Applying Document Clustering to Reddit’s Hobby Communities

Jonathan Wiseman

Elizabethtown College

(717)857-4069

wisemanj@etown.edu

**ABSTRACT**

Document clustering is an application of cluster analysis that seeks to group text collections such that documents in the same group are more similar to each other than to documents in other groups. Text data from publicly available Reddit submissions, comments, self-posts, and popular products can be used to produce document vectors for hobby subreddits, which in turn can be used to make personalized recommendations for active users that have defined interests. Additionally, subreddits can be visually represented based on their textual content, enabling a holistic view of Reddit’s communities.

# INTRODUCTION

Reddit is a popular news aggregation and content hosting website that allows users to form directed communities, called subreddits, for discussion of relevant topics. The fifth most visited site in the United States, Reddit boasts upwards of 300 million unique users. Users can share content in a variety of ways: uploading of original content (including images and videos), links to content hosted elsewhere (such as news articles), and by making text-only “self-posts;” users are also able to comment on content. A positive-negative voting system, collectively referred to as karma and accumulated through “upvoting” and “downvoting” content, gives explicit feedback on comments and posts.

Cluster analysis is an unsupervised learning technique that seeks to group a set of objects together such that objects in the same group are more similar to each other than to objects in other groups. Document clustering is a problem in cluster analysis that seeks to cluster together “documents” based on their natural language content. In this context, documents may be web pages, books, news articles, etc. Document clustering is being used for a variety of tasks in information retrieval, including search engine optimization, topic extraction, fast information retrieval, and resolving lexical ambiguity [1].

Document clustering often relies on the representation of documents as word vectors; this is known as the bag-of-words approach. Bag-of-words entails representing each document as a sparse matrix consisting of binary attributes (with features being words: a 0 denotes the absence of a word, a 1 indicates the presence of a word) for each word in the collection’s vocabulary. To reduce the size of the vocabulary, stop words (common words like: a, the, is, etc.) are removed and some textual preprocessing (like lemmatization or stemming, both of which seek to get the root form of words) are performed [2]. Next, a measure of similarity or distance (called the metric) between word vectors must be chosen. Examples include Euclidean distance, cosine similarity, the Jaccard coefficient, Pearson’s correlation coefficient, and averaged Kullback-Leibler Divergence [2]. With document vectors and a sufficient similarity metric, a clustering algorithm must be chosen. Examples include k-means (in which documents are separated into k groups based on distance from a group’s centroid) and hierarchical clustering methods. Additionally, other preprocessing can be performed to increase model performance, such as feature selection (further selection of important words), word weighting (ex. using tf-idf weights), and standardizing of vector lengths to 1 to avoid long documents dominating the analysis [2].

There are existing applications of cluster analysis on data from Reddit. In [3], subreddits are clustered according to user overlap: subreddits that contain comments from a single user are more similar to subreddits in which that user does not comment. This is an approach more akin to collaborative filtering: subreddit similarity is being determined by implicit user ratings in the form of activity. This reveals only similarity in user base, not in actual content.

[4] is a site that contains useful data for every existing subreddit, including size, activity, top posts, top users, keywords, and similar subreddits both by shared users and by shared keywords. The shared keywords aspect is very interesting, but it is not used to find similar keywords in a very convincing manner. For one, finding subreddits by shared keywords does not work well for general subreddits with no defined topics like r/AskReddit, r/funny, or r/pics. Rather than simply finding subreddits with shared keywords, it will be interesting to find similar subreddits by all language used.

A better solution for cluster analysis of Reddit’s communities is to narrow the subreddit range and establish ground-truth labels of topic. A good starting place is hobby subreddits that can have good ground-truth labels, such as Entertainment, Arts, Sports, Humor, etc. A good staring list is available at [5]. With ground-truth labels, it will be possible to assess a clustering model and help guide selection of hyperparameters like k (for a k-means algorithm). Secondly, a better solution must include text from a variety of sources in addition to comment data: subreddit descriptions, post titles, rules, and text-based self-posts should be included to more completely represent the content of a subreddit. This will also help to highlight vocabulary that is unique to each subreddit. Finally, information about the purchasing habits of each subreddit can provide further detail that is not at once obvious or clear from other text data.

The goal of this project is to produce a clustering model that can be used to make a recommendation system for users based on subreddits they enjoy. This system will both recommend subreddits like the ones the user already enjoys and recommend text posts that are related to ones they have already viewed. This implies two models: one for clustering subreddits, and a further one for clustering text posts.

# DESIGN

The existing “Things on Reddit” dataset in [6] will be used to gather purchasing information about each subreddit. It contains data about the top products of each subreddit, listing product name, category, and an Amazon link. Due to difficulty in scraping Amazon pages, price and other information available via the links will not be used. However, product names and categories will be counted in the word vectors produced for each subreddit.

Secondly, comment data, subreddit descriptions, post titles, rules, and text-based self-posts will be scraped from subreddits via Reddit’s Python API (documentation is available at [7]). The standard preprocessing steps will be applied to the text data. Additionally, terms will be weighted according to tf-idf, and I may do some feature selection based on the top terms (in order to speed up model application and reduce complexity). Additionally, self-posts will be turned into their own word vectors.

To start, I will cluster hobby subreddits for which ground-truth labels can be easily established. Depending on the time required to produce clustering models for these subreddits takes, I may expand to other subreddits. In general, I will avoid the large, general subreddits that have no clear topic. In all, the major theme of this project is about hobbies and interests, and how to find similar communities to the ones a user is already interested in.

# EXPERIMENTS

Under most circumstances, the unsupervised nature of cluster analysis obscures the nature of a model’s performance. However, with the manually generated ground-truth labels it will be possible to see how well models cluster subreddits that should theoretically be similar. However, the insights generated from the cluster analysis will be valuable, even if the groups formed don’t correspond to the manually generated labels. The major focus will be on finding similar subreddits and posts that will be useful or interesting to a given user.

I’m not the most familiar with cluster analysis now. However, I do know that k-means is a popular and effective way to perform document clustering [8]. This will be the first model I make, and I will move on to any others that I research and find useful. I plan on using cosine similarity (since it lends itself naturally to document vectors) initially, but I am also interested in how changing the similarity metric will impact the insights generated by the models.

Finally, I also want to be able to generate a visual representation of subreddits. To do this, I will need to perform dimensionality reduction so that document vectors can be plotted (either two- or three-dimensionally). This plot will be useful as a visual representation of subreddit similarities.

# REFERENCES

1. A. Di Marco and R. Navigli, “Clustering and Diversifying Web Search Results with Graph-Based Word Sense Induction,” *Computational Linguistics*, vol 39, no. 3, August 6, 2013. [Online]. Available: <https://www.mitpressjournals.org/doi/10.1162/COLI_a_00148>. [Accessed September 30, 2019]
2. A. Huang, “Similarity Measures for Text Document Clustering,” *Proceedings of the 6th New Zealand Computer Science Research Student Conference*, January 1, 2008. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.402.4633&rep=rep1&type=pdf>. [Accessed September 26, 2019]
3. T. Martin, “Interactive Map of Reddit and Subreddit Similarity Calculator,” *shorttails.io*, November 28, 2016. [Online]. Available: <https://www.shorttails.io/interactive-map-of-reddit-and-subreddit-similarity-calculator/>. [Accessed September 30, 2019]
4. “Subreddit Stats,” *subredditstats.com*. [Online]. Available: <https://subredditstats.com/>. [Accessed September 26, 2019]
5. u/jb491000, “List of hobbies, with links to their sub reddit,” *Reddit Inc.,* 2016. [Online]. Available: <https://www.reddit.com/r/Hobbies/comments/3yab11/list_of_hobbies_with_links_to_their_sub_reddit/>. [Accessed September 26, 2019]
6. A. Bilogur, “Things on Reddit,” *Kaggle*, 2015. [Online]. Available: <https://www.kaggle.com/residentmario/things-on-reddit>. [Accessed September 9, 2019]
7. B. Boe, “PRAW”, 2017. [Online]. Available: <https://praw.readthedocs.io/en/latest/index.html>. [Accessed September 26, 2019]
8. M. Steinbach, G. Karypis, and V. Kumar, “A Comparison of Document Clustering Techniques,” *KDD Workshop on Text Mining,* 2000. [Online]. Available: <http://www.cs.cmu.edu/~dunja/KDDpapers/Steinbach_IR.pdf>. [Accessed September 26, 2019]