbors.

Since we were unsure whether iteration over dynamic attributes would improve our prediction, we trained both a static classifier and an iterative classifier.

IV. Data

A. Data Collection

Google BigQuery hosts all Reddit comments posted since January 2015.[17] Using SQL commands on Google Cloud Engine, we scraped a large dataset of approximately 24,000,000,000 Reddit comments over around 1,980 subreddits.

We calculated the top 2,000 subreddits by number of unique users. First, the set should be diverse in content and tone, spanning toxic and non-toxic and subreddits. This list met our diversity criterion, with members running the gamut from large general-interest subreddits like AskReddit to fringe communities such as TheRedPill, a popular hub for misogynists. Second, to build a meaningfully connected graph, the subreddits should share users. The 2,000 most popular subreddits were better-connected than smaller subreddits, satisfying this second criterion.

We also hand-checked and filtered 20 foreign-language subreddits, such as r/montreal or r/de. We filtered out non-English subreddits since 1) the toxicity API was designed for English, and 2) it would be difficult for us to spot-check foreign languages. This left 1,980 subreddits total.

Fig. 1. An example Reddit comment


```
{
    "body": "Did you actually have romantic feelings
         for him?",
    "author": "unlucky_ducky",
    "created_utc": 1541663617,
    "link_id": "t3_9v7n4r",
    "subreddit_id": "69e6ce46531d",
    "parent_id": "t3_9fi9u5",
    "score": "2",
    "retrieved_on": "1538725335",
    "subreddit": "relationships",
    "id":  "e5qx8ym"
}
```


From each subreddit, we scraped 20,000 comments. Of these, we filtered out all comments whose body was "[removed]." Each comment was associated with a range of information including basic attributes like content and score, but also relational attributes such as author, parent thread, and subreddit id. These ID fields would allow us to create graphs connecting authors and posts.

B. Toxicity Score Labeling

Google's Perspective API provides an estimated "toxicity" score for any sample of text. A "toxic" comment is defined as one likely to make a

participant leave the conversation. Using the API, we calculated a ground truth toxicity score for each subreddit. For each subreddit comment, the API could return a detailed toxicity score.

We wanted to be certain that the toxicity score provided by the Google API was an accurate baseline. Unfortunately, Hosseini et. al found that the API was gameable.[12] An adversary could slightly modify his comment to avoid being flagged as toxic. For example, misspelling "idiot" as "idiiot" reduced its toxicity score from 84% to 20%. It was beyond the scope of this project to check for all misspellings. However, we could spot-check whether subreddit scores were sensible, based on our knowledge of these subreddits.

Also, we wanted to build a more sophisticated model than an average. Our original approach to calculating toxicity scores for subreddits was to simply take the average of the toxicity scores for all comments, weighing the score of each comment equally without any additional considerations. We found that this method led to inaccurate, or at least unintuitive, toxicity scores. Our new approach addresses a few of the shortcomings of the original approach.

Removing enthusiastic profanity. One of the issues affecting our original score calculations was the outsized effect of certain words on comment scores. For example, "holy shit that's so cool" yields a 0.84 toxicity score, whereas the phrase "holy that's so cool" scores 0.07. While swear words like "shit" may be offensive alone, in context these words are often used to increase the positivity of a comment. However, other words, like "fucking," were generally used in a negative context. After sampling several comments and comparing scores with and without profanity, we decided to calculate scores for comments containing "shit", "damn", or "crap" differently. We compared two methods for improving the score: simply calculating the score with the words removed, and taking the average of the sanitized and original versions. The first approach produced more accurate scores.

Weighing by length and amplifying toxicity. Since longer comments often leave larger impact in a conversation, we also weighted the scores by the comment length and amplified the effects of larger scores per the following equation:

$$T(s) = \sum_{c \in C} \frac{\gamma \tau(c)^2 \; e^{\frac{len(c)}{\psi}}}{\sum_{c' \in C} e^{\frac{len(c')}{\psi}}}$$

where s is the subreddit, C is the set of all comments for that subreddit, $T(s)$ calculates the toxicity score for subreddit s, $\tau(c)$ is the raw score reported by Google's Perspective API, $len(c)$ is the length of comment c, ψ is the length of the longest comment in all subreddits, and γ is a constant weighting factor.
Intuitively, we are amplifying the effects of the most toxic comments by squaring it, and weighing the score of comments by the length of the comment.
To check that our toxicity scores represented a strong baseline, we ranked subreddits by toxicity score. The top 10 most toxic subreddits were dominated by pornography, which suggested that this toxicity calculation was a suitable baseline. The 10 least toxic subreddits, however, spanned many innocuous topics from animals to free karma to tech support.

Fig. 2. Top 10 Most Toxic Subreddits

Subreddit	Toxicity Score
MassiveCock	1.6544
JerkOffToCelebs	1.5234
AmItheAsshole	1.4461
penis	1.3258
Cumtown	1.3104
copypasta	1.3050
assholegonewild	1.2592
ratemycock	1.2311
opieandanthony	1.2030
WouldYouFuckMyWife	1.1891
jobudsn	1.1805
cock	1.1747

Fig. 3. Top 10 Least Toxic Subreddits

Subreddit	Toxicity Score
CatsStandingUp	0.01538
cotrader	0.02441
FreeKarma4You	0.04142
AskOuija	0.04222
FreeKarma4U	0.04517
pokemongotrades	0.05440
BCoinsg	0.05832
SuggestALaptop	0.05957
hardwareswap	0.06127
GameSale	0.07052
BoldmanCapital	0.07056
reactjs	0.07193

Our results aligned with our expectations for a toxic subreddit. Therefore our toxicity baseline model seemed satisfactory.

Fig. 4. One hop and two hop subreddit neighbors of CatsStandingUp in the subreddit-subreddit graph.

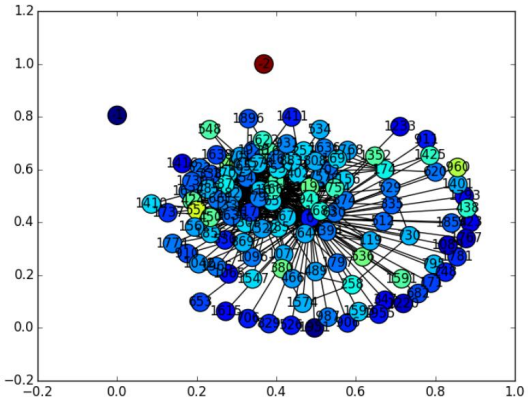
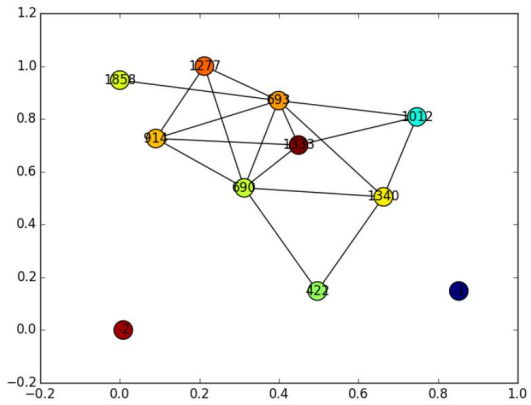


Fig. 5. One hop and two hop subreddit neighbors of MassiveCock in the subreddit-subreddit graph.



Examining the neighborhoods of subreddits in a subreddit-subreddit graph in figures 5 and 6, we also see a stark difference in the most toxic and least toxic subreddits. The neighborhoods are based on on one- and two-hop neighbors of the given subreddit. Only edges with weights greater than 100 (more than 100 users in common) are considered. Extreme blue nodes are less toxic, while extreme red nodes are very toxic (Note: for reference, the node labeled -1 is extremely toxic, while the node labeled -2 is not very toxic). These neighborhoods show that toxic subreddits have more toxic neighbors and non-toxic subreddits have more non-toxic neighbors. This suggests that iterative classification is a powerful technique

V. Models

A. Graph Model

1) Graph 1: Subreddit-Subreddit Graph

Nodes were subreddits; edges existed if the subreddits shared at least one common user. We see that this graph is strongly connected and nearly all of the nodes belong to one large cluster. A majority of high-degree nodes was surprising, since since the Olson and Neal paper claimed that Reddit's degree distribution followed a power law. However, our sample is the top 2,000 most popular subreddits by unique users. This suggests that there is strong overlap between the most popular subreddits, and perhaps a large number of poorly-connected smaller subreddits.

2) Graph 2: Author-Thread Graph

Author nodes and thread nodes, connected if the author commented on the thread.

Fig. 6. Subreddit-subreddit graph

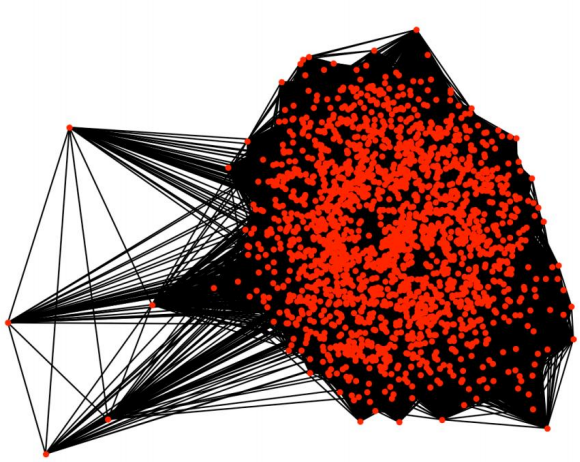


Fig. 7. Degree distribution of subreddit-subreddit graph

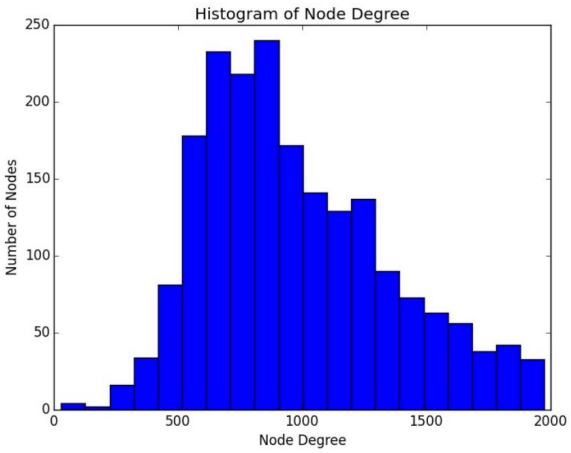


Fig. 8. Author-Thread Bipartite Graph

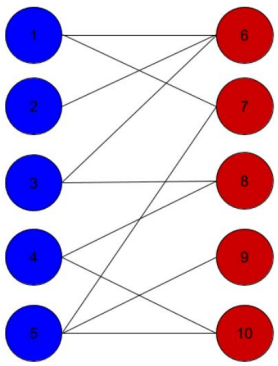
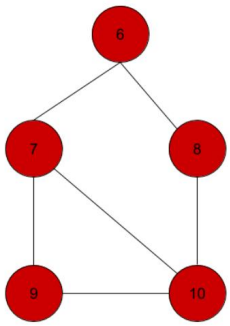


Fig. 9. Author-Author Folded Graph



3) Graph 3: Author-Author Graph

Author nodes, connected if they commented on at least one common thread.

4) Graph 4: Thread-Thread Graph

Thread nodes, connected if they share at least one author.

B. Prediction Model

Iterative classification improves on a normal classifier by considering neighborhood graph labels along with node attributes. We split the data into a training set consisting of 80% of the data and a test set consisting of 20% of the data, with an even proportion of toxic and non-toxic subreddits in each. We trained a basic classifier and iterative classification using several techniques, including logistic regression, SVM with multiple kernels, and random forests.

Basic classifier.

Compute feature vector based on node network attributes and run one-time prediction with an SVM.

Iterative classification, bootstrapped.

Follow the textbook iterative classification algorithm. The algorithm consists of the following steps:

C. Features

1) Static

- *Average threads per author.* Average degree of author nodes in the bipartite author-thread graph.
- *Average authors per thread.* Average degree of thread nodes in the bipartite author-thread graph.

Algorithm 1 Iterative node classification, bootstrapped.

1: Train a classifier on static attributes alone.
2: Train a classifier on both static and dynamic attributes.
3: Predict labels for the test set using the static classifier.
4: **do**:
5: Update node labels with predictions.
6: Calculate dynamic attributes as necessary.
7: Predict test set labels with full classifier.
8: **while** not converged

- *Average comments per author*. Total number of comments in a subreddit, divided by total number of authors commenting on that subreddit.
- *Clustering coefficient of the author graph.* Measure of cliquishness in the author graph - do groups of authors tend to comment on the same threads?
- *Clustering coefficient of the thread graph.* Measure of cliquishness in the thread graph - are groups of threads commented on by the same authors?
- *Maximum connected component of the author graph.* Measure of largest community of authors based on their common threads
- *Maximum connected component of the thread graph.* Measure of largest community of threads based on their common authors
- *Average shortest path in the author graph.* Measure of connectedness of authors based on their common threads.
- *Average shortest path in the thread graph.* Measure of connectedness of threads based on their common authors.
- *Degree in the subreddit network.* The number of edges to other nodes in the subreddit network.
- *Betweenness centrality in the subreddit network.* The node betweenness centrality of a given node, or the probability that a shortest path passes through the node. Intuitively, who are the hubs?
- *PageRank of subreddit in the subreddit network.* The importance of the node in the Graph as measured by PageRank (how likely we are to land on this node while traversing the graph).
- *Harmonic centrality in the subreddit network.* Intuitively, who are the bridges in the network? Very similar to closeness centrality.
- *Eigenvector centrality in the subreddit network.* Eigenvector centrality is a measure of influence in the network. A high score means that it is connected to other nodes with high scores. Intuitively, if degree measures walks of count 1, then eigenvector centrality measures walks of infinite length.

2) Dynamic

- *Toxic percentage of labeled neighbors.* Average weighted by edge and toxicity.

VI. Results

We trained the the model following a two-step process: **Training an optimal baseline classifier** testing naive Bayes, SVMs, and random forests with various hyperparameters. **Training an iterative classifier** using the optimal baseline classifier.

Step 1: Baseline Classifier

Naive Bayes, Random Forest, SVM (linear kernel). In our initial pass, we compared three classification models, planning to select one and tune its hyperparameters further.

TABLE I
BASELINE CLASSIFIERS

Random Forest				
Label	**Precision**	**Recall**	**F1**	**Support**
Toxic	0.61	0.50	0.55	412
Non-Toxic	0.58	0.68	0.62	412
Avg/Total	0.59	0.59	0.59	824
Naive Bayes				
Label	**Precision**	**Recall**	**F1**	**Support**
Toxic	0.56	0.63	0.60	412
Non-Toxic	0.58	0.51	0.54	412
avg/total	0.57	0.57	0.57	824
SVM				
Label	**Precision**	**Recall**	**F1**	**Support**
Toxic	0.59	0.60	0.60	412
Non-Toxic	0.60	0.59	0.59	412
avg/total	0.59	0.59	0.59	824

SVM Tuning. SVM tied for the best performance among the classifiers we compared in our initial pass. It seemed the most promising since it is

TABLE II
SVM BASELINE CLASSIFIER TUNING

SVM, C=0.001				
Label	**Precision**	**Recall**	**F1**	**Support**
Toxic	0.65	0.38	0.48	412
Non-Toxic	0.56	0.79	0.66	412
avg/total	0.60	0.58	0.57	824
SVM, C=0.01				
Label	**Precision**	**Recall**	**F1**	**Support**
Toxic	0.66	0.33	0.44	412
Non-Toxic	0.55	0.83	0.66	412
avg/total	0.61	0.58	0.55	824
SVM, C=0.1				
Label	**Precision**	**Recall**	**F1**	**Support**
Toxic	0.60	0.59	0.59	412
Non-Toxic	0.60	0.61	0.60	412
avg/total	0.60	0.60	0.60	824
SVM, C=1				
Label	**Precision**	**Recall**	**F1**	**Support**
Toxic	0.62	0.55	0.58	412
Non-Toxic	0.59	0.66	0.62	412
avg/total	0.61	0.60	0.60	824
SVM, C=10				
Label	**Precision**	**Recall**	**F1**	**Support**
Toxic	0.65	0.62	0.64	412
Non-Toxic	0.64	0.66	0.65	412
avg/total	0.64	0.64	0.64	824

highly flexible: we could tune *C*, the regularization parameter that controls the tradeoff between a low training error and low test error; and *gamma*, which determines the influence of any single training example. When testing *C*, we held *gamma* constant at its default 'auto'; when testing *gamma*, we held *C* constant at its default 1.

TABLE III
SVM ITERATIVE CLASSIFIER TUNING

SVM Iterative Classifier				
	precision	**recall**	**f1-score**	**support**
Toxic	0.66	0.63	0.65	103
Non-Toxic	0.65	0.68	0.66	103
avg/total	0.66	0.66	0.66	206

TABLE IV
SVM BASELINE CLASSIFIER, TUNING GAMMA

SVM, gamma=0.001				
Label	**Precision**	**Recall**	**F1**	**Support**
Toxic	0.64	0.64	0.64	412
Non-Toxic	0.64	0.64	0.64	412
avg/total	0.64	0.64	0.64	824
SVM, gamma=0.01				
Label	**Precision**	**Recall**	**F1**	**Support**
Toxic	0.62	0.65	0.63	412
Non-Toxic	0.63	0.60	0.62	412
avg/total	0.63	0.63	0.63	824
SVM, gamma=0.1				
Label	**Precision**	**Recall**	**F1**	**Support**
Toxic	0.63	0.63	0.63	412
Non-Toxic	0.63	0.63	0.63	412
avg/total	0.63	0.63	0.63	824
SVM, gamma=1				
Label	**Precision**	**Recall**	**F1**	**Support**
Toxic	0.61	0.58	0.60	412
Non-Toxic	0.60	0.64	0.62	412
avg/total	0.61	0.61	0.61	824

Step 2: Iterative Classifier

In this step, we ran the iterative classification algorithm using the most successful baseline SVM we trained in step 1. We used the full baseline model to calculate predicted labels for the training set. On each iteration on the test set, we recalculate the dynamic attributes using at first the bootstrapped labels, and on subsequent iterations the result of the predicted model, repeating until convergence.

Fig. 10. Iterative SVM Feature Weights

Feature	**Weight**
Eigencentrality	3.901
Betweenness Centrality	1.681
Author Clustering Coefficient	1.091
Largest Strongly Connected Component	0.706
Average Author Degree	0.648
Average Thread Shortest Path	0.494
Thread Clustering Coefficient	0.285
PageRank	0.135
Average Thread Degree	-0.096
Average Comments	-0.205
Average Author Shortest Path	-0.497
Max Weakly Connected Component	-0.573
Subreddit Degree	-1.993
Harmonic Centrality	-1.993

VII. Discussion

In this paper, we explored classifying subreddits based on toxicity. In our first approach, we gathered static features of the subreddit themselves based up on the user-thread, thread-thread, and thread-user graphs as well as the attributes of the node in the subreddit-subreddit network. Using an SVM, we were able to obtain an f1-score of just under 0.65. Thus our approach had moderate success in predicting toxicity although there remains room for improvement. This result suggests that there is some correlation between structural network properties and toxicity, and shows promise for future work.

In our second approach, we supplemented our efforts with an iterative classification approach. Using this approach, we were able to obtain a slight improvement over our first approach with an f1-score of just over 0.65. In this approach, we tried to take advantage of the network property that many toxic subreddits had toxic neighbors. However, this approach saw a minimal gain in performance. We believe this result is because we did not guarantee an even proportion of toxic subreddits in the training and test sets. Additionally, in the worst case, the split could have partitioned the graph such that all neighbors of a node end up in the other partition. This partition would minimize the benefits of using neighbor features in the iterative model. This may explain why the iterative model did not make significant improvements to our static classifier. However, such a result may also suggest that network properties of the node itself based upon the different types of networks discussed are simply better approximators of toxicity than a subreddit's neighborhood. This seems unlikely, however. In future work, we would try to improve the split between train and test sets to better approximate the real-world distribution.

Figure 12 demonstrates which features were particularly useful in prediction. In particular, a subreddit's centrality in the subreddit network proved very important. Eigenvector centrality and betweenness centrality were the most influential features. This could suggest that our classifier began to correlate this level of influence with toxicity, which might imply that more toxic nodes occupy more influential positions in the network and are connected to more influential nodes. Or, more central nodes may have been "infected with" toxicity by their neighbors in the iterative steps.

VIII. Future Work

Many avenues remain for future exploration. While our approaches were successful, they could be improved. In particular, adding more dynamic features incorporating neighbor labels could further test the importance of a subreddit's neighbors, and decide whether an iterative model was actually a significant improvement over a relational classifier. Other types of centrality measures may also be helpful. Finally, it would be helpful to expand the dataset include more nodes in the subreddit-subreddit graph. Given the number of different graphs (thread-author, author-author, thread-thread, subreddit-user, subreddit-subreddit), this would take much longer to process that amount of data, but would be useful for painting a more complete picture of the Reddit universe.

Such an approach may also be useful in other social networks. One of the thorniest scaling issues for social networks is detecting content that violates their terms of service. Companies currently rely on manual flagging by users, followed by a human review, to detect this content. Automated network-based solutions would be a significant improvement over content-based scans. They would also be a useful tool for researchers studying the emergence of online political communities, even as a proxy for real-life communities.

IX. Conclusion

Reddit is a rich source for research into dialogue on social networks. Before this paper, little research had attempted to predict content characteristics based on network features. In this paper, we sought to predict subreddits' toxicity as a function of their network properties without regard to content. Our approach considered both the features of subreddit graphs, and the positions of those subreddits in the larger Reddit community. We used both static node attributes and dynamic neighborhood labels in an iterative classification approach. Our approach was moderately successful in predicting toxicity, but there remains room for improvement. We find a significant relationship between structural network properties and toxicity. More broadly, our work suggests that network properties can predict the content of online discussion.

References

[1] Reno v. ACLU - Brief of Appellees. "Reno V. ACLU - Brief Of Appellees." 2018. *American Civil Liberties Union.* Accessed Dec. 7 2018. https://www.aclu.org/legal-document/reno-v-aclu-brief-appellees

[2] "Packingham v. North Carolina." *Oyez*. Accessed Dec. 6, 2018. https://www.oyez.org/cases/2016/15-119

[3] LaFrance, Adrienne. "The Internet Is Mostly Bots." Jan. 31, 2017. *The Atlantic*. Accessed Dec. 7 2018. https://www.theatlantic.com/technology/archive/2017/01/bots-bots-bots/515043/

[4] Bowles, Nellie. "How Doxxing' Became A Mainstream Tool In The Culture Wars." Aug. 30, 2017. *The New York Times*. Accessed Dec. 7 2018. https://www.nytimes.com/2017/08/30/technology/doxxing-protests.html

[5] Saxena, Roheeni. 2017. "The Social Media Echo Chamber Is Real". *Ars Technica*. Accessed December 7 2018. https://arstechnica.com/science/2017/03/the-social-media-echo-chamber-is-real/

[6] Siegel A, Nikitin E, Barbera P, Sterling J, Pullen B, Bonneau R, Nagler J, Tucker J. "Measuring the prevalence of online hate speech, with an application to the 2016 U.S. election." 2018. *Stanford Canvas*. Accessed December 7 2018.

[7] Fan R, Xu K, Zhao J. "Higher contagion and weaker ties mean anger spreads faster than joy in social media." Aug. 12 2016. *Social and Information Networks.* Accessed December 7 2018. https://arxiv.org/abs/1608.03656v3

[8] Hauser, C. "Reddit Bans Incel' Group for Inciting Violence Against Women." Nov. 9 2017. *The New York Times.* Accessed Dec. 9 2018. https://www.nytimes.com/2017/11/09/technology/incels-reddit-banned.html

[9] Olson RS, Neal ZP. Navigating the massive world of reddit: using backbone networks to map user interests in social media. 2015. *PeerJ Computer Science*. Accessed Oct. 19 2018. https://doi.org/10.7717/peerj-cs.4.

[10] Gitari NG, Zuping Z, Damien H, Long J. "A Lexicon-based Approach for Hate Speech Detection." 2015. *International Journal of Multimedia and Ubiquitous Engineering.* Accessed Dec. 7 2018. https://preventviolentextremism.info/sites/default/files/A\%20Lexicon-Based\%20Approach\%20for\%20Hate\%20Speech\%20Detection.pdf

[11] Martin, Trevor. "Dissecting Trump's most Rabid Online Following." 2017. *538*. Accessed Nov. 8 2018. https://fivethirtyeight.com/features/dissecting-trumps-most-rabid-online-following/

[12] Hosseini H, Kannan S, Zhang B, Poovendran R. "Deceiving Google's Perspective API Built for Detecting Toxic Comments." Feb. 27 2018. *Machine Learning.* Accessed Dec. 11 2018. https://arxiv.org/abs/1702.08138

[13] Buntain C, Golbeck J. "Identifying social roles in reddit using network structure." Apr. 2014. *ACM Digital Library*. Accessed Nov. 8 2018. https://dl.acm.org/citation.cfm?id=2579231

[14] Hutchinson, Andrew. "Reddit Now Has As Many Users As Twitter, And Far Higher Engagement Rates". Apr. 20 2018. *Social Media Today.* Accessed Dec. 9 2018. https://www.socialmediatoday.com/news/reddit-now-has-as-many-users-as-twitter-and-far-higher-engagement-rates/521789/

[15] Neville J. "Iterative Classification in Relational Data." 2000. *Knowledge Discovery Laboratory, University of Massachussetts.* Accessed Dec. 6 2018. https://www.cs.purdue.edu/homes/neville/papers/neville-jensen-srl2000.pdf

[16] Bhagat S, Cormode G, Muthukrishnan S. "Node Classification in Social Networks." 17 Jan. 2011. *Social and Information Networks.* Accessed Dec. 7 2018. https://arxiv.org/abs/1101.3291

[17] "1.7 million Reddit comments." 2018. *Google BigQuery.* Accessed Nov. 8 2018. https://bigquery.cloud.google.com/table/fh-bigquery:reddit_comments.2015_05