SIADS 696 Milestone II Project Report

Clustering Reddit Threads to Analyze Correlation Across Stock Market

Stephen Moilanen, Carl Debski & Kento Oigawa

Introduction

Our project seeks to understand if we can identify clusters within finance-related social media forums like Reddit and use them to discover any significant relationship with stock market performance. By solving this, we can link activities within social media forums to market activities. In terms of the motivation for studying this topic, it was primarily driven by the fact that both finance and social media play important roles in determining our quality of life. As the inflation rate is still very high, any lack of financial literacy can lead to lowering of net assets. However, with record high valuations in not only stocks but also in meme stocks and non-traditional assets such as Non-Fungible Tokens (NFT) and cryptocurrencies on the rise, further fueled by commission-free online trading platforms such as Robinhood and gamification of the investing process, we have more choice than ever in terms of where and how to invest. As many of the younger, new entrants to the market are familiar with and active on forums within Reddit, we naturally became interested in knowing whether we can establish meaningful relationship between the market performance and activities on the forums. The approach used for this project was combination of two unsupervised machine learning methods, in the form of Latent Dirichlet Allocation (LDA) and Clustering algorithms, and a supervised machine learning method that takes input into the form of times series, created out of forum topics and forum communities outputted respectively from the two unsupervised models, and then performs a regression on the combined data. This is different from other, related projects in the sense that we are using a combination of topics and communities, so going beyond positive/negative sentiment, as well as benchmarking our performance against a single security. Moreover, at least for the dataset in question, our choice of the models as well as the approach chosen to fine-tune the models are clearly distinct. In terms of the outcome, the main findings of the project were that distinct, interpretable communities and topics could be identified within the activity on Reddit using PCA and LDA, respectively. Using those labels as features in a Support Vector Regression model generated a low MSE on training data. However, the model did not generalize well to the test data, in part due to the short term volatility of the period examined, followed by a longer period of low activity. For the coding components of this project, please refer to Appendix Section B.

Related Work

1. Stock Sentiment Analysis using News Headlines by Siddarth Tyagi, June 2021

This was a Kaggle based notebook project which sought to use sentiment analysis using new headlines, where the sentiment was either bullish and bearish and the stock, to determine the stock price, defined by binary value of 0s and 1s to indicate buy/sell stocks. Our project instead does not define sentiment but instead performs topic and community discovery in the forum and also performs regression on a numerical variable, the price, instead of classification as in this project.

2. Introduction to clustering-based customer segmentation by Kaixin Wang, November 2023

A study by Microsoft Data Scientist on performing K-means clustering and elbow method to identify customer segmentation. It talks about how you can use recency, frequency and monetary as three dimensions on which you can segment customers. This was inspiring in the sense that Reddit users can also similarly be segmented by factors such as high rating, level of engagement and level of controversy.

However, unlike with our project, it did not aim to use customer segmentation results to make predictions or perform forecasting.

3. Stock Prediction GAN + Twitter Sentiment Analysis, December 2022

This project attempts to utilize a particular technique (Generative Adversarial Networks - GANs) to forecast the price of Amazon shares given both historical financial data and social media data from Twitter. As such, a technical indicator of price changes, like moving average, and a sentiment score (positive, neutral, negative) are used as features to predict share price. Our approach focuses on meme stocks, like GameStop, that do not have solid financial fundamentals. Instead price movements are considerably more volatile and heavily influenced by social media. As such, we exclude the technical financials and instead focus deeper on social media, identifying the communities and discussion topics and their relationship to stock price movement.

Our project is not an extension of Milestone 1 or other previous course projects.

Data Source

This project used two different data sources: a Kaggle dataset that contains information about user posts based on various finance-related forums on Reddit, and financial data available on Yahoo Finance which was obtained using APIs built into YFinance libraries.

Kaggle dataset on finance-related Reddit forums

Location: Available on Kaggle (link)
Format: CSV files stored in folders (e.g., /finance/submissions_reddit.csv)
Important variables contained:
Number of records retrieved: 273,327 rows with 24 columns
Time period: 2021-01-01 to 2021-12-31
Initial pre-processing: Preprocessing steps differed slightly depending on the model. For the LDA model, only the main body of text was required for analysis and thus was isolated from the rest of the dataset. Preprocessing consisted of dropping instances that did not contain text, had been deleted, or had been removed by Reddit moderators. This reduced the total number of usable posts from 273,327 to 94,039. For the user community detection that used clustering techniques, preprocessing steps included dropping non-numeric columns such as posts, title, etc and then filtering out posts which belonged to users who had been deleted. Afterwards, in order to allow for testing of the model to ensure its generalizability, the data has been split into train and test sets using 5:7 split where data from 2021-01-01 to 2021-05-30 is used to train the model and data from 2021-05-31 to 2021-12-31 is used to test the model.

Stock price data on Yahoo Finance retrieved via API in YFinance library

Location: Available through yfinance python package (link)
Format: API (original), Pandas DataFrame
Important variables contained: Price of GME (Gamestop) stock
Number of records retrieved: 251 rows and 2 columns
Time period: 2021-01-01 to 2021-12-31
Initial pre-processing: Primary preprocessing involves filling in all dates with the current share price, including weekends and holidays when the market is closed.

Feature Engineering

Feature Engineering for the LDA model: Only the ‘selftext’ field was utilized in the topic modeling using LDA, and therefore was isolated from the rest of the dataset. Preprocessing (performed as part of feature engineering) included tokening each post, removing tokens that were punctuation or included in the stopword list, lowercasing all letters in each token, and lemmatizing tokens to combine equal words with differing conjugations. Lemmatization was performed using the NLTK library, while other preprocessing was performed using built-in components of the CountVectorizer class of the Scikit-Learn library.

Even after preprocessing, not all tokens were utilized by the model, and some tokens were combined into bigrams and trigrams. Parameter tuning was conducted to determine the best number of features to include and what level of n-gram should be included. Perplexity and coherence were chosen as performance measures to find the best parameters for feature development. Since perplexity measures require unseen data, parameter tuning utilized cross validation. Final features included the most frequent 10,000 unigrams, bigrams, and trigrams. To be included, each feature had to be present in at least 25 documents but not present in more than 90% of documents.

Feature Engineering for K-Means/Agglomerative Clustering model: In order to ensure that meaningfully distinct user communities could be found using non-textual data from Reddit forum posts, a series of feature engineering steps were needed before feeding the data into the clustering model. First, because the GME Reddit forum data is on post level, these posts have to be aggregated to the level of users. To allow this, numerical fields are all averaged, which was determined to be appropriate based on what each of the fields entails. Additionally, one additional field called ‘num_post’ was added to indicate number of posts that the user has made, and this field was used to exclude users who have made 3 or less posts, as these users have insufficient postings to determine communities and introduce additional noise. Afterwards, all the values are standardized and then dimensionality reduction was applied as there were too many features to discern user communities in the original form. In terms of dimensionality reduction, T-Distributed Stochastic Neighbor Embedding (T-SNE), Principal Component Analysis (PCA), Uniform Manifold and Projection (UMAP) and Multidimensional Scaling (MDS) were all performed with the dimension of 2 in order to visually assess if there were any interesting visible structures. While T-SNE looked the most visually promising, the issue with T-SNE is that it should not be used for clustering and is also unable to transform based on a test dataset and fit to a new, unseen dataset, ruling it out as an option since we don’t want to introduce data leakage during the testing phase. Moreover, MDS and UMAP, while good potential dimension reduction, require significant computation time using a high performance computing (HPC) environment which is undesirable for practicality reasons. This naturally left PCA, at higher dimension, as a viable candidate, and its result can be viewed after reducing it to two dimensions using T-SNE to visually confirm the goodness of clustering.

The communities and topics provided by the unsupervised learning models form the basis for the features used in the supervised learning models. Daily counts of labeled reddit posts for communities, topics, and combinations (community discussing topic) become the features. To account for potential lag in the social media activity and market price changes, features are also generated by shifting the time period a given number of days. All features are then ranked based on their correlation with the stock price, the top ten of which are selected to be used in the supervised modeling.

Unsupervised Machine Learning

Methods Description

Unsupervised learning consisted of two portions: developing topics from the text contained in the Reddit posts, and determining user communities contained within the broader dataset. These tasks were

performed separately and each was intended to provide new features to the dataset for use in the supervised learning portion of the project.

LDA was chosen for the topic modeling due to its interpretability and explainability as a probabilistic method. Features were developed by tokenizing the texts from each Reddit post, and determining the 10,000 most frequent words or word groups used. The tokens were manually spot-checked to determine their quality, and after realizing that many words were being double or triple counted due to pluralisms or various verb conjugations, lemmatization was incorporated into the preprocessing. While most preprocessing steps are incorporated directly into the vectorizer function, the lemmatization was performed using a different library prior to vectorization so that the lemmatized results could be more thoroughly inspected. Each post was tokenized, lemmatized by noun, verb, adverb, and adjective, and then put back together into a full text for input into the vectorizer. Some tokens were questionable, such as numbers and tokens like “www” and “https”, which hold inherent information but could contribute noise to the results. These tokens were initially kept so results could be explored and fine-tuned later if deemed appropriate, however some, such as “www”, “http”, and “com”, were later removed before final topics were provided .

Since manually exploring results of an LDA model requires evaluating large amounts of text data, loops were written to test various combinations of topic counts, maximum feature limits, and n-gram level. Perplexity was one of the two performance measures used, and requires unseen data to be calculated - cross validation was used in the parameter tuning for this reason. After running and plotting 106 trials, it was decided to focus on using 10 topics, 10,000 features, and word combinations up to tri-grams. This decision was primarily based on the second performance metric used - coherence - which seemed to cease accelerating around 10 topics. Although perplexity was considered, and minima for perplexity could be observed for differing parameter combinations, perplexity did not change a great deal with regard to topic number. Furthermore, since total token count is considered in the calculation of perplexity, comparisons between trials using different max token counts was difficult since they could not be made relative to each other. The selected parameters were then utilized in a vectorizer and LDA model without using cross-validation. Results of this model were inspected by viewing the top 10 words of each topic, the coherence measure of each topic, as well as reviewing the actual top ten texts of each topic to determine how closely related they were. Verbal descriptions of these topics were then created and shared for use in the supervised portion of the project.

In addition to the LDA used to identify topics within the Gamestop (GME) forum, unsupervised, clustering methods were explored to identify distinct user communities that may exist within the forum. The unsupervised learning workflow would be that we would feed dimensionally reduced, historical data, as in belonging to posts which were made prior to the stock price date, into a model with the optimal number of clusters that we have chosen based on performance during evaluation and manual inspection. In terms of the models to be used for this task, we considered Agglomerative Clustering, K-Means Clustering, and Density-Based Spatial Clustering of Applications (DBSCAN). First of all, we considered whether to use K-means or DBSCAN as a base clustering method. Inspecting the dimensionally reduced data in a two dimensional plot, we observed that there is no significant amount of outliers and lack of complexity in structure, so there might not be significant advantage of using DBSCAN over K-Means, especially given it’s computationally heavier. There were also several practical reasons why K-Means is preferable such as interpretability and scalability. When it comes to determining the appropriate number of clusters, it is much easier to adjust K value in K-Means rather than EPS and min_samples hyperparameters, especially considering that the groupings will be manually inspected. Moreover, given that we want to create a pipeline on which to determine community for large volumes of new, potentially unseen users, it seemed preferable to have a model that was deterministic and able to predict new data based on existing clusters rather than recompute the entire cluster. This was the reason why Agglomerative clustering was also not chosen as a base model.

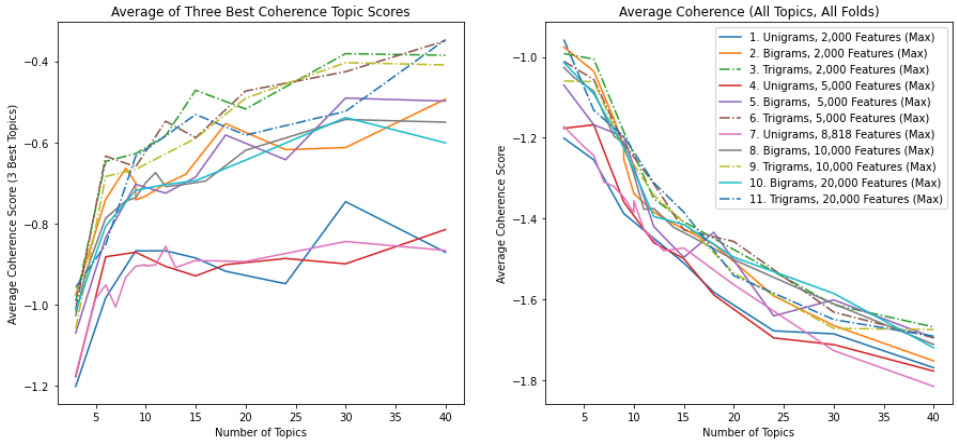
Once we have settled on using K-Means for the above-mentioned reasons, we wanted to explore if combining with Agglomerative clustering would be additive if there is any underlying tree-based structure to the users. As a result, we explored performance of models that combine K-Means with centroids based on Agglomerative clustering to see whether it performs standalone K-Means. In terms of how the performance was evaluated, measures such as Within-Cluster Sum of Squares (WCSS), Davies Bouldin Scores (DBS), Calinkski Harabasz Score (CHS), Silhouette Scores and Primacy Ratio (PR) (not an official measure, with its name inspired by urban primacy - it is a function created to return proportion of the largest cluster to penalize clusters for having a large, singular cluster) across different combination of dimension for PCA and number of clusters. First, the best dimension was chosen by grouping by dimension and looking at the averages for the aforementioned measures. Then, once the dimension was chosen for PCA, we assessed measures for different numbers of clusters and picked the optimal number K, where the Elbow Method was adopted to choose K at which we see diminished returns from increasing the number of K. Moreover, we have also performed hyperparameter tuning for “tol” and “n_init”. Given that for unsupervised methods evaluation and hyperparameter tuning are done at the same time, please see the “Unsupervised Evaluations" section for more details on this result.

Unsupervised Evaluations

For LDA modeling, evaluation includes quantitative measures of perplexity and coherence, as well as qualitative evaluation of visual inspection of results. Perplexity and coherence were chosen because they are measures of model quality - they provide a probabilistic measure for how well the documents fit the model (perplexity) and how interpretable each topic is (coherence). They were calculated first to narrow down the range of available parameters, and were based on varying values of topic count, maximum number of features, and level of n-gram. Families of models were evaluated on matching max number of features and level of n-gram, and models within each family were plotted based on number of chosen topics.

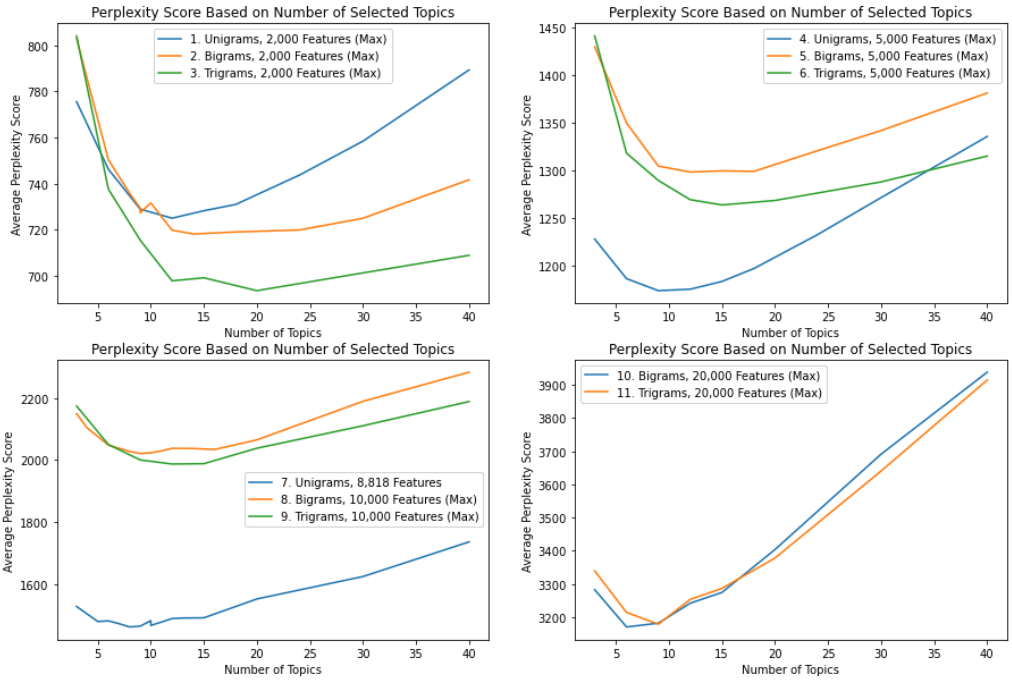
Trigrams showed the best overall coherence scores (trigrams are dashed in Plot 1.1 and Plot 1.2 below for clarity). While the overall coherence of all topics decreased as the topic count increased (see Plot 1.2, the coherence of the top three topics increased as topic count decreased (see Plot 1.1, suggesting that as topic count increases the lower-ranking topics become less interpretable. The rate at which the best three topics improved coherence scores started to lessen at around 10 topics, which was part of the decision to use 10 topics for the manual inspection.

Plot 1.1 (L), Plot 1.2 (R)



Perplexity generally appeared to be minimized between 5 and 15 topics, but varied by model family. The rate of change of the perplexity scores for each family was not considered substantial, which made pinpointing a best topic count difficult given the probabilistic nature of the modeling. Perplexity between families with different maximum feature counts was not considered comparable since total number of tokens is a variable used in the calculation of perplexity. Plot 1.3 can be viewed below:

Plot 1.3



While choosing parameters for the primary model, coherence was the main focus as it is a better indicator of interpretability of model topics. Perplexity was helpful in determining the best number of topics, but since the rate of change of perplexity was considered small in most model families, it was deemed less important.

The observations from the perplexity and coherence results were neither stark nor congruent; for this reason, the selection of 'best' model parameters was partly a judgment call, and not stiffly defined by tuning results. The selection of trigram level of word combinations was chosen along with 10 topics and 10,000 maximum features as the best combination of coherence, perplexity, and practicality of manually investigating topics (a lot more effort would be required to evaluate 40 topics versus 10). See Table 1.1 for a summary of each model family with the lowest perplexity score.

Table 1.1

Maximum Features Allowed	Largest N-Gram	Number of Topics	Actual Features Included	Avg. Perplexity	Avg. Coherence	Avg. Coherence - Top 3 Topics	Avg. Coherence - All Remaining Topics (Besides Top 3)
2000.0	(1, 1)	12.0	2000.0	724.961994	-1.448782	-0.866885	-1.642748
2000.0	(1, 2)	14.0	2000.0	718.093745	-1.413202	-0.678807	-1.613491
2000.0	(1, 3)	20.0	2000.0	693.474479	-1.476754	-0.517521	-1.646030
5000.0	(1, 1)	9.0	5000.0	1173.642879	-1.358999	-0.870413	-1.603292
5000.0	(1, 2)	12.0	5000.0	1298.052374	-1.419821	-0.724915	-1.651457
5000.0	(1, 3)	15.0	5000.0	1263.523170	-1.428041	-0.588133	-1.638018
10000.0	(1, 1)	8.0	8818.0	1462.675878	-1.320700	-0.932260	-1.553764
10000.0	(1, 2)	9.0	10000.0	2021.509591	-1.214125	-0.736156	-1.453109
10000.0	(1, 3)	12.0	10000.0	1987.445694	-1.342035	-0.627803	-1.580112
20000.0	(1, 2)	6.0	20000.0	3169.989209	-1.091240	-0.806299	-1.376180
20000.0	(1, 3)	9.0	20000.0	3178.226319	-1.196635	-0.633914	-1.477996

For the initial round of manual inspection, all the available data was utilized for model training. All ten topics were interpretable, and were determined by looking at the titles, most common words, and text of the ten posts with the highest probability of being assigned that specific topic. The text was found to be more helpful in determining context than the titles and top words. The probability of the top ten posts belonging to its assigned topic was consistently over 98%. That said, the model results weren't without noise. Some of the top posts were shown to be nonsensical, containing just a few words repeated over and over again. In addition, some topics were not chosen as expected. For example, one of the initial topics was posted memes/photos (matching on tokens such as 'png'), which technically isn't really a contextual topic. The following round of manual inspection was performed the same way, but split the data into a train and test set. Topics were developed using only the train data, which was done to prevent data leakage in the supervised portion of the project. The entire dataset encompassed a year's worth of Reddit posts - the training set included the first 5 months of data while the test set included the final 7 months. User posts in the thread seemed to have decreased significantly in the last half of the year, as only about 8,000 of the usable text posts were included in the test set.

Upon manual inspection, topics were still interpretable after the splitting of the data into test and train sets, however more effort and consideration was required. Furthermore, it seemed that some topics could be split into further subtopics - for example, one grouping contained posts talking about the future of gamestop as a company, political/moral opinions about wealth, and other less-clear topics. These results were deemed sufficient for this project, however if more time was available, additional fine-tuning of the model would occur such as removing additional stop words, adjusting the vectorizer parameters for maximum/minimum document frequency, and discounting documents with less than ten tokens.

In terms of the unsupervised evaluations for finding user communities via clustering methods, as mentioned in the previous section, various measures such as WCSS, DBS, CHS, Silhouette and PR scores were used to first determine dimensions to use for PCA and then the ideal number of clusters. In Table 1.2 below, you can see how the various measures shift as we increase the number of dimensions. As expected, the WCSS increases as the dimension increases, meaning it might be better if we treat this measure as a reminder of the performance tradeoff that occurs as K-means will take longer to compute on higher dimensions. Then, we can find the optimal tradeoff between decrease in DBS and increase in CHS whilst keeping in mind the performance tradeoff. We see that the marginal benefit of adding additional dimensions to PCA begins to decrease after the dim=4.

Table 1.2

dim	wcss_orig	wcss_adj	dbs_orig	dbs_adj	chs_orig	chs_adj	silo_orig	silo_adj	pr_orig	pr_adj
2	13182.980025	16603.257568	1.666532	1.603774	1378.787722	1240.777726	0.263653	0.405412	0.514017	0.720654
3	22229.814761	30493.908156	1.505501	1.315872	1519.198378	1242.354747	0.304520	0.444605	0.538616	0.770006
4	29080.878648	38170.191268	1.267717	0.775419	1780.064072	1279.922120	0.302507	0.595668	0.550776	0.929106
5	37978.322465	51639.963332	1.163654	0.693789	1890.328632	1179.309967	0.328714	0.595065	0.564737	0.945161
6	45687.902039	60745.682453	1.209002	0.633722	1880.258025	1239.855258	0.327019	0.594400	0.577498	0.903669
7	52555.191158	66493.033059	1.092988	0.563515	2003.301890	1236.426192	0.344285	0.620030	0.602195	0.902189

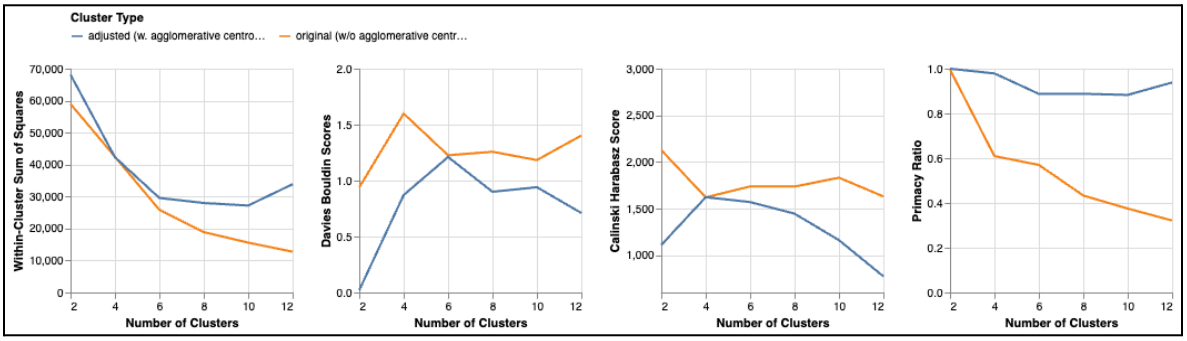
After dimension has been decided, we compared the performance of models with different numbers of clusters. As you can see, we found that the benefit of increasing K starts to diminish between 6 and 8, so we have chosen K=6 as due to lower DBS and higher CHS for the original clusters.

Table 1.3

	cluster_num	wcss_orig	wcss_adj	dbs_orig	dbs_adj	chs_orig	chs_adj	silo_orig	silo_adj	pr_orig	pr_adj
0	2	58965.288924	68116.321939	0.938584	0.020474	2126.964450	1109.356969	0.873739	0.969539	0.995644	0.999916
1	4	42458.208720	42298.441228	1.598281	0.871005	1620.057142	1621.962338	0.205934	0.687485	0.609482	0.978723
2	6	25846.423408	29564.949352	1.226071	1.210468	1737.234400	1569.316661	0.225861	0.480142	0.569945	0.887335
3	8	18881.165659	27981.991495	1.257303	0.899884	1735.505898	1445.048816	0.210756	0.480114	0.433155	0.887753
4	10	15592.369084	27195.143994	1.183312	0.940670	1830.863921	1160.929534	0.152063	0.472157	0.374937	0.882644
5	12	12741.816094	33864.299600	1.402751	0.710013	1629.758619	772.918403	0.146689	0.484572	0.321494	0.938264

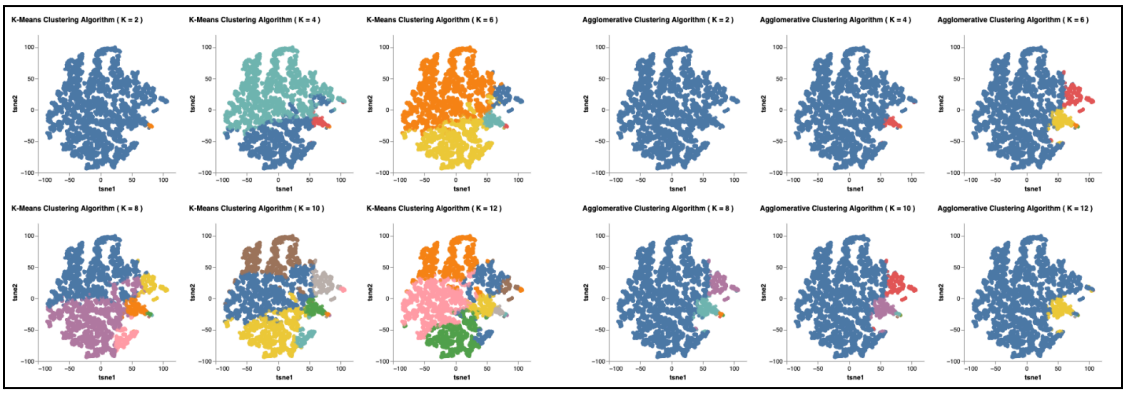
The result can be found in the visualization Plot 1.4. Given that we should expect WCSS and CHS to continually decrease while DBS continually increases as we increase the number of K, we can do the elbow method to confirm that we have chosen an optimal cut-off. From WCSS, we visually see that adding more clusters stops lowering the score as much after 6 to 8 clusters. Interestly, we also see that performance diverges between DBS and CHS for original K-Means and adjusted K-Means that use centroids from Agglomerative clustering. That being said, the PR for adjusted K-Means remains very high, suggesting that the scores are due to assignment of most of the users to a large cluster while others are treated like outliers and put into very small clusters.

Plot 1.4



In order to have an intuitive understanding of why we observe the above measures for the two models, we have also created side-by-side visualizations of the K-Means model and Agglomerative Clustering model. In order to show any hidden structures, we have further reduced the 4 dimensional PCA data into 2 dimensions using T-SNE. You can see the visualizations comparing the models in Plot 1.5:

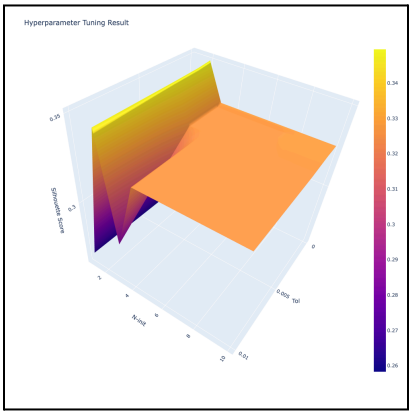
Plot 1.5



After the model type, PCA dimensions and number of clusters have been decided, we have performed a qualitative deep-dive by looking at the mean values of each of the clusters to see if they are reasonably distinct. For any clusters which were found to be too small, they were manually merged into larger, most similar clusters. For the details on the deep-dive, qualitative analysis of the individual cluster groups and silhouette analysis performed on K=6 clusters, please see the Appendix Section A.

Furthermore, sensitivity analysis was conducted to visualize how the average silhouette score changes as the hyperparameters in the K-Means models were adjusted in order to optimize performance. Specifically, we have adjusted tolerance (tol) and number of times the k-means algorithm is run with different centroid seeds (n_init). The chart on Plot 1.6 shows the outcome of that. Based on the sensitivity analysis, it seems like the scores are not too sensitive to those two main hyperparameters available for K-Means, and that highest performance is achieved for tol=0.000003 and n_init=2 to get the silhouette score of 0.349551.

Plot 1.6



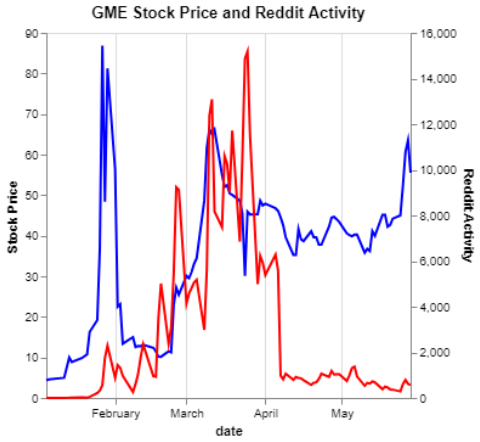
Supervised Machine Learning

Methods Description

The supervised learning portion of this project seeks to identify relationships between Reddit activity and the share price of GameStop. To represent Reddit activity, we are first identifying communities and topics using the Clustering and LDA models from the unsupervised learning portion of the project. These labels both individually and combined (e.g. Community A talking about Topic B), form the basis of the supervised learning features. Activity is calculated as the daily count of this community or topic posting. Our belief is that who (communities) is posting about what (topics) on Reddit will be critical to finding a relationship in Reddit data to the share price, if one does exist.

The community and topic labels are provided as two separate outputs, both assigning a label (community or topic) to a particular Reddit post. In order to prepare the data for the supervised learning model, the data is first consolidated into a single Reddit source that includes the communities, topics, and combined communities discussing topics labels. The GameStop daily stock price data is then blended with the Reddit data. There is some minor data cleanup required and data imputation to handle missing values.

Plot 2.1



The primary data transformation to prepare the data for use in the model is converting the labeled posts to community and topic activity. This converts our data from rows representing posts on dates with labels to counts of community and/or topic posts by date. These counts represent the values of our features, the community and/or topic activity. At this point, we are able to take an overall look at the Reddit activity and changes in share price, there does appear to be some common movement as seen in Plot 2.1.

We recognized that the timing of this activity may lead or lag changes in the stock price. As our interest is primarily in leading indicators, we include additional features that shift the date of the existing

features by a given number of days. Our assumption here is that we need to account for response time in the market following activity on Reddit.

Our objective was to explore the relationship between Reddit activity and share price, so we were focused on regression models. To find the most suitable model type, we chose 3 regression models with different strengths and methods.

Model	**Reason for Selection**
Linear Regression	Due to its simplicity and interpretability, we included an ordinarily least squares linear regression model.
Support Vector Regression (SVR)	This model was selected due to its versatility. It can be an effective solution with a large number of features and allows for a variety of different kernel functions to adapt to both linear or non-linear relationships in the data.
Decision Tree Regression	This model was selected given its strengths with complex patterns, non-linear relationships, ease of use, and interpretability.

In order to evaluate the three models, we focused on these key factors.

- Feature Selection: To select optimal features, features were ranked based on their correlation with share price. The top ten features were selected and used in the training of the three supervised models.
- 5 Fold Cross Validations on Time Series: Traditional cross validations randomly select folds. This generally isn't appropriate for time series data as it can result in leakage when mixing values from prior and future periods. The TimeSplitCV module was used to ensure cross validation splits respect the temporal order of the data.
- Grid Search Hyperparameter Tuning: To effectively compare the three models, each requires some tuning to ensure comparison of a reasonably optimal model. GridSearchCV was used to return the optimal model, parameters, and score for each of the three models being compared.

This overall methodology allowed us to select optimal features, use them to train all three models while simultaneously leveraging cross validation and hyperparameter tuning to compare which was best suited for our application.

Supervised Evaluations

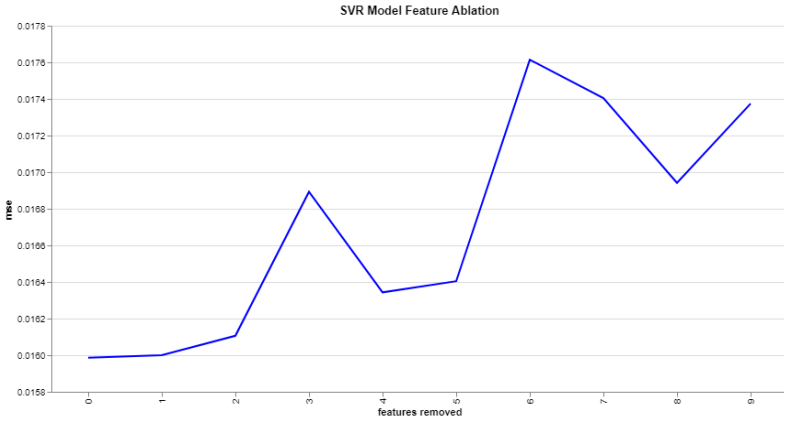
To evaluate the effectiveness of the three models, the mean squared error (MSE) was compared for the three models given the same features and optimal hyperparameters. The results of that comparison are identified in Table 2.1. MSE was chosen given its wide use in assessing regression model results, its sensitivity to outliers ensuring that all predictions are close to the expected value, and its ease of interpretability.

Table 2.1

Model	**Parameters**	**MSE**	**STD**
Linear Regression	{'Linear__fit_intercept': True}	12.77	0.466
Support Vector Regression	{'Support Vector (SVR)__C': 0.1, 'Support Vector (SVR)__gamma': 0.1, 'Support Vector (SVR)__kernel': 'rbf'}	0.017	0.016
Decision Tree Regression	{'Decision Tree__criterion': 'squared_error', 'Decision Tree__max_depth': 20}	0.159	0.214

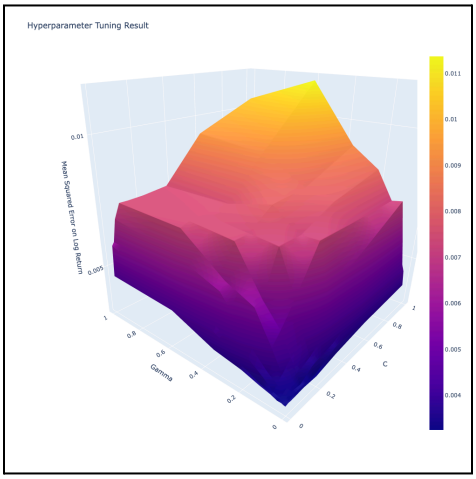
Given the three models, the Support Vector Regression model performed the best, with an MSE of 0.016. This was not particularly surprising given its versatility. However, as we observed that the Decision Tree model performed considerably better than the Linear model, we inferred a non-linear relationship between the share price and features. Using the Support Vector Regression model, we examined further how features and hyperparameters affect the results. First, we explored feature ablation to determine how removal of the features impacted the results of the model. Reconstructing a model with the same hyperparameters above, we trained and tested the model removing each of the ten features. The results are shown in figure Plot 2.2.

Plot 2.2



The feature ablation analysis showed that removal of most features increased the MSE of the SVR model. Two in particular created a significant jump in the error. They were the 3rd feature removed ('Core, influential Redditors community talking about Unclear Topic - Some posts about holding, FINRA, and a lot of external links_shift1') and the 6th feature removed ('MOD (moderator) Announcements'). This appeared unusual, suggesting the correlation may be with overall activity in the forum and not the content of particular groups and topics.

Plot 2.3



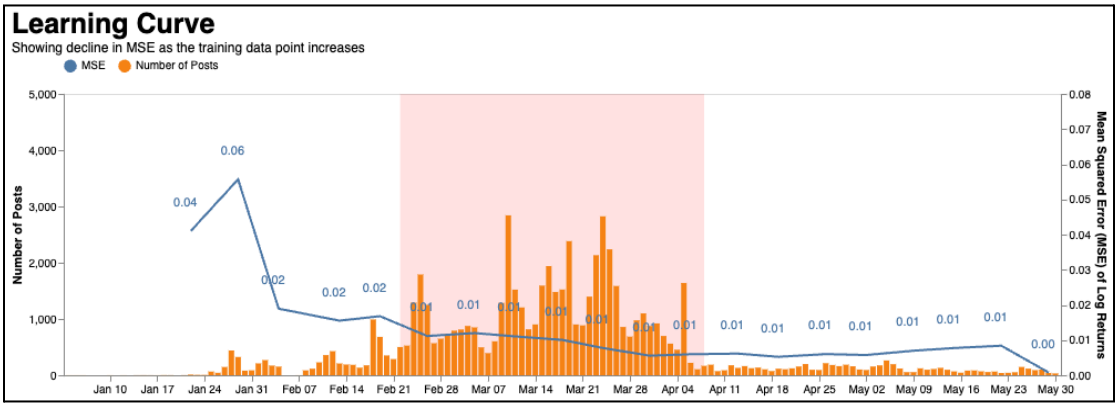
Secondly, while we have already obtained hyperparameter values using GridSearch on training data, we also wanted to evaluate how changing the hyperparameter values would impact the performance on testing data by conducting sensitivity analysis, looking at how the MSE changes as Gamma and C are shifted. The result can be found in Plot 2.3. We observe the same result as GridSearch in that lower value of Gamma and C leads to better performance. This makes sense, as having low values for both leads to a more generalizable model. Moreover, we were able to conclude that the model was indeed sensitive to changes in hyperparameters as MSE from 0.01 to 0.003. The fact that MSE was lower than in GridSearch can be attributed to the lack of decrease in volatility of GME price after May.

Finally, we conducted a learning curve analysis in order to see how the performance of the model changes as the data point increases. The model used for this had the gamma and C values obtained from GridSearch hyperparameter tuning. This type of testing was highly informative for our topic of interest because while we do have price for all periods, the volume of posts, split by communities and topics, change over a period of time. By knowing when the score, measured by mean squared error, stopped decreasing, we were able to understand the appropriate volume of new training data needed to get the model to be performant. The result of the learning curve analysis can be found in Plot 2.4.

In terms of the result, we observed that the mean squared error starts to level after February. This was because data volume was very low at the start of January and began to gradually increase. The MSE score stopped decreasing completely after April, which made sense since the number of posts was

highest between February and April, so there was no marginal improvement in performance after that point. As for the actual scores, while the initial MSE of 0.04 to 0.06 for January seemed quite low, we

Plot 2.4

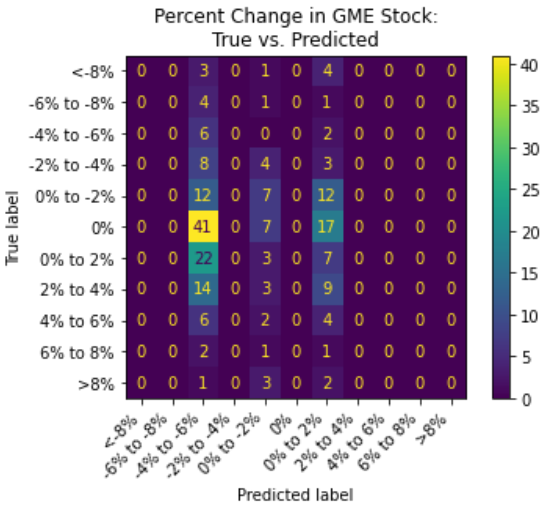


need to recall that this is MSE of log return, so getting the returns incorrect by 0.06 or roughly 6% indicates huge deviations. The MSE of 0.01 after all the data until April suggested a much more acceptable level of performance. This improvement after April could be due to the decrease in volatility, not improved training of the model given the additional data.

Given the results, the relationship between share price and the activity of communities and topics was non-linear. This raised a potential tradeoff between predictability and inference. Mendex (2019) describes, as non-linear models become increasingly complex, they may perform well at predictions but interpreting the nature of the relationship to make inference becomes more difficult. Failure Analysis

Failure Analysis

Plot 2.5



Binning was used to make the data test results discreet, so that they could be represented in a heat map for interpretability. A bin for zero price movement was created for the 65 days in which the market was closed, or the price didn't change. For the remaining 148 days, the model was able to successfully predict positive price movement in 23 of them and negative price movement in 46 of them. There were 57 days in which the model predicted the stock to drop but it instead increased (false negative) and 22 days in which it predicted the stock to go up but it decreased (false positive).The false negatives may be less of an issue if the model were to be used to try to increase earnings from trading (investors won't lose anything if the stock increases when they expect it to go down). From this perspective, the model could incorrectly give the impression that it performed decently. However, the model was quite poor in determining the magnitude of the price movement. The R2 value was measured at -0.69 which is an extremely poor score for a regression model. The worst three failures were differences of 28.5%, -27.3%, and -18.6% when subtracting the observed value from the predicted value.

The most extreme failure (difference of 28.5%) showed that only two of the ten included features had a value other than 0, those being (1) “Core Influential Redditors community talking about Moderator Announcements” and (2) the topic of “Moderator Announcements” without an assigned community. Each of these fields only had a value of 1. The next worst failure (difference of -27.3%) only had a single field with a number other than 0: the “Moderator Announcements” without an assigned community. It had a value of 2. The third worst failure (difference of -18.6%) was a zero vector across all ten fields. Furthermore, analysis of the next three largest failures, ranging in absolute differences of 12.5% to 14.2%, showed that the only field with non-zero values was that of “Moderator Announcements” without an assigned community.

There are a few obvious problems here: first, the matrix that resulted for the test set is a sparse matrix. Investigation showed that many instances are zero vectors, and there are four fields that are also zero vectors. Only two of the ten fields have counts of more than 2 across the entire 7 months the test set represents – those being “Moderator Announcements” without an assigned community (139 posts) and “Core Influential Redditors community talking about Moderator Announcements” (11 posts). Clearly a dataset with only 10 fields and 213 instances that is also sparse does not hold much information. Furthermore, the information it does hold seems to be largely about one topic: moderator posts.

Drastic changes would be required to allow more data to be available for the predictor to be effective. For starters, this could include (but is not limited to) the following:

- Include more fields in model training. Feature selection left out most of the fields due to the assumption that they were less relevant. Additional topics could be determined in the LDA portion of the project to allow for more fields and allow the categories to be finer grained. Furthermore, certain parameters were built into the design model to allow for additional fields, such as date shift and rolling averages.
- Better balance train and test data. Despite the train data only spanning 5 months, it still contained well over 90% of all the Reddit posts that were assigned a topic. The sets were split in such a way due to concerns of data leakage present in a time series. Perhaps other methodologies could be considered in which the analysis is not treated like a time series. This may be an important consideration because activity in the GME Subreddit was subdued in the later half of the year when compared to the first few months of 2021. The two periods are not really comparable.
- Include more data instances. The GameStop Subreddit used for analysis contained over 270,000 posts but only about 94,000 were used because of deletions, removals, etc. Since they had no topic assigned, they were dropped by default in the supervised learning portion of the project. Another solution is to use other posts in other Subreddits that mention GameStop by name, or to expand the data used to include other years besides 2021.
- Represent data differently to prevent information loss. The LDA model assigned probabilities to each data instance for each topic, however, each instance was only assigned one topic based on the highest scored probability. This means that even if a post only had a 20% chance of being assigned a topic, it was still assigned that topic if the probability was the highest. This throws away important context about a post. Moreover, the additional probabilities could be used to create additional features if they were to be included in the supervised portion of the project.

Discussions

The most surprising aspect of the LDA results was how easily the topic narratives were determined, even though sometimes a topic wouldn’t be completely clear or seemed to be a combination of more than one topic. It was also surprising how much the topic semantics could change with different runs of the model or with differing input parameters. To increase interpretability of some of the less-obvious topics, certain tokens such as “www”, “http”, and “com” were removed from the token list since they would appear as top

words in multiple topics. This seemed to slightly help the interpretability, although it increased the coherence measure. One disappointing aspect of the model was that many of the text posts did not clearly belong to one topic. With the probability distribution spread across 10 topics, a text post could easily be assigned a topic even if the probability of belonging to that topic was less than 50%. The perplexity readings during parameter tuning were also problematic, because different model families used different maximum feature input parameters, making it harder to compare them. This caused us to focus more on coherence as a performance measure. With more time and resources, we would explore different coherence measures to see if interpretability could be hastened by being better quantified, and perhaps replace perplexity with log-likelihood as a performance measure (to make models with different maximum feature parameters more comparable). We'd also explore other parameter modifications such as removing additional stop words or changing the document frequency of the vectorizer. We would also use Latent Semantic Indexing to see if it yielded a better result, and perhaps run a sub-model on documents belonging to an unclear topic to see if additional topics could be determined.

In terms of the result for unsupervised, clustering methods conducted to find distinct user communities, we were surprised by both the lack of the balance in number of users between the clusters and the instances of clusters which had very few users, likely those who would otherwise have been considered. In order to deal with the uneven distribution of users, the idea of primacy ratio, which refers to the percentage of total users the largest cluster makes up, was taken into consideration. Additionally, users with low numbers of users were merged into larger clusters to make it easier to choose the appropriate features for the supervised model and not unnecessarily increase that model's complexity. There are several ways that this component of the project would have been extended had we been given more time. First of all, we would spend more time exploring MDS and UMAP as an alternative to PCA, which was chosen due to the practical reason of decreased computational requirements (testing the two other dimension reduction techniques in depth would have taken too long). It may be that those two would have yielded better results. Secondly, another extension would be deploying effective outlier removal techniques so that clustering won't be influenced by outliers. While Kernel Density Estimator (KDE) was initially considered for this purpose, we have decided not to use it in the end as KDE requires the entire distribution and cannot adapt to new, unseen data, something that our model required.

The supervised results were surprising in the nature of the relationship between Reddit activity and stock price. Given the volatility of meme stocks like GameStop, we anticipated a stronger linear relationship with certain key communities or topics. One of the challenges we encountered was applying cross validation techniques using time series data. Initially, our model results were considerably worse than the mean model. Once the issue was identified, we were able to research and implement a 5 fold cross validation technique that respected the temporal nature of the data. With more time, we could extend this solution to other social media sources and stocks, extracting more robust features observed over a longer time period. This would allow us to further refine our model and understanding of the relationship between social media and investing.

Ethical Considerations

With respect to the unsupervised model, some of the ethical considerations that we harbor are concerns about privacy, reidentification and informed consent. Given that we are taking user posts from Reddit on various forums to perform clustering and topic identification, we are processing a lot of information that might have personally identifiable information (PII) in the form of user names and post content where users may have unknowingly included such information. While we did aggregate data into communities and topics, we should still be conscious that there is always concern about reidentification of individuals. In order to mitigate any risk that arise from privacy and reidentification, one of the ways we could address

it is by learning, for any future projects or improvement on this project, to research ways to systematically eliminate PII (e.g., AWS' Marcie).

Another potential issue is to do with informed consent. While users who post on Reddit forums are aware that any posts are public and have the ability to delete comments, they may have not given explicit consent for their comments to be used in an analysis. There is also an idea of the right to be forgotten where posts they have deleted on the forum will not be deleted from the content of our model. In order to address this, we can improve the model by allowing for real time updates on removed posts and work with Reddit to improve interpretability of disclaimers on the confidentiality and external usage of user posts.

As for the supervised component, while we did conduct this study with learning outcome in mind, there are real ethical issues if this model is put into production. First is definitely regarding the concept of "harms". This model that we have created might be used by someone to predict stock prices (although we are neither qualified professionals in investing nor endorse the use of this model to inform on investing in any way). In such a case, it is very possible that the person in question might incur financial losses. This ability for the model to cause real world harms should be kept in mind. In order to mitigate these issues, we can be explicit in the educational nature of the content (as in, it educates the report creators) and that the model should in no way be used for investing. Another concern is that, if put into production, there could be an issue of accountability and liability in that it can be vague. We can improve this by making sure that our code is not used by any external parties if possible or clearly marking in the code via comments and performing version controls to keep track of any updates to the original code. We have also used an Algorithmic Impact Assessment tool to assess potential impact of our model. Please refer to the Appendix C for the responses.

Statement of Work

Throughout the project, we have aimed for and achieved equitable distribution of work. In terms of the actual distribution, we have split the three distinct parts needed to complete the final model amongst the three project members:

- Stephen Moilanen researched, created and evaluated the unsupervised model in the form of LDA to perform topic discovery within the text content of the Reddit forum posts. Assisted with failure analysis for supervised machine learning.
- Kento Oigawa researched, created and evaluated the unsupervised model in the form of K-Means clustering to perform user communities identification within the non-text data of the Reddit forum posts. Assisted in feature selection/ablation, learning curve and hyperparameter tuning for the supervised model.
- Carl Debski researched, created and evaluated three different supervised models (Linear Regression, SVR and Decision Tree Regressor) to perform regression analysis on the data output from the two unsupervised machine learning models

It was decided earlier in the project that two members should work on two distinct models for unsupervised machine learning due to the fact that the final performance of the supervised model depended heavily on the performance of the unsupervised models that will produce the input data for it. As for other tasks such as creating the proposal drafts, creating stand-ups and writing of the final report, we sought for and achieved equitable distribution by making sure that everyone contributed to each component and adjusting contributions depending on the level of work required for model research, as some models inherently have more challenges than others.

References

Roder, Both, and Hinneburg. 2015. Exploring the Space of Coherence Methods.
WSDM 2015: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, pages 399-408.

Cote, Dave. 2022. Experimenting Confusion Matrix for Regression - A powerful model analysis tool
https://medium.com/@dave.cote.msc/experimenting-confusion-matrix-for-regression-a-powerfull-model-analysis-tool-7c288d99d437

Mendex, R. 2019. *The 3 BIG trade-offs in Statistical Learning (1. Prediction vs Inference).*
https://medium.com/@ro.flores.mendez/the-3-big-trade-offs-in-statistical-learning-1-prediction-vs-inference-b731d2f0904f

Appendix

Section A: Deep-dive into clusters found by optimal K-Means for identifying user communities and Silhouette Chart

Once the optimal K-Means have been identified, in addition to the visualizations, we wanted to group the values of the original data based on labels so that we can manually inspect the goodness of the clusters to ensure that each group contained distinct users. The result of the groupings can be found in Table 3.1. The numbers are the mean values of the original columns - the mean was the most appropriate.

Table 3.1

orig_6	locked	removed	is_self	is_video	is_original_content	upvote_ratio	score	gilded	total_awards_received	num_comments	num_crossposts
0	0.027860	0.495104	0.623059	0.009204	0.000000	0.893123	44.544141	0.003637	0.146322	6.873124	0.014693
1	0.000144	0.048453	0.732524	0.002335	0.000000	0.826637	57.711746	0.004085	0.170996	10.138169	0.015654
2	0.011154	0.024799	0.705554	0.021412	0.001068	0.912346	2273.044459	0.824670	26.678168	304.336596	1.580260
3	0.001727	0.035969	0.582401	0.025500	0.000212	0.900097	810.144280	0.135941	3.807998	67.054093	0.304721
4	0.000000	0.000000	1.000000	0.000000	0.250000	0.980000	1859.250000	1.750000	25.500000	180.000000	0.500000
5	0.000141	0.049164	0.242928	0.067422	0.000012	0.885645	94.805441	0.006475	0.245938	9.403415	0.031414

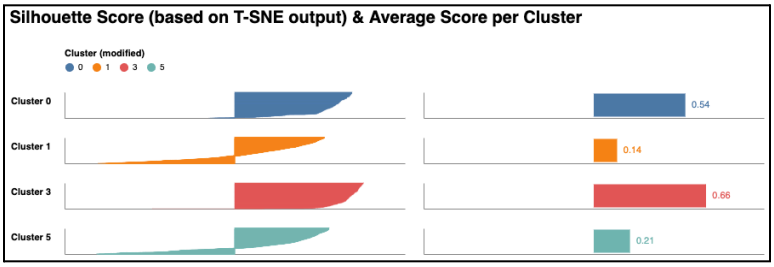
Based on the results, we can find some clusters which are distinct from others. Some, however, had very low numbers of users, suggesting they were potential outliers. For the clusters with low number of users (namely clusters 2 and 4), they were merged with larger, more distinct clusters. Below are the basic description of the clusters which were identified:

- Cluster 0 (Highly controversial): This community was marked by high number of locked and removed posts
- Cluster 1 (Unpopular): This community was marked by low upvote and low score, suggesting not much engagement from others and opinion not too popular
- Cluster 3 (Core, influential redditors): This community was marked by high scores, upvote ratio, and very high number of comments and crossposts
- Cluster 5 (Scriptophobic): This community was marked by low proportion of text based posts and high ratio of video posts

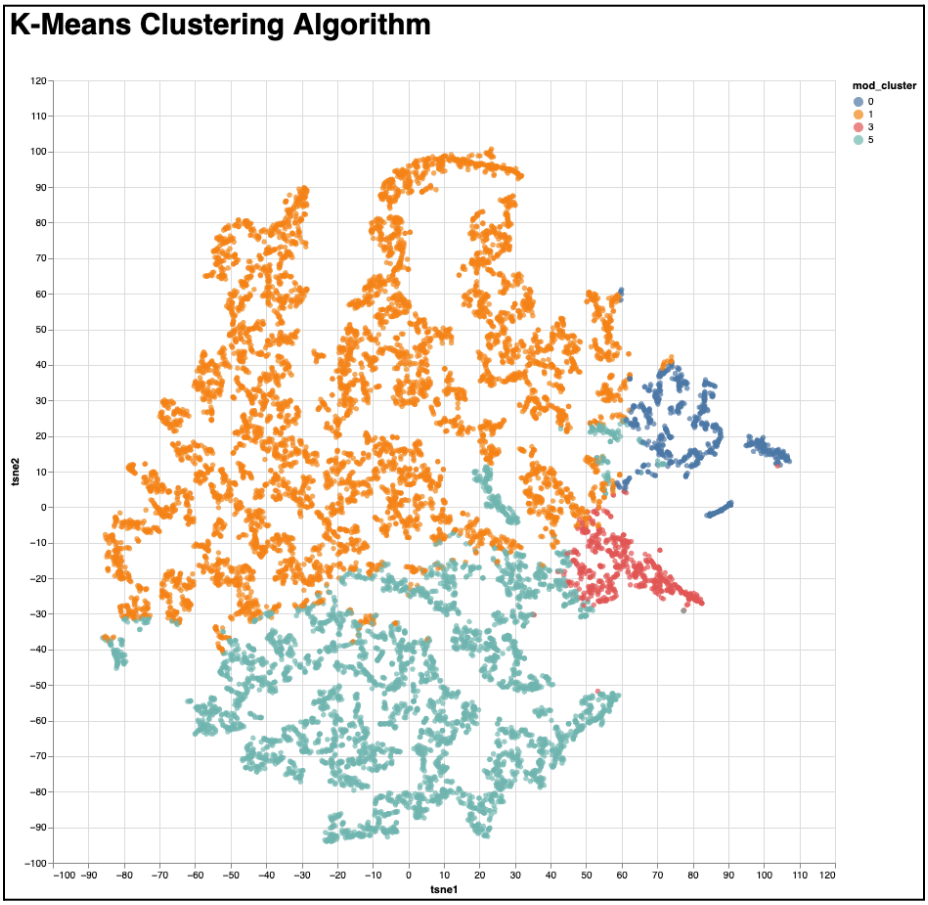
On Plot 3.1 is a Silhouette chart to understand the quality of each cluster. While it is standard to use the original, standardized data to calculate Silhouette scores, the two-dimensional, reduced T-SNE data was used as an input due to the high number of features in the original data. It is interesting to see that

clusters 0 and 3 have much higher scores than other clusters.Lastly, we can print out the visualization of the final resultant clusters with labels to see if clusters which are spatially close to one another represent similar users to confirm that our communities have been split as expected. In the Plot 3.2, you can see that clusters 1 and 5 are the main clusters that divide users horizontally. Then you can see that clusters 0 and 3 were located towards the edge on the right hand side. Given that Clusters 0, 3, and 5 represent users with high upvotes, scores etc, it's possible that y=-x represent the degree to which users' posts are well-received.

Plot 3.1



Plot 3.2



Section B: Instruction on How to Run Project Codes

The link to our Github Repository where we store our codes can be found here.

Section	Content	Details
Unsupervised Machine Learning: Topic Detection	LDA Modeling Code, located within subfolder "smoilanen_milestone_II".	To perform topic modeling, run Jupyter Notebook "GME_MAIN.ipynb" and follow instructions within the notebook. For parameter tuning plot coding for LDA, run Jupyter Notebook "Plots_GME.ipynb" and follow instructions within the notebook. In order for the script to run, please download Kaggle data from the link provided in the data source section.
Unsupervised Machine Learning: Community Clustering	Requirements	koigawa_milestone_II/requirements.txt The text file above contains all the libraries (with versions) required to run all the unsupervised components as well as learning curve and hyperparameter tuning components of supervised ML
	Visualizations	Running all cells within communities_clustering.ipynb notebook will produce visualizations which are seen in this report for community clustering. Please note that the first cell should install all the required libraries via pip command so everything proceeding should be runnable.
	Scripts to produce data (CSV) to be used by downstream supervised machine learning model	In order to produce user communities data (CSV) used in the supervised section, run below scripts in the specified order. 1. python3 shared_milestone_II/download_reddit.py 2. python3 koigawa_milestone_II/return_clustering_results.py The first script will download Reddit forum data from Kaggle into a file reddit/gme/submission_reddit.csv whereas the second file will produce koigawa_milestone_II/community_output_gme_train.csv and koigawa_milestone_II/community_output_gme_test.csv which is referenced by the supervised models. Please note that these output files can be large in size (greater than 50MB) so any files within Coursera submission are only samples of the files, not the entire files. For the entire files, please run the scripts mentioned above in

		your own environment.
Supervised Machine Learning	Within the GitHub repository, folder cdebski_milestone_II, model_comparison.ipynb	Import the required libraries using the requirements file in the Github repository folder. Open the notebook, model_comparison.ipynb and select run all. This will save the best models, preprocessed data, and model comparison results in the cdebski_milestone_II folder.
	Visualizations for feature ablation, hyperparameter tuning, and learning curve	Visualizations for the feature ablation can be obtained by running all cells in shared_milestone_II/feature_ablation_analysis.ipynb Visualization for the learning curve can be obtained by running all cells in shared_milestone_II/learning_curve_report.ipynb, but only after creating all the required files by running all scripts mentioned above. Visualization for the hyperparameter tuning can be obtained by running all cells in shared_milestone_II/hyperparameter_tuning_report.ipynb, but also only after creating all the required files by running all scripts mentioned above.
	Failure Analysis Code, located within subfolder "smoilanen_milestone_II".	To perform failure analysis, run Jupyter Notebook "GME_FAILURE_ANALYSIS.ipynb" and follow instructions within the notebook.

.

Section C: Responses to Algorithmic Impact Assessment Tool

We have taken the Algorithmic Impact Assessment Tool provided by the Government of Canada to place a numerical value on the level of impact our model will have. Even though not all questions are applicable to our model, mostly because it is intended for models that are used within the government, it is still valuable as it poses many questions related to ethics which allows us to come up with ways to address them in the future.

Link to the questionnaire (not the result): https://www.canada.ca/en/government/system/digital-government/digital-government-innovations/responsible-use-ai/algorithmic-impact-assessment.html Project Details

1. Job Title
Student
2. Project Title
Clustering Reddit Threads to Analyze Correlation Across Stock Market
3. Project Phase
Implementation [Points: 0]
4. Please provide a project description:
This project seeks to find any relationship between activities and posts on social media platform and the prices of stocks that are mentioned within the platform.
About The System
5. Please check which of the following capabilities apply to your system.
Text and speech analysis: Analyzing large data sets to recognize, process, and tag text, speech, voice, and make recommendations based on the tagging
Content generation: Analyzing large data sets to categorize, process, triage, personalize, and serve specific content for specific contexts
Section 1: Impact Level : 1
Current Score: 26

Raw Impact Score: 26

Mitigation Score: 7

Section 2: Requirements Specific to Impact Level 1
Peer review
None

Gender-based Analysis Plus
None

Notice
None

Human-in-the-loop for decisions
Decisions may be rendered without direct human involvement.

Section 3: Questions and Answers
Section 3.1: Impact Questions and Answers
Reasons for Automation
1. What is motivating your team to introduce automation into this decision-making process? (Check all that apply)
Use innovative approaches
The system is performing tasks that humans could not accomplish in a reasonable period of time
Improve overall quality of decisions
2. How effective will the system likely be in meeting client needs?
Slightly effective [Points: +2]
3. Have alternative non-automated processes been considered?
No [Points: +1]
4. What would be the consequence of not deploying the system?

Service cannot be delivered at all [Points: +3]
Risk Profile
5. Is the project within an area of intense public scrutiny (e.g. because of privacy concerns) and/or frequent litigation?
No [Points: +0]
6. Are clients in this line of business particularly vulnerable?
No [Points: +0]
7. Are stakes of the decisions very high?
No [Points: +0]
8. Will this project have major impacts on staff, either in terms of their numbers or their roles?
No [Points: +0]
9. Will the use of the system create or exacerbate barriers for persons with disabilities?
No [Points: +0]
Project Authority
10. Will you require new policy authority for this project?
No [Points: +0]
About the Algorithm
11. The algorithm used will be a (trade) secret
No [Points: +0]
12. The algorithmic process will be difficult to interpret or to explain
Yes [Points: +3]
About the Decision
13. Please describe the decision(s) that will be automated.
Which topics/communities a post from a Reddit user belongs to.
14. Does the decision pertain to any of the categories below (check all that apply):
Economic interests (grants and contributions, tax benefits, debt collection) [Points: +1]
Impact Assessment
15. Which of the following best describes the type of automation you are planning?
Partial automation (the system will contribute to administrative decision-making by supporting an officer through assessments, recommendations, intermediate decisions, or other outputs)
[Points: +2]
16. Please describe the role of the system in the decision-making process.
The system will not make decision in any way, but instead seeks to inform the viewer of a relationship, if any.
17. Will the system be making decisions or assessments that require judgement or discretion?
Yes [Points: +4]
18. Please describe the criteria used to evaluate client data and the operations applied to process it.
Series of unsupervised machine learning models will process the data after initial preprocessing.
19. Please describe the output produced by the system and any relevant information needed to interpret it in the context of the administrative decision.
It will output correlation to measure how effective the supervised machine learning component is based on regressing topic/community proportion on time series of stock prices.
20. Will the system perform an assessment or other operation that would not otherwise be completed by a human?
No [Points: +0]
21. Is the system used by a different part of the organization than the ones who developed it?
No [Points: +0]
22. Are the impacts resulting from the decision reversible?

Reversible [Points: +1]
23. How long will impacts from the decision last?
Impacts are most likely to be brief [Points: +1]
24. The impacts that the decision will have on the rights or freedoms of individuals will likely be:
Little to no impact [Points: +1]
25. The impacts that the decision will have on the equality, dignity, privacy, and autonomy of individuals will likely be:
Little to no impact [Points: +1]
26. The impacts that the decision will have on the health and well-being of individuals will likely be:
Little to no impact [Points: +1]
27. The impacts that the decision will have on the economic interests of individuals will likely be:
Moderate impact [Points: +2]
28. Please describe why the impacts resulting from the decision are as per selected option above.
If anyone decides to use the model to inform their decisions on trading in the financial markets, there can actually be financial impact.
29. The impacts that the decision will have on the ongoing sustainability of an environmental ecosystem, will likely be:
Little to no impact [Points: +1]
About the Data - B. Type of Data
30. Will the system require the analysis of unstructured data to render a recommendation or a decision?
Yes [Points: 0]
31. What types of unstructured data? (Check all that apply)
Audio and text files [Points: +2]
Section 3.2: Mitigation Questions and Answers
Consultations
1. Internal Stakeholders (federal institutions, including the federal public service)
No [Points: +0]
2. External Stakeholders (groups in other sectors or jurisdictions)
No [Points: +0]
De-Risking and Mitigation Measures - Data Quality
3. Do you have documented processes in place to test datasets against biases and other unexpected outcomes? This could include experience in applying frameworks, methods, guidelines or other assessment tools.
No [Points: +0]
4. Is this information publicly available?
Yes [Points: +1]
5. Have you developed a process to document how data quality issues were resolved during the design process?
Yes [Points: +1]
6. Is this information publicly available?
Yes [Points: +1]
7. Have you undertaken a Gender Based Analysis Plus of the data?
No [Points: +0]
8. Is this information publicly available?
No [Points: +0]
9. Have you assigned accountability in your institution for the design, development, maintenance, and improvement of the system?
No [Points: +0]

10. Do you have a documented process to manage the risk that outdated or unreliable data is used to make an automated decision?
No [Points: +0]
11. Is this information publicly available?
No [Points: +0]
12. Is the data used for this system posted on the Open Government Portal?
No [Points: +0]
De-Risking and Mitigation Measures - Procedural Fairness
13. Does the audit trail identify the authority or delegated authority identified in legislation?
No [Points: +0]
14. Does the system provide an audit trail that records all the recommendations or decisions made by the system?
No [Points: +0]
15. Are all key decision points identifiable in the audit trail?
No [Points: +0]
16. Are all key decision points within the automated system's logic linked to the relevant legislation, policy or procedures?
No [Points: +0]
17. Do you maintain a current and up to date log detailing all of the changes made to the model and the system?
Yes [Points: +2]
18. Does the system's audit trail indicate all of the decision points made by the system?
No [Points: +0]
19. Can the audit trail generated by the system be used to help generate a notification of the decision (including a statement of reasons or other notifications) where required?
No [Points: +0]
20. Does the audit trail identify precisely which version of the system was used for each decision it supports?
No [Points: +0]
21. Does the audit trail show who an authorized decision-maker is?
Yes [Points: +1]
22. Is the system able to produce reasons for its decisions or recommendations when required?
No [Points: +0]
23. Is there a process in place to grant, monitor, and revoke access permission to the system?
Yes [Points: +1]
24. Is there a mechanism to capture feedback by users of the system?
No [Points: +0]
25. Is there a recourse process established for clients that wish to challenge the decision?
No [Points: +0]
26. Does the system enable human override of system decisions?
No [Points: +0]
27. Is there a process in place to log the instances when overrides were performed?
No [Points: +0]
28. Does the system's audit trail include change control processes to record modifications to the system's operation or performance?
No [Points: +0]
29. Have you prepared a concept case to the Government of Canada Enterprise Architecture Review Board?

No [Points: +0]
De-Risking and Mitigation Measures - Privacy
30. If your system uses or creates personal information, have you undertaken a Privacy Impact Assessment, or updated an existing one?
No [Points: +0]
31. Have you designed and built security and privacy into your systems from the concept stage of the project?
No [Points: +0]
32. Is the information used within a closed system (i.e. no connections to the Internet, Intranet or any other system)?
No [Points: +0]
33. If the sharing of personal information is involved, has an agreement or arrangement with appropriate safeguards been established?
No [Points: +0]
34. Will you de-identify any personal information used or created by the system at any point in the lifecycle?
No

Section D: Schema of the Reddit data

Below is a schema for the original CSV data.

Name	Type	Note	Used or contained in LDA	Used or contained in Clustering
id	String	The id of the submission	Y	Y
author	String	The username of the Redditor	N	Y
created	Datetime	Time the submission was created	Y	Y
retrieved	Datetime	Time the submission was retrieved	N	N
edited	Datetime	Time the submission was edited	Y	Y
pinned	Integer	Whether the submission is pinned	N	Y
archived	Integer	Whether the submission is archived	N	N
locked	Integer	Whether the submission is locked	N	Y
removed	Integer	Whether the submission is mod removed	N	Y
deleted	Integer	Whether the submission is user deleted	N	N

is_self	Integer	Whether the submission is user deleted	N	Y
is_video	Integer	Whether the submission is a video	N	Y
is_original_content	Integer	Whether the submission has been set as original content	N	Y
title	String	The title of the submission	Y	N
link_flair_text	String	The submission link flairs text content	N	N
upvote_ratio	Integer	The percentage of upvotes from all votes on the submission	N	Y
score	Integer	The number of upvotes for the submission	N	Y
gilded	Integer	The number of gilded awards on the submission	N	Y
total_award_received	Integer	The number of awards on the submission	N	Y
num_comments	Integer	The number of comments on the submission	N	Y
num_crossposts	Integer	The number of crossposts on the submission	N	Y
selftext	String	The submission selftext on text posts	Y	N
thumbnail	String	The submission thumbnail on image posts	N	N
shortlink	String	The submission short URL	N	N