Reddit Financial Channels

Adam Duke
Whitney Schreiber

{Data Mining Autumn 2021}

Agenda

1. Problem Introduction
2. Data Profile and Exploration
3. Feature Engineering
4. Modeling
5. Results

Executive Summary

“At the start of the year, traders added more than $150 billion to the market cap of GameStop, AMC, and 48 other businesses” –Bloomberg

In early 2021, select NASDAQ and NYSE listed securities that had mainly underperformed during the COVID-19 pandemic experienced suddenly radical and unstable price movements. This phenomenon was largely driven by simultaneous short position liquidation of institutional investors and mass opposing, collective retail trading activity. Popular speculation suggests observed market behavior was a consequence of observable posts, primarily those on specific Reddit channels. **We examine whether latent market volatility may be associated with text and other metadata of Reddit posts for known meme stocks, in particular GME.**

Reddit Financial Channel Project Decision

GOAL

Identify signals of abnormally high volatility in future stock price of a given “meme” stock using Reddit posts.

VALUE

Advance indication of volatility can afford reduced value at risk or provide opportunities for advantageous options trading strategies.

Reddit Financial Channel

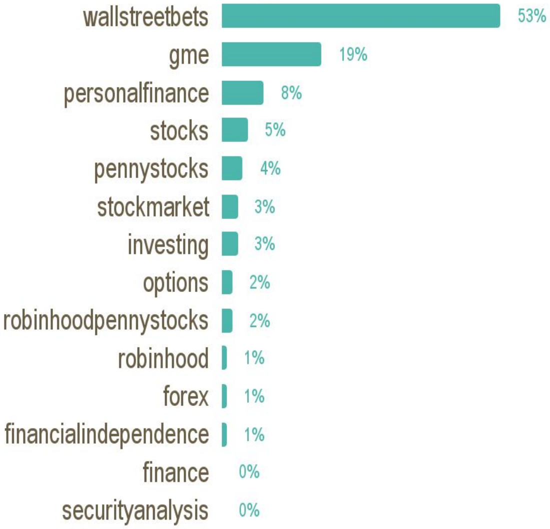
Data Exploration and Feature Engineering

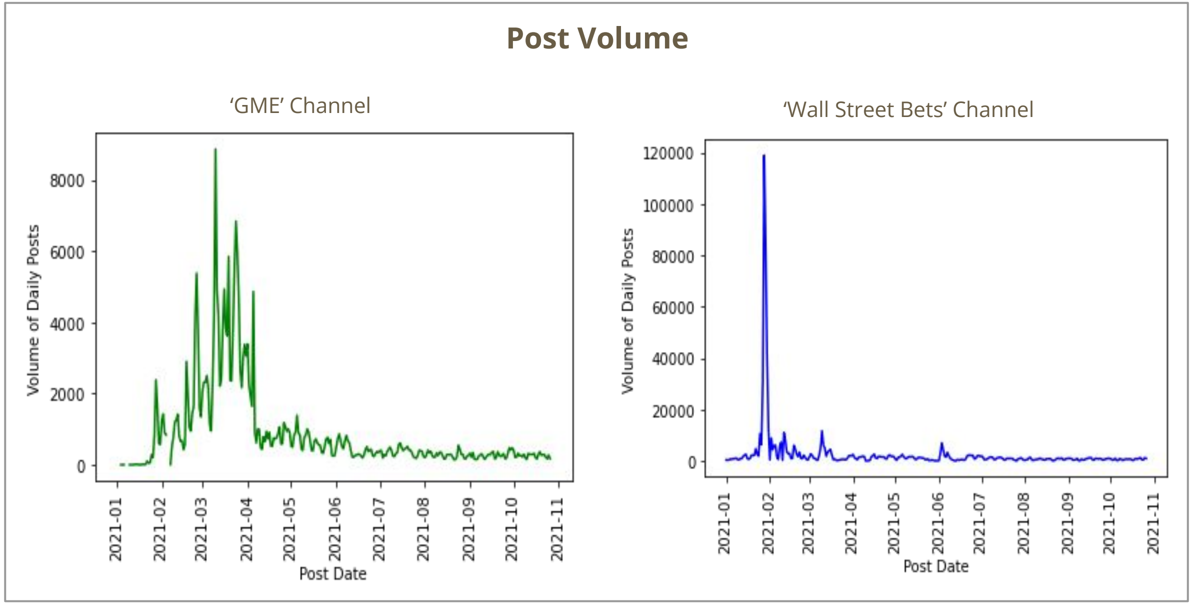
Reddit Financial Channel Raw Data

- **14** Channels
- **1,377,932** records (posts)

- Date range:
  1/1/2021 - 11/21/2021

Percent of Total Posts by Channel




Post Volume

Reddit Channel Data

Data Profile

Number of gilded awards
`gilded`
(integer)

Channel
`channel`
(string)

Author Name
`author`
(string)

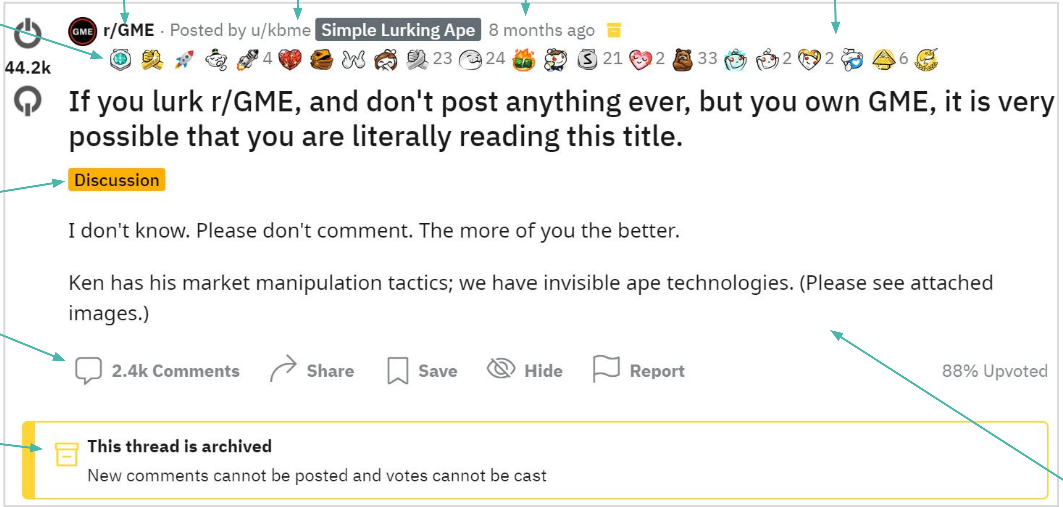
Post Creation Date
`created`
(datetime)

Total awards
`total_awards_received`
(integer)

Score
(number of upvotes)
`score`
(numeric)

Post Title
`title`
(string)

Link Flairs
`link_flair_text`
(string)



r/GME · Posted by u/kbme Simple Lurking Ape 8 months ago

If you lurk r/GME, and don't post anything ever, but you own GME, it is very possible that you are literally reading this title.

Discussion

I don't know. Please don't comment. The more of you the better.

Ken has his market manipulation tactics; we have invisible ape technologies. (Please see attached images.)

This thread is archived

New comments cannot be posted and votes cannot be cast

Number of Comments
`num_comments`
(integer)

Upvote Ratio
`upvote_ratio`
(numeric)

Post was removed
`removed`
(binary)

Post was deleted
`deleted`
(binary)

Post was pinned
`pinned`
(binary)

Post was locked
`locked`
(binary)

Archived
`archived`
(binary)

Body text
`self_text`
(string)

Number of Crossposts
`num_crossposts`
(integer)

Image post thumbnail
`thumbnail`
(string)

Post's short url
`shortlink`
(string)

Post is a text
`is_self`
(binary)

Post is a video
`is_video`
(binary)

Post set as original content
`is_original_content`
(binary)

Stock Market Data
Data Profile

Retrieve stock market data using **yfinance**

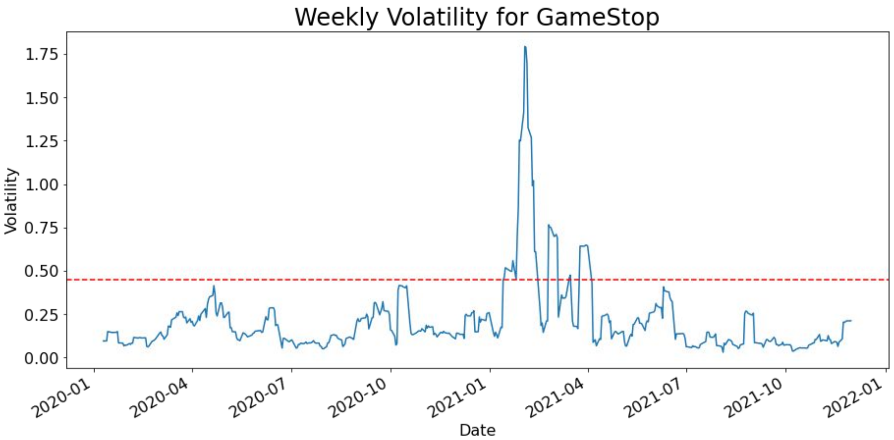
- Ticker Symbols: **GME**
- Date range: **1/1/2021 - 11/21/2021**
- Metrics: Open, High, Low, Close, Adj Close, Volume
- Calculate **volatility** using Adjusted Closing price

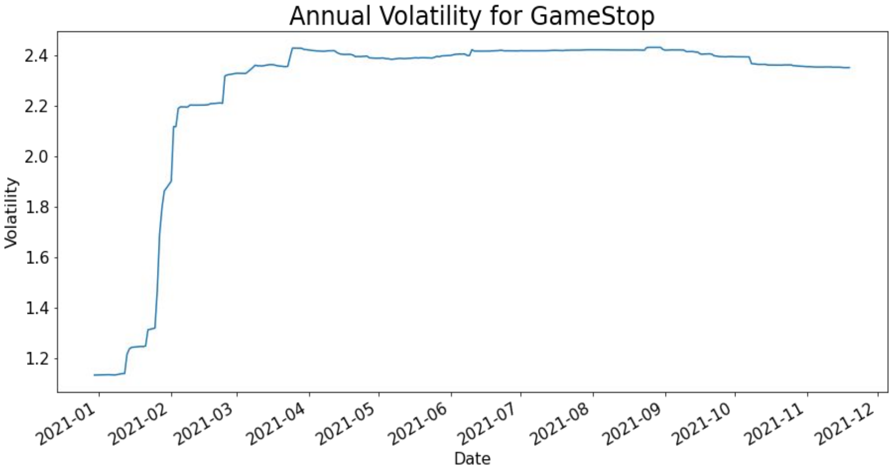
```
return = log(closing_price_tomorrow / closing_price_today)
```

```
volatility = 7_day_rolling_std(return) * sqrt(trading days)
```

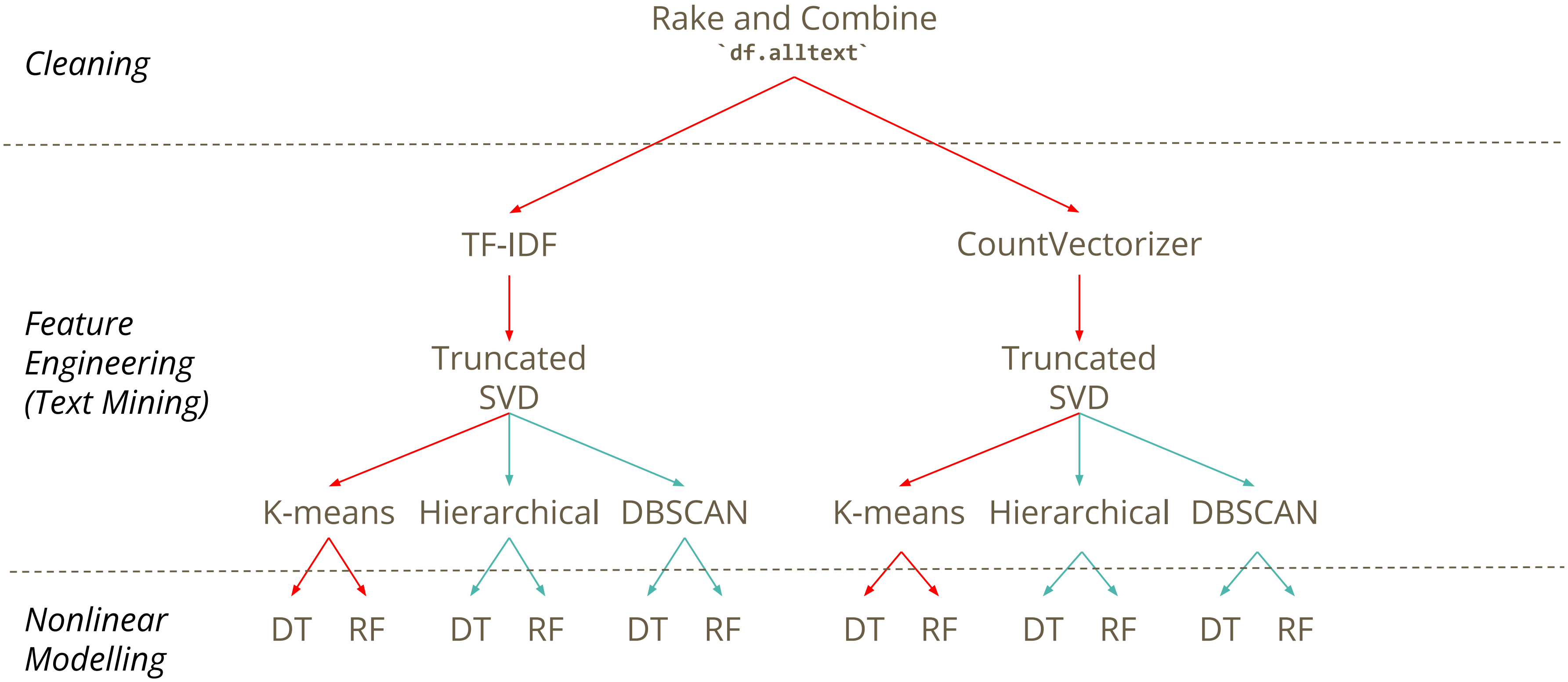
Target Variable

- Define **Mania**, a binary variable describing volatility
  - **True:** volatility >= threshold (0.45)
  - **False:** volatility < threshold (0.45)


Weekly Volatility for GameStop


Annual Volatility for GameStop

Methodology



Cleaning

Rake and Combine
`df.alltext`

Feature Engineering (Text Mining)

TF-IDF

CountVectorizer

Truncated SVD

Truncated SVD

K-means Hierarchical DBSCAN

K-means Hierarchical DBSCAN

Nonlinear Modelling

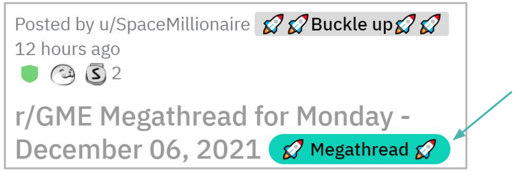
DT RF DT RF DT RF

DT RF DT RF DT RF

`link_flair_text`

Feature Engineering

One-hot encode the top 10 most common link flairs



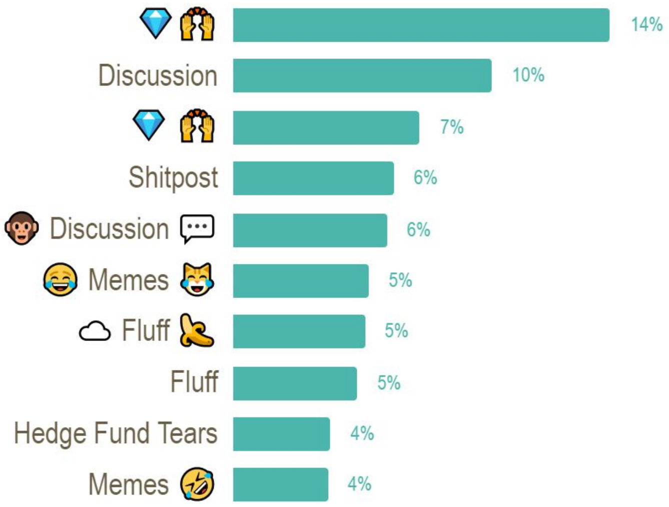
Posted by u/SpaceMillionaire 🚀🚀Buckle up🚀🚀
12 hours ago
2

r/GME Megathread for Monday - December 06, 2021 🚀 Megathread 🚀

Link Flair

'GME' Reddit Channel

Percent of Posts with Link Flair



	link_flair_text
0	Discussion
1	Shitpost
2	Discussion
3	Fluff
4	God Tier DD

	link_flair_text_DD	link_flair_text_	link_flair_text_Discussion	link_flair_text_Shitpost	link_flair_text_Memes	...
0	False	False	True	False	False	...
1	False	False	False	True	False	...
2	False	False	True	False	False	...
3	False	False	False	False	False	...
4	False	False	False	False	False	...

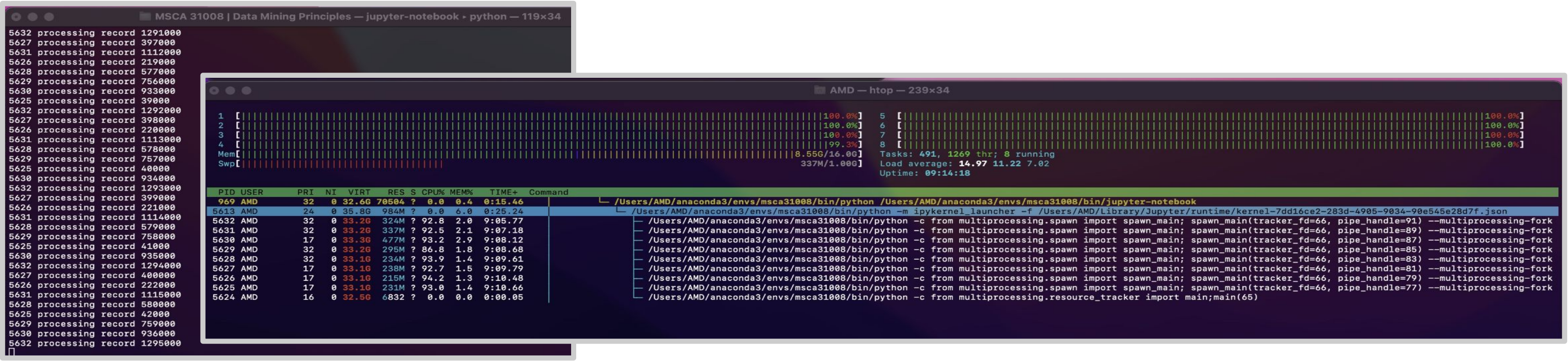
`title`, `selftext`

Feature Engineering

Generate Basket of Key Words

Parallel **rake** ‘title’ and ‘selftext’ fields and combine the keywords

Post title `title`	Post body text content `selftext`	Basket of words generated by rake `alltext`
GME is FINALLY going to the moon, this technical analysis looks very nice 🚀🚀🚀	After some downwards movement, I think everybody needs some good news, and here they are. We are seeing some very strong technical indications showing that we are in fact on the verge of breaking up once again, the target is $26! :) https://youtu.be/JlwXg5-H7cg	gme finally going moon technical analysis looks nice 🚀🚀🚀 downwards movement think everybody needs good news seeing strong technical indications showing fact verge breaking target 26 :) https :// youtu jlwxg5 h7cg](



MSCA 31008 | Data Mining Principles — jupyter-notebook ▸ python — 119×34

```
5632 processing record 1291000
5627 processing record 397000
5631 processing record 1112000
5626 processing record 219000
5628 processing record 577000
5629 processing record 756000
5630 processing record 933000
5625 processing record 39000
5632 processing record 1292000
5627 processing record 398000
5626 processing record 220000
5631 processing record 1113000
5628 processing record 578000
5629 processing record 757000
5625 processing record 40000
5630 processing record 934000
5632 processing record 1293000
5627 processing record 399000
5626 processing record 221000
5631 processing record 1114000
5628 processing record 579000
5629 processing record 758000
5625 processing record 41000
5630 processing record 935000
5632 processing record 1294000
5627 processing record 400000
5626 processing record 222000
5631 processing record 1115000
5628 processing record 580000
5625 processing record 42000
5629 processing record 759000
5630 processing record 936000
5632 processing record 1295000
```

AMD — htop — 239×34

```
  1  [||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||100.0%]   5  [|||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||100.0%]
  2  [||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||100.0%]   6  [|||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||100.0%]
  3  [||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||100.0%]   7  [|||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||100.0%]
  4  [||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||| 99.3%]   8  [|||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||100.0%]
  Mem[||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||8.55G/16.0G]   Tasks: 491, 1269 thr; 8 running
  Swp[||||||||||||||||||||||||||||                                                  337M/1.00G]   Load average: 14.97 11.22 7.02
                                                                                                  Uptime: 09:14:18

  PID USER      PRI  NI  VIRT   RES S CPU% MEM%   TIME+  Command
  969 AMD        32   0 32.6G 70504 ?  0.0  0.4  0:15.46 │      └─ /Users/AMD/anaconda3/envs/msca31008/bin/python /Users/AMD/anaconda3/envs/msca31008/bin/jupyter-notebook
 5613 AMD        24   0 35.8G  984M ?  0.0  6.0  0:25.24 │         └─ /Users/AMD/anaconda3/envs/msca31008/bin/python -m ipykernel_launcher -f /Users/AMD/Library/Jupyter/runtime/kernel-7dd16ce2-283d-4905-9034-90e545e28d7f.json
 5632 AMD        32   0 33.2G  324M ? 92.8  2.0  9:05.77 │            ├─ /Users/AMD/anaconda3/envs/msca31008/bin/python -c from multiprocessing.spawn import spawn_main; spawn_main(tracker_fd=66, pipe_handle=91) --multiprocessing-fork
 5631 AMD        32   0 33.2G  337M ? 92.5  2.1  9:07.18 │            ├─ /Users/AMD/anaconda3/envs/msca31008/bin/python -c from multiprocessing.spawn import spawn_main; spawn_main(tracker_fd=66, pipe_handle=89) --multiprocessing-fork
 5630 AMD        17   0 33.3G  477M ? 93.2  2.9  9:08.12 │            ├─ /Users/AMD/anaconda3/envs/msca31008/bin/python -c from multiprocessing.spawn import spawn_main; spawn_main(tracker_fd=66, pipe_handle=87) --multiprocessing-fork
 5629 AMD        32   0 33.2G  295M ? 86.8  1.8  9:08.68 │            ├─ /Users/AMD/anaconda3/envs/msca31008/bin/python -c from multiprocessing.spawn import spawn_main; spawn_main(tracker_fd=66, pipe_handle=85) --multiprocessing-fork
 5628 AMD        32   0 33.1G  234M ? 93.9  1.4  9:09.61 │            ├─ /Users/AMD/anaconda3/envs/msca31008/bin/python -c from multiprocessing.spawn import spawn_main; spawn_main(tracker_fd=66, pipe_handle=83) --multiprocessing-fork
 5627 AMD        17   0 33.1G  238M ? 92.7  1.5  9:09.79 │            ├─ /Users/AMD/anaconda3/envs/msca31008/bin/python -c from multiprocessing.spawn import spawn_main; spawn_main(tracker_fd=66, pipe_handle=81) --multiprocessing-fork
 5626 AMD        17   0 33.1G  215M ? 94.2  1.3  9:10.48 │            ├─ /Users/AMD/anaconda3/envs/msca31008/bin/python -c from multiprocessing.spawn import spawn_main; spawn_main(tracker_fd=66, pipe_handle=79) --multiprocessing-fork
 5625 AMD        17   0 33.1G  231M ? 93.0  1.4  9:10.66 │            ├─ /Users/AMD/anaconda3/envs/msca31008/bin/python -c from multiprocessing.spawn import spawn_main; spawn_main(tracker_fd=66, pipe_handle=77) --multiprocessing-fork
 5624 AMD        16   0 32.5G  6832 ?  0.0  0.0  0:00.05 │            └─ /Users/AMD/anaconda3/envs/msca31008/bin/python -c from multiprocessing.resource_tracker import main;main(65)
```

`alltext`

Feature Engineering

Transform Unstructured Text

alltext
gme moon 🚀🚀
need see gme 🚀🚀🚀🚀🚀🚀 watching took position rig...
short squeeze incoming 🚀🚀🚀🚀🚀🚀🚀
convinced 💰 gme extreme pump coming guy explai...
already know must brothers sisters submit comp...

TF-IDF

Transform text to an array by comparing the word frequency in a doc and the number of docs with the word

Output: sparse array

```
[[0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 ...
```

CountVectorizer

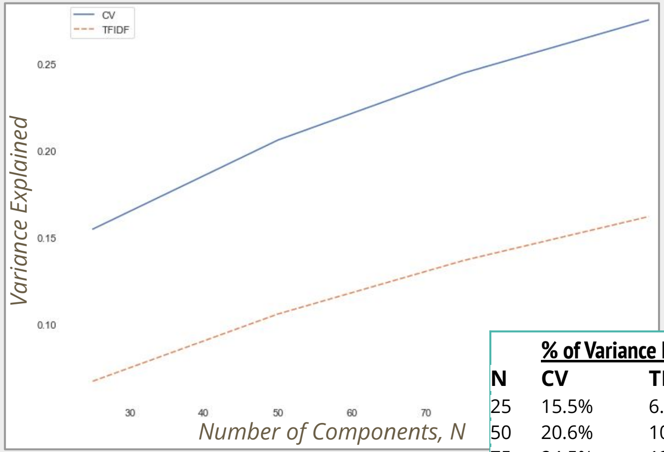
Produce array of the frequency of each unique word that occurs in the entire text

Output: sparse array

```
[[0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 ...
```

Truncated SVD

Linear dimensionality reduction of sparse matrix



	% of Variance Explained	
N	CV	TFIDF
25	15.5%	6.7%
50	20.6%	10.6%
75	24.5%	13.7%
100	27.5%	16.2%

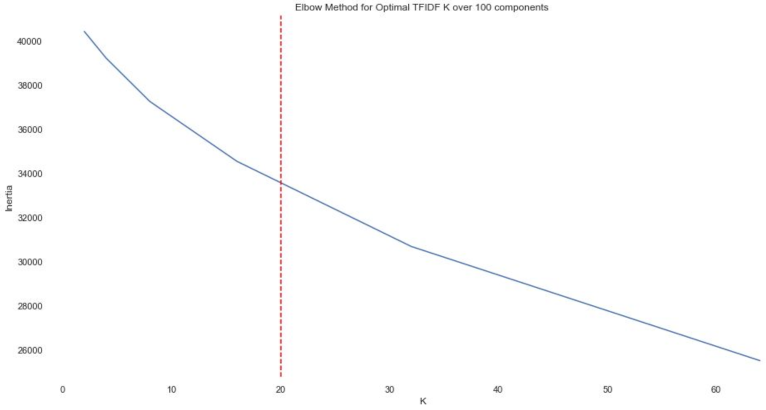
`alltext`

Feature Engineering

Cluster Transformed Text

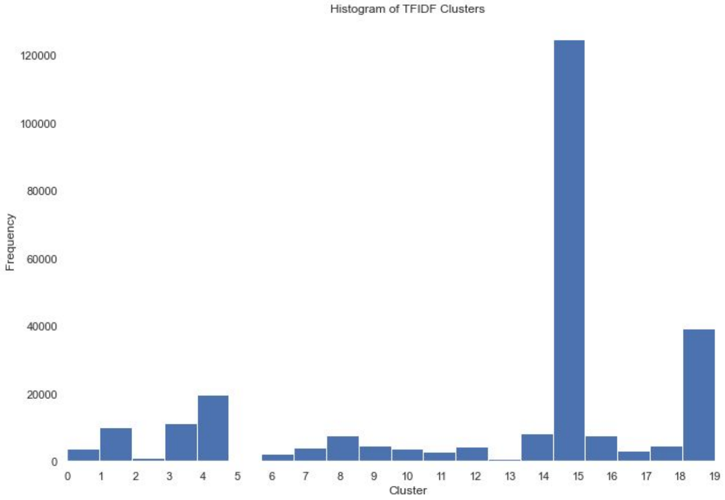
k-Means of count vectorization

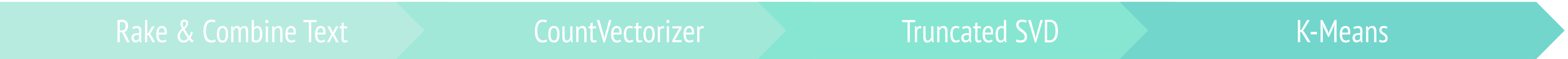
Use elbow method to determine the appropriate number of clusters, k



Use k = 20

Resulting cluster sizes *(k = 20)*





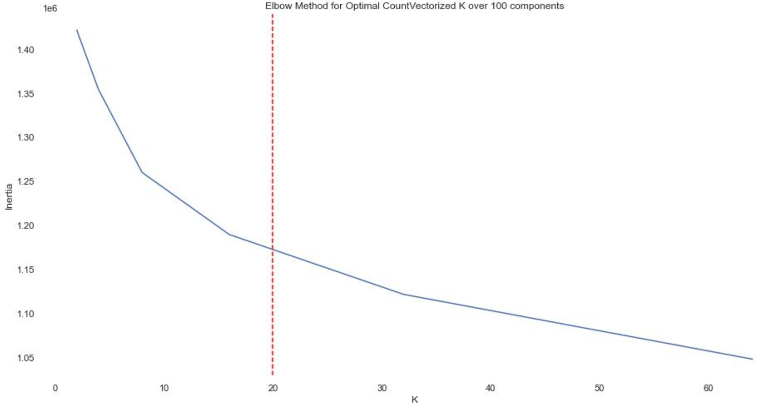
`alltext`

Feature Engineering

Cluster Transformed Text

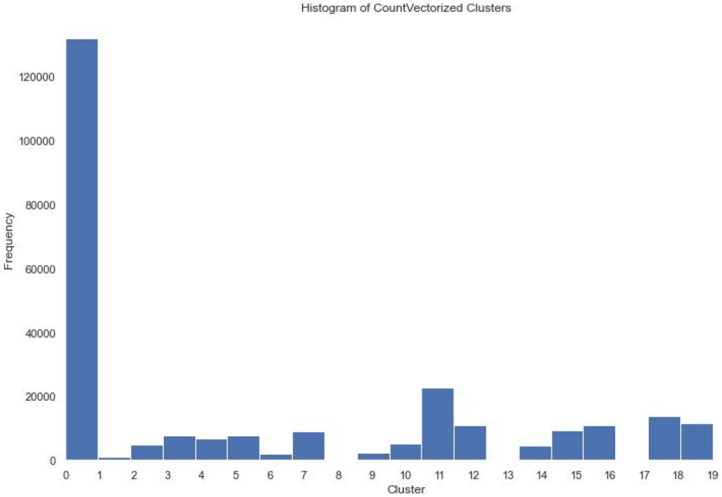
k-Means of TF-IDF vectorization

Use elbow method to determine the appropriate number of clusters, k



Use k = 20

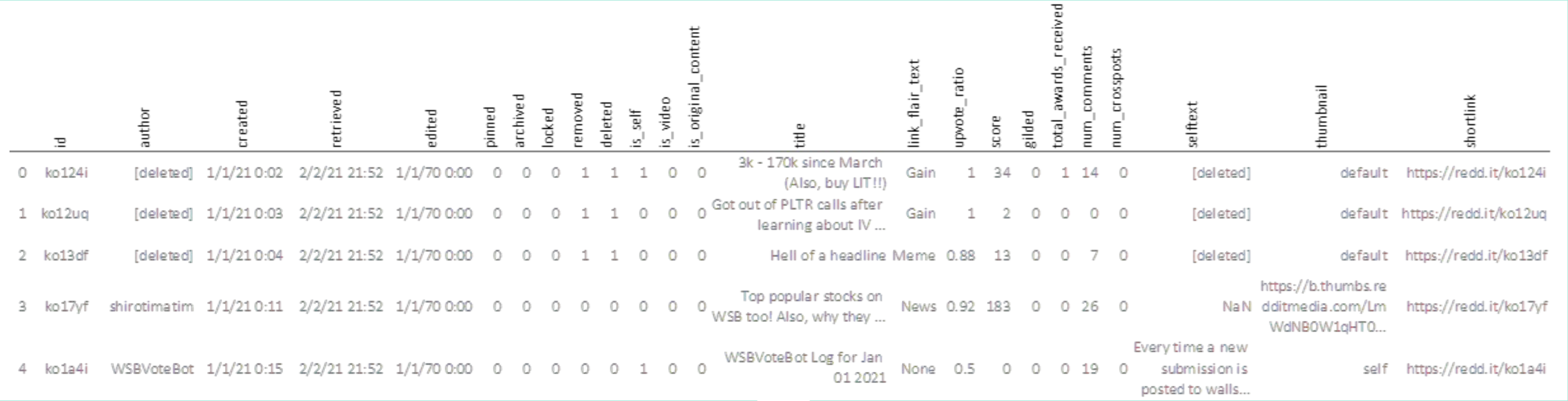
Resulting cluster sizes *(k = 20)*



Rake & Combine Text → TF-IDF → Truncated SVD → K-Means

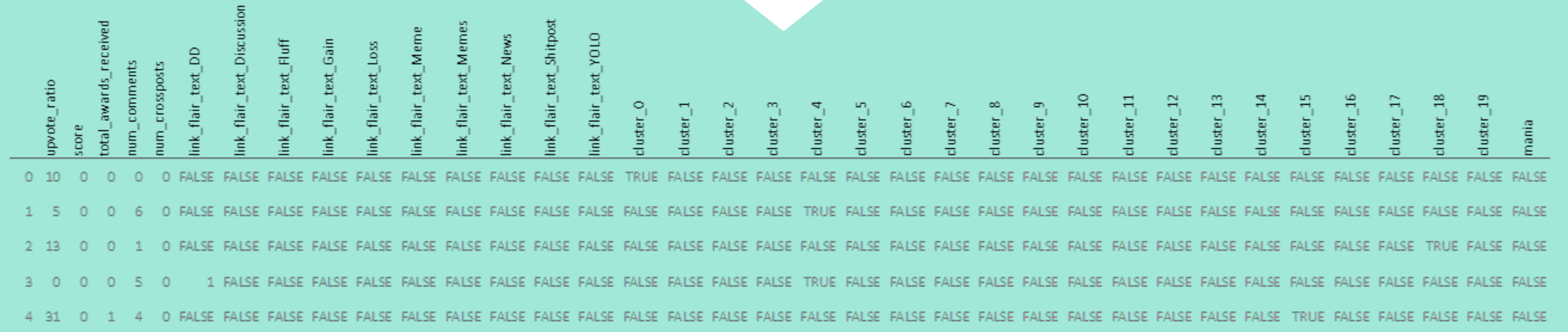
Modeling Data Frame

Raw Data Frame



	id	author	created	retrieved	edited	pinned	archived	locked	removed	deleted	is_self	is_video	is_original_content	title	link_flair_text	upvote_ratio	score	gilded	total_awards_received	num_comments	num_crossposts	selftext	thumbnail	shortlink
0	ko124i	[deleted]	1/1/21 0:02	2/2/21 21:52	1/1/70 0:00	0	0	0	1	1	1	0	0	3k - 170k since March (Also, buy LIT!!)	Gain	1	34	0	1	14	0	[deleted]	default	https://redd.it/ko124i
1	ko12uq	[deleted]	1/1/21 0:03	2/2/21 21:52	1/1/70 0:00	0	0	0	1	1	0	0	0	Got out of PLTR calls after learning about IV ...	Gain	1	2	0	0	0	0	[deleted]	default	https://redd.it/ko12uq
2	ko13df	[deleted]	1/1/21 0:04	2/2/21 21:52	1/1/70 0:00	0	0	0	1	1	0	0	0	Hell of a headline	Meme	0.88	13	0	0	7	0	[deleted]	default	https://redd.it/ko13df
3	ko17yf	shirotimatim	1/1/21 0:11	2/2/21 21:52	1/1/70 0:00	0	0	0	0	0	0	0	0	Top popular stocks on WSB too! Also, why they ...	News	0.92	183	0	0	26	0	NaN	https://b.thumbs.redditmedia.com/LmWdNB0W1qHT0...	https://redd.it/ko17yf
4	ko1a4i	WSBVoteBot	1/1/21 0:15	2/2/21 21:52	1/1/70 0:00	0	0	0	0	0	1	0	0	WSBVoteBot Log for Jan 01 2021	None	0.5	0	0	0	19	0	Every time a new submission is posted to walls...	self	https://redd.it/ko1a4i

Final Data Frame



	upvote_ratio	score	total_awards_received	num_comments	num_crossposts	link_flair_text_DD	link_flair_text_Discussion	link_flair_text_Fluff	link_flair_text_Gain	link_flair_text_Loss	link_flair_text_Meme	link_flair_text_Memes	link_flair_text_News	link_flair_text_Shitpost	link_flair_text_YOLO	cluster_0	cluster_1	cluster_2	cluster_3	cluster_4	cluster_5	cluster_6	cluster_7	cluster_8	cluster_9	cluster_10	cluster_11	cluster_12	cluster_13	cluster_14	cluster_15	cluster_16	cluster_17	cluster_18	cluster_19	mania
0	10	0	0	0	0	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
1	5	0	0	6	0	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
2	13	0	0	1	0	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE
3	0	0	0	5	0	1	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
4	31	0	1	4	0	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE

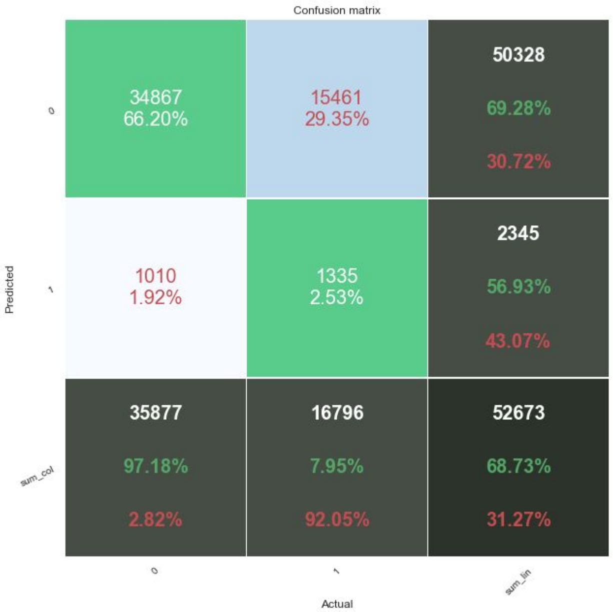
Reddit Financial Channel

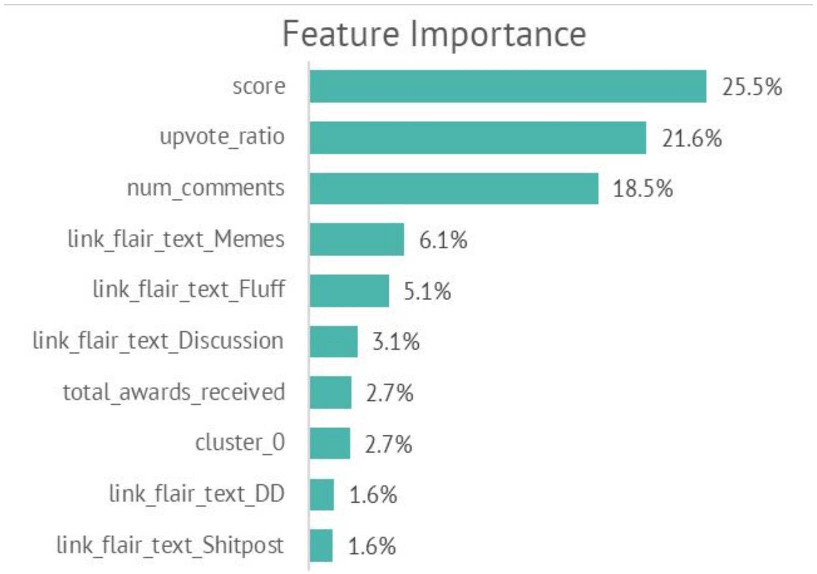
Modelling

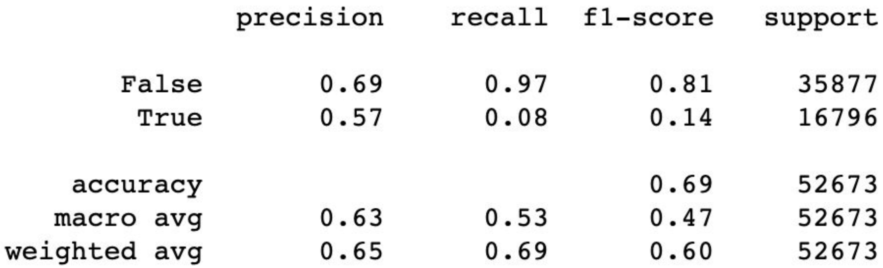
Modeling

Decision Tree Classifier

from CountVectorizer







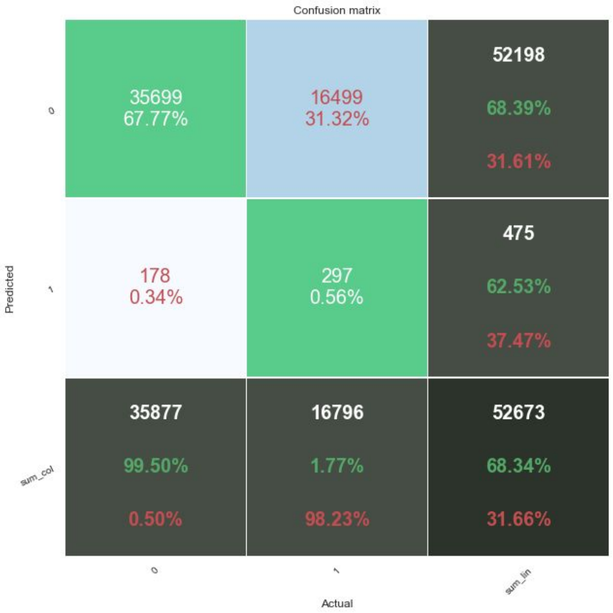
```
              precision    recall  f1-score   support

       False       0.69      0.97      0.81     35877
        True       0.57      0.08      0.14     16796

    accuracy                           0.69     52673
   macro avg       0.63      0.53      0.47     52673
weighted avg       0.65      0.69      0.60     52673
```

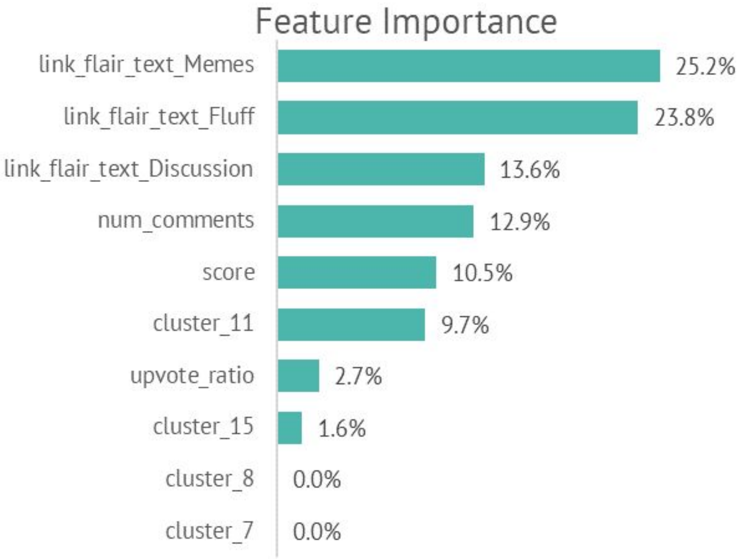
Rake & Combine Text › CountVectorizer › Truncated SVD › K-Means › Decision Tree

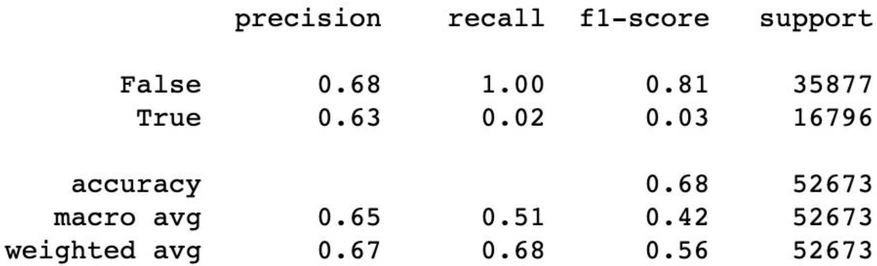
Modeling

Decision Tree Classifier

from TF-IDF







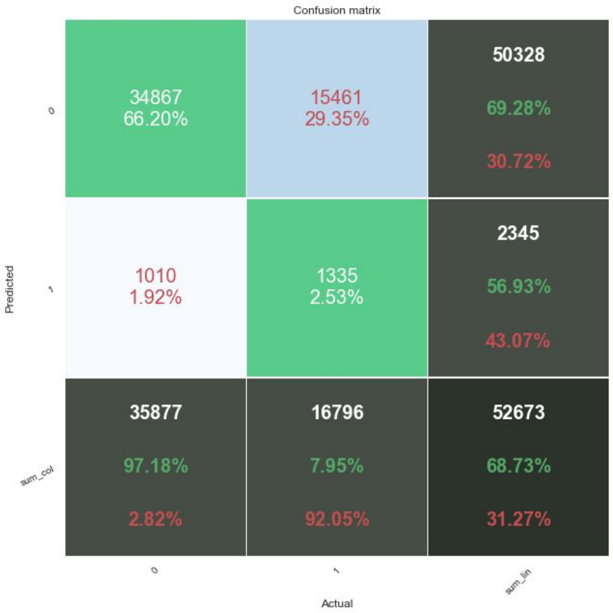
```
              precision    recall  f1-score   support

       False       0.68      1.00      0.81     35877
        True       0.63      0.02      0.03     16796

    accuracy                           0.68     52673
   macro avg       0.65      0.51      0.42     52673
weighted avg       0.67      0.68      0.56     52673
```

Rake & Combine Text → TF-IDF → Truncated SVD → K-Means → Decision Tree

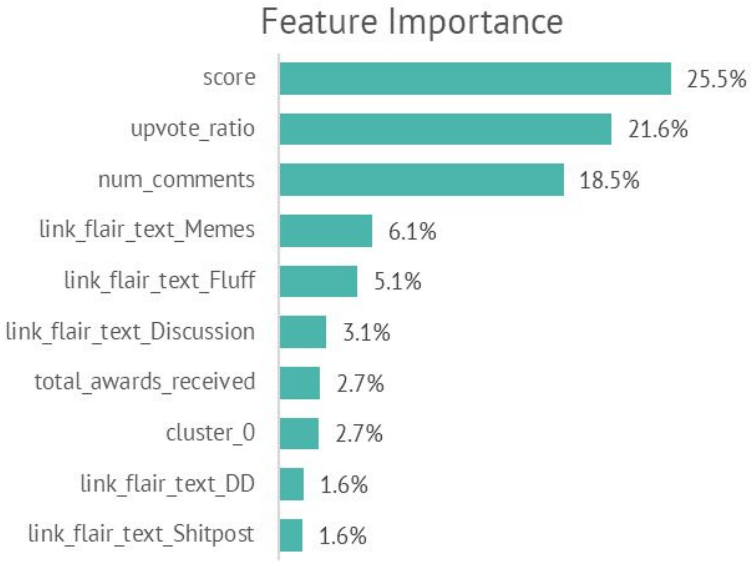
Modeling

Random Forest Classifier

from CountVectorizer


Confusion matrix

Predicted \ Actual	0	1	sum_lin
0	34867 66.20%	15461 29.35%	50328 69.28% 30.72%
1	1010 1.92%	1335 2.53%	2345 56.93% 43.07%
sum_col	35877 97.18% 2.82%	16796 7.95% 92.05%	52673 68.73% 31.27%

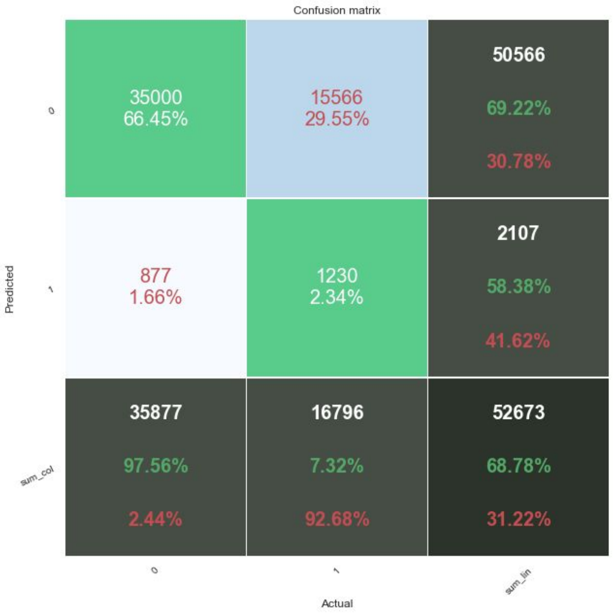

Feature Importance

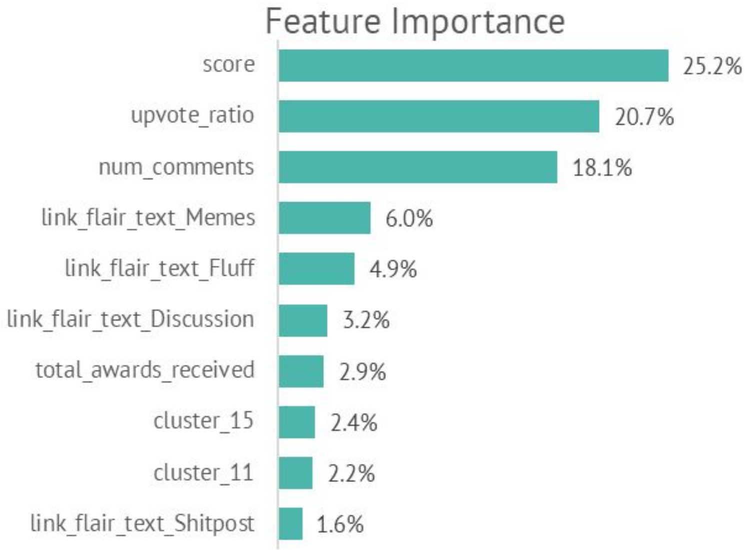
```
              precision    recall  f1-score   support

       False       0.69      0.97      0.81     35877
        True       0.57      0.08      0.14     16796

    accuracy                           0.69     52673
   macro avg       0.63      0.53      0.47     52673
weighted avg       0.65      0.69      0.60     52673
```

Modeling

Random Forest Classifier

from TF-IDF

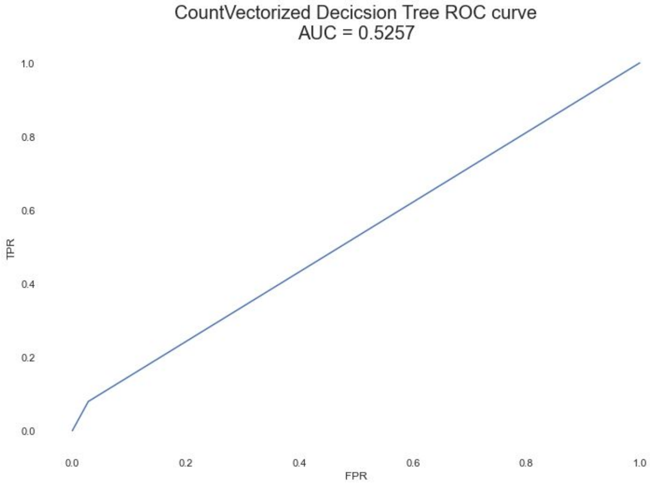


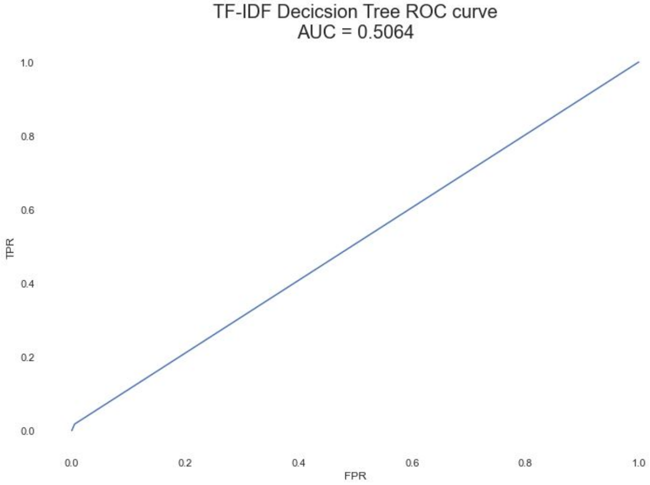


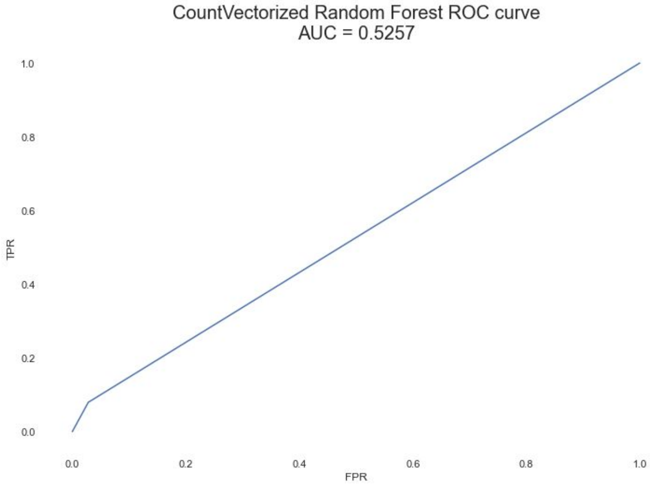
```
              precision    recall  f1-score   support

       False       0.69      0.98      0.81     35877
        True       0.58      0.07      0.13     16796

    accuracy                           0.69     52673
   macro avg       0.64      0.52      0.47     52673
weighted avg       0.66      0.69      0.59     52673
```

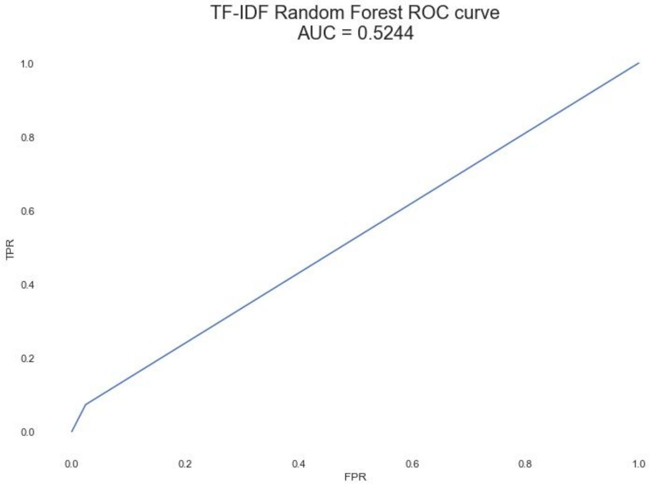
Results

Given a nearly straight ROC curve and AUC curve near 0.5, neither model (decision tree nor random forest) trained on count or tf-idf vectorized text features is able to strongly separate true samples from false ones. Despite a relatively high accuracy, the models perform hardly better than guessing along proportion of true class of target variable.


CountVectorized Decicsion Tree ROC curve
AUC = 0.5257


TF-IDF Decicsion Tree ROC curve
AUC = 0.5064


CountVectorized Random Forest ROC curve
AUC = 0.5257


TF-IDF Random Forest ROC curve
AUC = 0.5244

Takeaways

Key Findings

- **Text mined features were not the most significant factors**
- **Models marginally improved upon guessing forward volatility**

Challenges

- Large data
  - Over 1.3M documents
- Imbalanced data
  - Low proportion of volatile days
- Noisy data
  - Many irrelevant or spam documents
- Hardware & time constraints
  - DBSCAN and hierarchical clustering failed or crashed kernel

Next Steps

- Enhanced data cleaning
- Various target variables
- Other securities (i.e. AMC)
- Other models (CNN, etc.)
- More granular price data
- Scrape comments text
- Classify popular posts and then pipe only that classification into a volatility prediction model

Thank You