# Gateway to understanding social attitude trends in subreddits

COSC 425 - Team 1
Joshua D. Dalton, Tristen Finley, Heather Haynie, Christopher Mobley
[1]Department of Electrical Engineering and Computer Science
University of Tennessee

## Abstract

Tensions are high throughout society, and overwhelming disagreement on many important topics is all too common. Claims of propaganda and ”fake news” are regularly thrown about. A possible contributing factor to these controversies is the concept of an echo chamber. An echo chamber is an online community which is created, ”as a result of selective exposure and ideological segregation” (Barberá et al. 2015). These aspects can have serious consequences when an echo chamber reinforces and maintains a false system of belief. Evidence of such can be seen in many cases of scientific persecution, such as Copernicus, Galileo, and Semmelweis to name a few. Given the potential negative outcomes of echo chambers, it would be highly beneficial to automate the identification the of these online communities. In response, the goal of this project is to take the first step towards the identification of echo-chamber-like behaviors on the social media platform Reddit.

## Introduction

The original goal of this project was to categorize subforums on Reddit (a.k.a subreddits) as either echo chambers or non-echo chambers. Limiting the domain to Reddit forums, it was first necessary to answer the question: what exactly constitutes an echo chamber? This question is not easily answered. The average person has neither the time nor ability to thoroughly consider every every aspect of modern society. As a general heuristic, a person must selectively filter out some ”noise” and get on with our lives. But where is the line between reasonably filtering out extraneous, unwanted information and maintaining an echo chamber? This difficulty is perhaps most dramatically highlighted in cases of mutual accusation (e.g., a user in r/liberal accusing r/conservative users of being in an echo chamber and vice versa. In this case, the deeper question is rooted in epistemology, the field of philosophy that distinguishes justified belief from opinion, what can be known, and how it is known. With this in mind, the project’s focus was shifted to a more approachable topic.

Without a precise definition of an echo chamber, subreddits could not be objectively be labeled as an echo chamber without imparting some team member bias. In addition, the topic-based structure of subreddits could lead to most, if not all, of the samples being labeled as an echo chamber. It is extremely challenging, if not impossible, to separate subreddits that simply try to keep their community bound to a specific topic and those that shun all dissenting opinions and are unwelcoming to new members. For these reasons, the project focus was shifted to a more direct question: can subreddits be separated from each other in a meaningful way, and if so, what attributes are most different between them? To explore this question, two subreddits with opposing premises: r/gatekeeping and r/gatesopencomeonin, were selected on the grounds that if any two subreddits can be differentiated between each other it would be these two.

The shift from actively seeking to identify echo chambers to attempting to classify gatekeeping and non-gatekeeping examples makes the immediate work more purely academic. After all, since the Reddit community is already performing the classifications – effectively a human hive mind that likely outperforms any readily available algorithm – the team’s approach does not extend knowledge regarding the classification of a posting. After all, all data was pulled from one or the other sub-forum, so its proper classification is already known. Instead, what this approach allows is to explore hypotheses about relationships between the data and the accurate classification of posts. It can be guessed, for example, that r/gatekeeping is guarding/protecting groups from certain perceived outsiders. This is likely to lead to the use of more negative language. Examples of negative language can then be used to train an algorithm on that assumption and test it against reality: the Reddit classification. If there is strong predictive value, that increases the confidence one might have in the hypothesis. By developing and testing multiple hypotheses over time, one can hopefully develop a deeper understanding of how even seemingly unrelated aspects of the data might be useful toward classifying posts on these forums. As those underlying hypotheses are established, tested, and shown to have useful predictive value, it is hoped that there might be discovered some ideas that aid in exploring the original problem of identifying echo chambers. With this in mind along with the new, more straightforward goal of testing hypotheses against reality, the team began brainstorming about how various features could be extracted and analyzed such that their predictive value allowed for accurate classification of posts from r/gatekeeping and r/gatesopencomeonin.

## Dataset

When creating this project’s dataset, the primary goal was to be as objective as possible and prevent team member bias from being incorporated into the collected data of the two subreddits. Using an automated web scraper removed the possibility of human error and bias leak and allowed for objective post and comment data collection. This allowed for a large amount of data to be collected and labeled in a timely manner. The data scraper was created using the PRAW (Python Reddit API Wrapper) library in Python. This library allows for easy access to post and subreddit information on Reddit. To remain as objective as possible, data was pulled from the top five-hundred posts of all time from each subreddit.

The features were chosen, when possible, based on statistical analysis. The features that were not chosen in this way were chosen due to the belief that the outcome be statistically based. For example, a feature of the dataset included a sentiment analysis score provided by the vaderSentiment Python library. Using a word cloud generator, a list of six words were identified to be in particularly high frequency in one subreddit vs. the other; so each of those words was tracked as a separate, individual feature. This process allowed for the words to be chosen on a statistical basis rather than creating bias by choosing words that seemed like they may be associated with a particular subreddit.

During implementation, the feature TotalKarma was thrown out, as it was determined to be unfairly biased towards the subreddit r/gatekeeping. It was proposed that the reason for this could potentially be due the collection method pulling from the top ranking posts of r/gatekeeping and r/gatesopencomeonin. Since r/gatekeeping is a much more popular sub, it can be concluded that the total karma in the top rank posts, from which the data was pulled, will always be the primary indicator of which subreddit a post could be classified under. However, this is not the sole purpose of the project's scope. The interest lies further in the attitudes of the users rather than overall popularity. See Table 1 for a complete list of our final features and their descriptions.

Table 1: Feature List and Descriptions

| Feature | Description |
|---|---|
| PercDeleted | The percentage of comments that have been deleted on a post. |
| AvgComKarma | An average of the karma (net up-votes) that each comment holds. |
| TotalKarma | The net value of upvotes on a post. |
| AvgVaderSentiment | A decimal value representing the sentiment of a post. Calculated from the average of the individual sentiments of each comment. Lower values represent a more negative sentiment. |
| ContainsCant | Average instances of the word “cant” or “ in each comment on a post. |
| ContainsDont | Average instances of the word “dont” in each comment on a post. |
| ContainsEveryone | Average instances of the word “everyone” in each comment on a post. |
| ContainsAnyone | Average instances of the word “anyone” in each comment on a post. |
| ContainsHappy | Average instances of the word “happy” in each comment on a post. |
| ContainsLove | Average instances of the word “love” in each comment on a post. |
| AvgComLen | The average length of each comment on the post, measured in words. |
| AvgWordLen | The average length of words used in the post comments, measured in character count. |
| AvgComTime | Average number of comments made on a post per minute, from when the post is submitted to when the last comment was made. |
| VoteTime | The number of votes on a post per minute. from when the post is submitted to when the last comment was made. |
| Post ID | Not a feature just an identifier for the post in case we need it for future data collection |

## Approach

Approaching the decision of possible classifiers, it was proposed to choose one model that would allow for a simplistic, straightforward approach, and a second model that was more complex and allowed for flexibility in the model. This resulted in the selection of a decision tree implementation, as well as a support vector machine (SVM) implementation.

The decision tree algorithm tested trees with splitting criteria of both gini and entropy as well as max node depths between two and fifteen, using the grid search method. For an extra layer of depth, this search was run three times, using the metric of either precision, recall, or F1 score to guide the training process and determine the best performing tree from the search. To visualize the best tree for each metric, node graphs of the trees were printed along with confusion matrices to visualize the performance itself.

Decision trees rely upon equations measuring node impurity to measure the most effective way to split data at each node. There are two main ways to measure node purity, either through information gain (based on the entropy equation which measures dispersion: $E(S) = \sum_{i=0}^{c} p_i \log_2 p_i$ where pi is the frequency of a particular target label) or the gini coefficient (a measurement of the homogeneity of a node : $G = \frac{A}{A+B}$, where A is the area above the Lorenz curve and B is the area below it, with the curve itself representing inequality between the target labels). These equations are implemented via the peer-reviewed, open-source algorithms built into the Scikit Learn library. (Note: The algorithms are assumed to be reliable and accurate, but a detailed analysis/discussion is beyond the scope of this paper.) In all three searches, entropy was selected as the best performing splitting criteria. While gini leads to significantly better performance in trees with smaller tree depths, as depth increases past eight entropy provides slightly better performance. It is unclear from what this pattern originates, as generally these two equations both measure node impurity and should lead to trees with similar performance. As a small note, no threshold was put on either of these values to stop tree generation. Since the algorithm tests trees of every possible depth, trees that overfit the data, as a result of nodes that are too pure, won't be selected because of their performance drop.

The data for the support vector machine (SVM) was split into three subsets: training set, validation set, and test set. These subsets were generated using sklearn's train_test_split method. By this method, samples were randomly selected from the data set until the new subset was 70% of the original data set size. This new subset became the training set. The remaining 30% of the original data set was split using the same process into another two subsets, the validation set and the test set. The validation and test sets consist of 15% each of the original data set.

Searching for the best model to fit the data, both a coarse and fine grain search was conducted. This process exhaustively compared the linear, polynomial, and radial basis function (RBF) kernels with potential ranges of corresponding hyperparameters for the SVM. The function for the linear kernel can be written as $K(x, y) = x^T y + c$. The equation for the polynomial kernel can be written as $K(x, y) = (ax^T y + c)^d$, d2. It should be noted that, for the polynomial kernel, the degree value (d) must be greater than or equal to two since a degree value of one is equivalent to the aforementioned linear equation. The RBF kernel equation is written as $K(x, y) = exp(-\gamma\|x - y\|^2)$. While all three of the kernels have the hyperparameter, c (or soft-margin constant), the polynomial and RBF kernels each have a second hyperparameter, degree and gamma, respectively. The values of the hyperparameters were generated using the numpy library functions, logspace, for soft-margin and gamma, and linspace, for degree. Logspace generated an array of eleven evenly spaced values between 2-5and 215 for the soft-margin constant, and 2-15 and 25 for gamma. Similarly, linspace generated an array of eleven evenly spaced values between 2 and 12 for the degree hyperparameter.

The results of the coarse grid search found that the best model used the RBF kernel with hyperparameter values c = 32.0 and gamma = 0.03125–this corresponds to 25 for c and 2-5 for gamma. Observing the classification report, the model achieved a higher precision score for r/gatesopencomeonin, and higher recall and f1 scores for r/gatekeeping samples. This model yielded an overall accuracy of 84%. After initial estimates for a best fitting model for the data set was found from the coarse grid search, the hyperparameter values needed to be tuned further. To accomplish this, a fine grid search using 10-fold validation was executed to find more exact values for optimal c and gamma. The fine grid search works similarly to the coarse grid search. Since the RBF kernel was found to be the most optimal, this was the only kernel used for the fine grid search. The hyperparameters for the model were improved by narrowing the scope of values to 20 to 28 for the soft-margin constant, and 2-10 to 2-3 for gamma. Additionally, the number of instances within the ranges increased to allow for 33 values and 29 values, respectively, to be tested.

The result of the fine grid search found that the best parameter for the RBF model to have a c value of 19.027, or 24.25, and a gamma value of 0.026, or 2-5.25. Similar to the initial, coarse model, the model with proposed better hyperparameters resulted in higher scores for recall and f1 for r/gatekeeping, while the score for precision was higher for r/gatesopencomeonin. Differently, however, the overall accuracy of the model yielded an accuracy of 82.67

## Evaluation

For tree depth performance, the model saw a steady increase up until depth nine, where performance starts to fluctuate. The best depth for this dataset tends to be between ten and thirteen depending on the metric used in training the tree. After observing the confusion matrices produced from each scoring metric, it was noticed that there was very little variance in both precision and recall, suggesting that with a larger sample of data the metric wouldn't matter for selecting the best tree. However, this process does give some insight into which features are most important to the classification process. While the trees did vary between metrics, the first three node layers remained the same (see figure 1), with features vaderSentiment and CommUpRatio dictating the first two splits. Although the specifics of how the sentiment of a comment was calculated was left up to the vaderSentiment library, it does seem to be a good predictor of which subreddit a post belongs to, with r/gatekeeping holding a more negative sentiment among its comments. Along with r/gatekeeping tending to have a lower average comment karma, it is clear that it can be classified as a much more controversial subreddit in general, which makes sense considering the subreddit is based upon the idea of looking at/making fun of people on the internet who are trying to gatekeep (the act of shunning a particular subset of people from a group based on arbitrary qualities). It is also worth noting that the individual word features had no appearance in the top layers of any of the trees, suggesting that the frequency of a particular word did not provide a definitive split between the data. These features are likely not suited to a decision tree and would need to be analyzed through another form of classifier to be useful.

Evaluating the proposed best fit SVM model was to run the model with the untouched test set. Observing the final model results, the model reported an overall accuracy of 86.67%, outperforming the predicted accuracy from both the model found with the coarse grid search as well as the initial accuracy from the fine grid search. Consistent with the previous results, the fine-tuned model scores higher for precision with regard to r/gatesopencomeonin but scores higher for recall and f1 scores when considering r/gatekeeping. The results of this model can further be seen in the confusion matrix, in figure 2. From the confusion matrix, it can be visually seen how the model yields a very high number of correctly predicted samples.

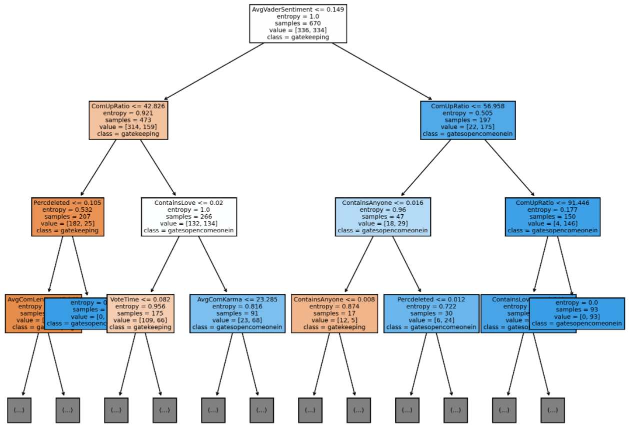

Figure 1: Visualization of the constructed decision tree. Nodes of the decision tree varies slightly between Precision, Recall, and F1; however the trees are the same for at least the first three layers shown.

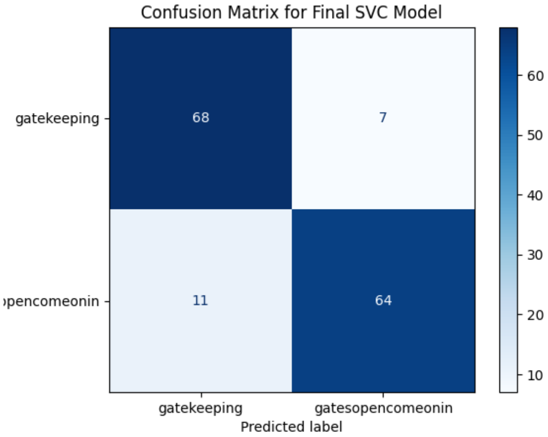

Figure 2: F1 score confusion matrix for decision tree implementation. Figure shows 142-True Negative, 22-False Positive, 45-False Negative, 121-True Positive.

Assessing the functionality of the SVM algorithm overall as applied to the dataset, it appears that an SVM algorithm is a good predictor for this task. This proof cannot only be seen in the final model with an accuracy around 86%, but also in the other kernels tested, as seen with the linear kernel predicting an average score around 84 and the polynomial contour plot showing similar results for c values between 10-1 to 100 and degree values between 6 to 9. It can be noted that the accuracy of the best fit model drops after executing the fine grid search but performs much higher when the same mode is run with the test set. This can potentially be attributed to overfitting of the model to the training data. The SVM further hints that the data set is linearly separable since the best performing model yielded a gamma value close to 0 and a soft-margin constant that renders a comparatively wide margin of separation.

## Future Work

Analysis of our algorithm's results prove a high degree of separability between subreddits, with the overall sentiment being the decisive factor in classification. Although we were able to separate these two communities with a high degree of accuracy, this may not extend to a wider use case. Even in cases where two subreddits have where members have opposing political or philosophical views might not have the same degree of separation as those with opposing premises. More research into how the more separable features, such as overall sentiment, vary across a diverse range of subreddits would be necessary before any categories or conclusions can be made about them solely on statistics.These statistics could also be applied to this project's original goal of identifying echo chambers in online communities. Although we had issues identifying ways to accurately define or measure an echo chamber, community sentiment and moderation statistics are likely to play a large role in that classification especially if they can be shown to have a significant degree of variability across the Reddit platform.

Another area of interest lies in the moods created by each subreddit's premise, that likely results in the separability we see. On one hand, r/gatekeeping is built on making fun of and shining a light on individuals that are themselves trying to exclude others. On the other hand, r/gatesopencomeonin shows examples of people being welcoming and inclusive and further promotes that positivity. Rather than focus on the ability to classify subreddits, future studies could research how statistics like a community's sentiment affects its need for moderation, or even the overall moods of people who frequent one vs another.

Caution should be taken when attempting to label or classify any community based on statistics alone. While this study looked only at separability, potential ethical concerns could arise if any of these statistics were used to single out communities. For example, the overall sentiment of subreddits could be used to target moderation at those with a more negative sentiment, however this could be construed as censorship if the targeted communities happened to share a political, social, or economic background. Of course there is also the concern of when moderation is necessary at all in an online space, and to what extent. These concerns are slightly outside the scope of our project however, so it will be up to future research to grapple with these issues.

## Contributions

Joshua D. Dalton - Contributed to underlying philosophical considerations and project design; assisted with writing, editing, and proofreading; streamlined and explained philosophical evolution of project for final report introduction; provided audio/video presentation and corresponding slide for relevant portion of the final presentation

Tristen Finley - Contributed to all milestone reports to varying degrees; Wrote data collection algorithm; Wrote decision tree algorithm; organized group meetings and workflow; Aided in feature generation; Presented dataset portion of presentation

Heather Haynie - Contributed to writing major sections of milestone reports; Reviewed algorithm implementations; Performed evaluation of SVM code; Managed group documents; Collected data results of final models.

Chris Mobley - Contributed to team project discussions; Assisted in writing, proofreading, and formatting of milestone reports; Wrote Support Vector Classification model implementation and tests; Aided in feature generation