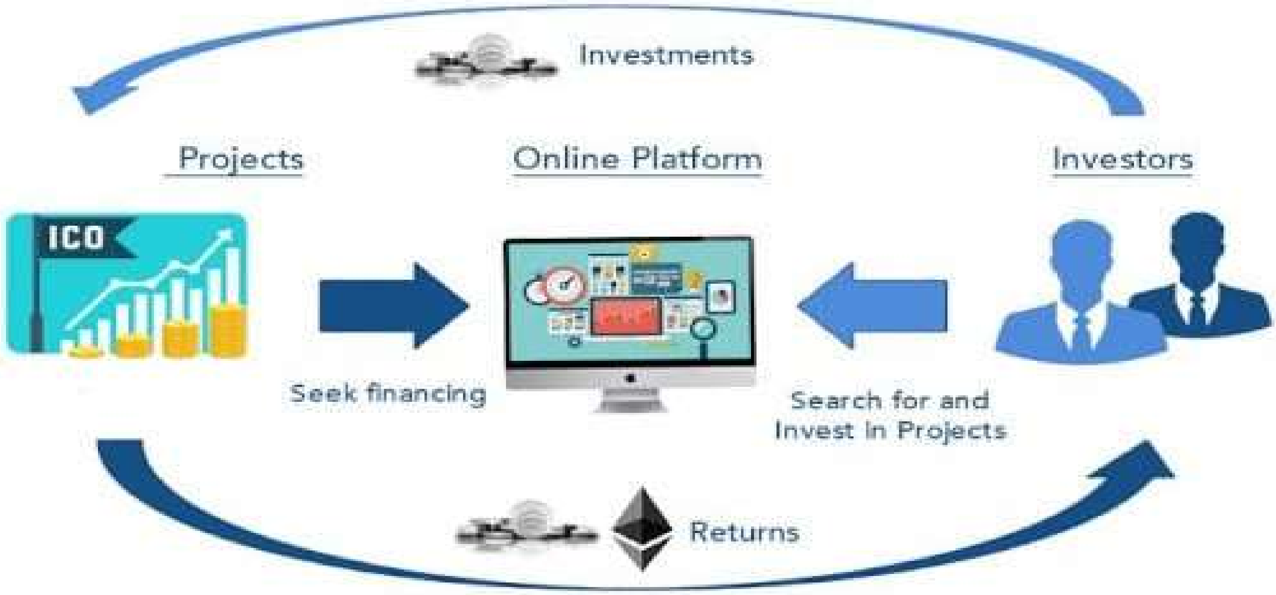
Introduction

Initial Coin Offering:

An initial coin offering (ICO) or initial currency offering is a type of funding using cryptocurrencies. It is often a form of crowdfunding, however, a private ICOs which does not seek public investment is also possible. In ICO, a quantity of cryptocurrency is sold in the form of "tokens" ("coins") to speculators or investors, in exchange for legal tender or other (generally established and more stable) cryptocurrencies such as Bitcoin or Ethereum. The tokens are promoted as future functional units of currency if or when the ICO's funding goal is met and the project successfully launches.

- Initial Coin Offerings (ICOs) are a popular fundraising method used primarily by startups wishing to offer products and services, usually related to the cryptocurrency and blockchain space.
- ICOs are similar to stocks, but they sometimes have utility for a software service or product offered.
- Some ICOs have yielded massive returns for investors. Numerous others have turned out to be fraud or have failed or performed poorly.
- To participate in an ICO, you will usually need to purchase a digital currency first and have a basic understanding of how to use cryptocurrency wallets and exchanges.
- ICOs are, for the most part, completely unregulated, so investors must exercise a high degree of caution and diligence when researching and investing in ICOs



Whitepaper:
A white paper is an authoritative report or guide that informs readers concisely about a complex issue and presents the issuing body's philosophy on the matter. It is meant to help readers understand an issue, solve a problem, or make a decision."

What Is An ICO Scam?

Due to the lucrative returns investors have been making in the cryptocurrency world, there has been an enormous rise in the number of people looking to invest in ICO's. Unfortunately, what comes with this surge in popularity, is scammers looking to benefit from vulnerable investors.

Before we get into outlining how to tell if an ICO is fake, we must define the word scam.

Scam

"Any project that expressed availability of ICO investment (through a website publishing, ANN thread, or social media posting with a contribution address), did not have/had no intention of fulfilling project development duties with the funds, and/or was deemed by the community (message boards, website or other online information) to be a scam."

A dishonest project carried out by untrustworthy people, that is completed in an attempt to steal something of value from another person.

Using this definition, one could make an assumption that an ICO is deemed a scam if:

- The ICO has a lack of trust – for example, the team appears fake or non-existent
- The ICO comes across as a dishonest project – for example, the sole purpose of the ICO is to steal an investors money

Motivation

Absence Of ICO Regulatory Oversight

One of the recent studies states that last year 81% of ICO’s were scams, this highlights the actual fact of how rampant ICO scams currently are. The main reason for this is the lack of rules or regulations in place to identify a level of legitimacy amongst ICO’s.

Some countries, however, have begun to tackle this issue and started creating or ideating rules and regulations to cover the space. The United States is a clear example of how ICO regulation is changing within their country.

More and more ICO projects are being proposed and so it makes the task of researching about each project difficult and time consuming for the experts as there is so much information to verify to decide if the ICO project is worth investing in and is legitimate and not a scam.

Problem Statement

To classify ICO ratings based on text extracted from Whitepapers and its description.

How an ICO coin is rated?

- ➔ The ICO project is analyzed by an expert to check for various information published by the ICO project to determine the rating of the ICO project. (Lower rating means the project cannot be trusted and it's risky to invest into it and vice versa).
- ➔ The experts go through the information published in the Whitepaper of the ICO project(It's a red flag if there is no whitepaper published) like
    - ◆ What problem the project is trying to solve?
    - ◆ Who are the founding team members behind the project? What is their vision about the project? What is their background in the field of cryptocurrency and blockchain?
    - ◆ What technology are they using to implement their vision and how they are going to implement it?
    - ◆ What is their sales and marketing strategy?
    - ◆ Are they applying any KYC/whitelist solution and other security solutions to prevent any fraudulent activity
    - ◆ What is the future scope of the project?
    - ◆ Are they making any unrealistic claims which cannot be fulfilled?

Goal

Our goal is to create a machine learning model to help the experts in the rating of the ICO projects by analyzing the text of whitepapers and other details of the ICO project to gather insights about the ICO project to make the task easy and less time consuming for the expert to determine the ratings.

Methodology

- **Data Collection**
- **EDA & Data Preprocessing**
- **Modeling**
- **Clustering**
- **Results**

Data Collection

Data Source

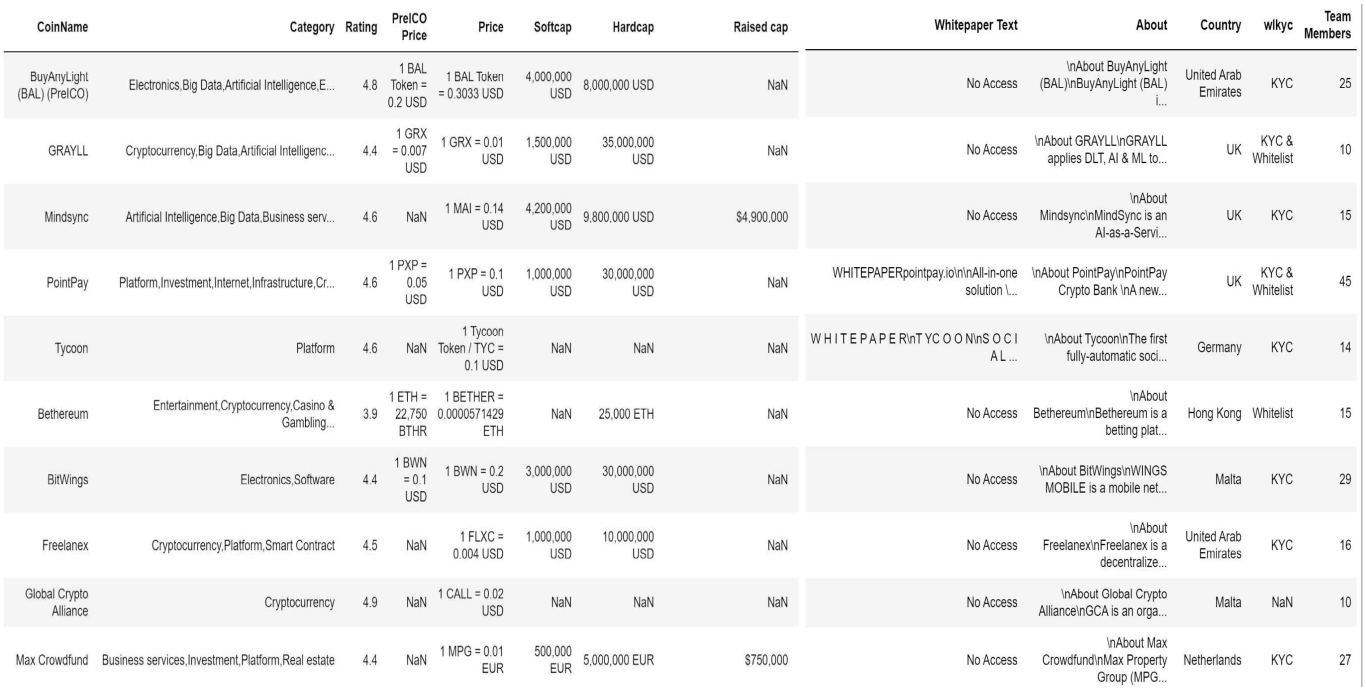
We collected the data about the ICO projects from www.icobench.com

Scraping

We used Beautifulsoup and web drivers to scrape data from the website.

We scraped various details about each ICO coin listed on icobench.com like ICO Name, About text description , Token name, Token type, Category to which the ICO project belongs , PreICO Price, Current Price, Soft Cap, Hard Cap, Platform , Country, does it implement Whitelist/KYC solutions , Countries where the coin is restricted, Total number of team members, Coin Rating as given by ico-bench and last but not least the whitepaper text for each coin if available using the links given on the website.

We were able to scrape data for around 5700 coins. Below is the sample of data we collected.

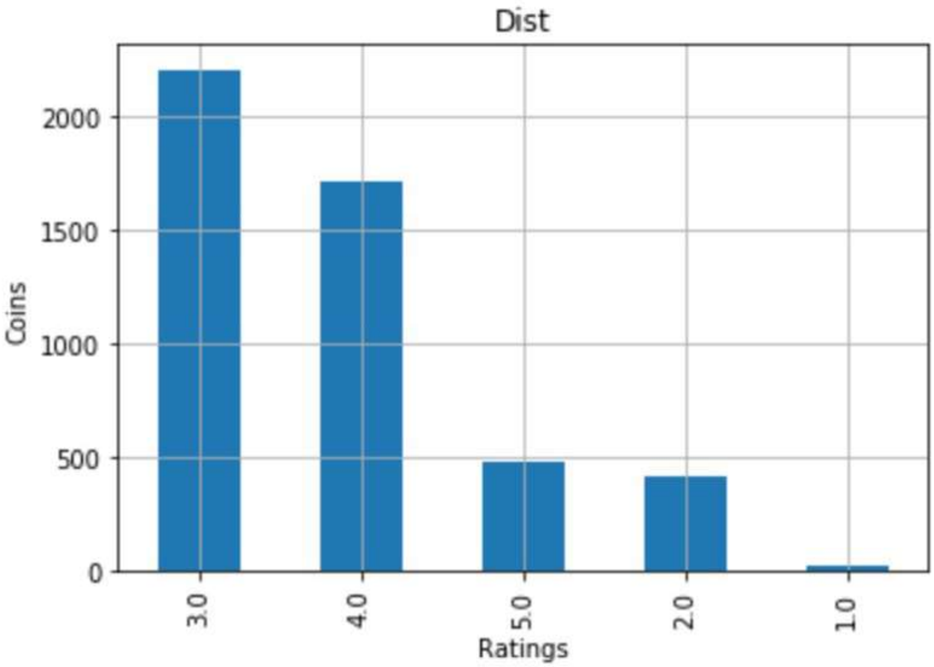


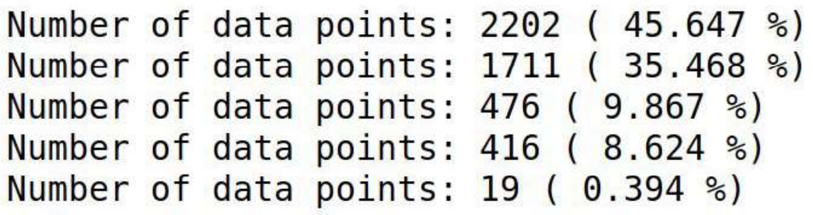
CoinName	Category	Rating	PreICO Price	Price	Softcap	Hardcap	Raised cap	Whitepaper Text	About	Country	wlkyc	Team Members
BuyAnyLight (BAL) (PreICO)	Electronics,Big Data,Artificial Intelligence,E...	4.8	1 BAL Token = 0.2 USD	1 BAL Token = 0.3033 USD	4,000,000 USD	8,000,000 USD	NaN	No Access	\nAbout BuyAnyLight (BAL)\nBuyAnyLight (BAL) i...	United Arab Emirates	KYC	25
GRAYLL	Cryptocurrency,Big Data,Artificial Intelligenc...	4.4	1 GRX = 0.007 USD	1 GRX = 0.01 USD	1,500,000 USD	35,000,000 USD	NaN	No Access	\nAbout GRAYLL\nGRAYLL applies DLT, AI & ML to...	UK	KYC & Whitelist	10
Mindsync	Artificial Intelligence,Big Data,Business serv...	4.6	NaN	1 MAI = 0.14 USD	4,200,000 USD	9,800,000 USD	$4,900,000	No Access	\nAbout Mindsync\nMindSync is an AI-as-a-Servi...	UK	KYC	15
PointPay	Platform,Investment,Internet,Infrastructure,Cr...	4.6	1 PXP = 0.05 USD	1 PXP = 0.1 USD	1,000,000 USD	30,000,000 USD	NaN	WHITEPAPERpointpay.io\n\nAll-in-one solution \...	\nAbout PointPay\nPointPay Crypto Bank \nA new...	UK	KYC & Whitelist	45
Tycoon	Platform	4.6	NaN	1 Tycoon Token / TYC = 0.1 USD	NaN	NaN	NaN	W H I T E P A P E R\nT YC O O N\nS O C I A L ...	\nAbout Tycoon\nThe first fully-automatic soci...	Germany	KYC	14
Bethereum	Entertainment,Cryptocurrency,Casino & Gambling...	3.9	1 ETH = 22,750 BTHR	1 BETHER = 0.0000571429 ETH	NaN	25,000 ETH	NaN	No Access	\nAbout Bethereum\nBethereum is a betting plat...	Hong Kong	Whitelist	15
BitWings	Electronics,Software	4.4	1 BWN = 0.1 USD	1 BWN = 0.2 USD	3,000,000 USD	30,000,000 USD	NaN	No Access	\nAbout BitWings\nWINGS MOBILE is a mobile net...	Malta	KYC	29
Freelanex	Cryptocurrency,Platform,Smart Contract	4.5	NaN	1 FLXC = 0.004 USD	1,000,000 USD	10,000,000 USD	NaN	No Access	\nAbout Freelanex\nFreelanex is a decentralize...	United Arab Emirates	KYC	16
Global Crypto Alliance	Cryptocurrency	4.9	NaN	1 CALL = 0.02 USD	NaN	NaN	NaN	No Access	\nAbout Global Crypto Alliance\nGCA is an orga...	Malta	NaN	10
Max Crowdfund	Business services,Investment,Platform,Real estate	4.4	NaN	1 MPG = 0.01 EUR	500,000 EUR	5,000,000 EUR	$750,000	No Access	\nAbout Max Crowdfund\nMax Property Group (MPG...	Netherlands	KYC	27

EDA

We analyzed all the features by calculating number of null values, unique values within each feature. At first, we removed all the duplicate values from the dataset then we also removed unnecessary features like Token type. We also removed features having 1000 or more null values like PreICO price, Softcap, Hardcap. The Whitepapers are the most crucial part of an ICO project. It describes how the crowdfunding is intended to work, such as the landscape of the ICO project, how the tokens will be allocated, how the crowd-funded money will be spent. Out of around 5700 ICO project/coins, we were able to scrape and transform white paper pdfs for around 1100 white papers. For all the coins with no whitepaper, we substitute it with ‘About’ description of the corresponding coin. After removing all the null values we have 4824 coins.

For the target variable **‘Rating’** we converted and round each value to categorize it into **1 to 5** ratings. Distribution of Ratings among ICO coins is shown below:





```
Number of data points: 2202 ( 45.647 %)
Number of data points: 1711 ( 35.468 %)
Number of data points: 476 ( 9.867 %)
Number of data points: 416 ( 8.624 %)
Number of data points: 19 ( 0.394 %)
```

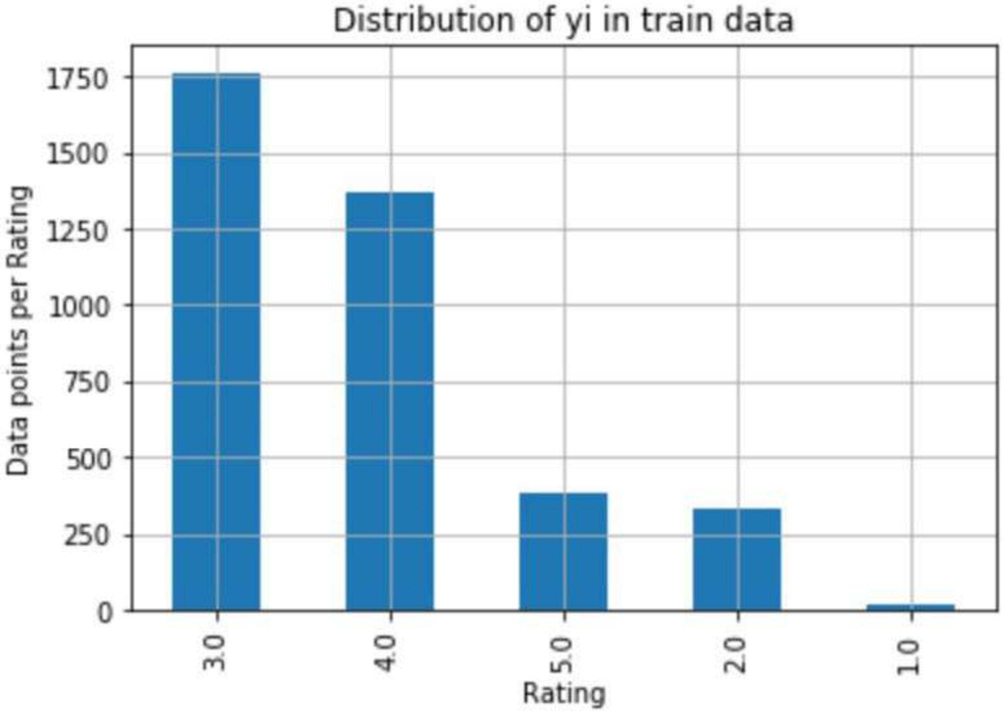
Text Preprocessing

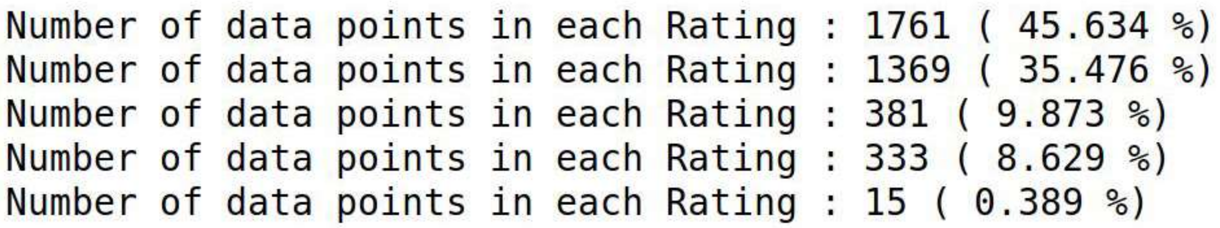
For cleaning the text(About and Whitepaper), we first removed non-alphanumeric text(except “‘ , - ._” symbols) using regular expressions and then we removed stop words from the text.

Then we tokenized text to create word and sentence tokens.

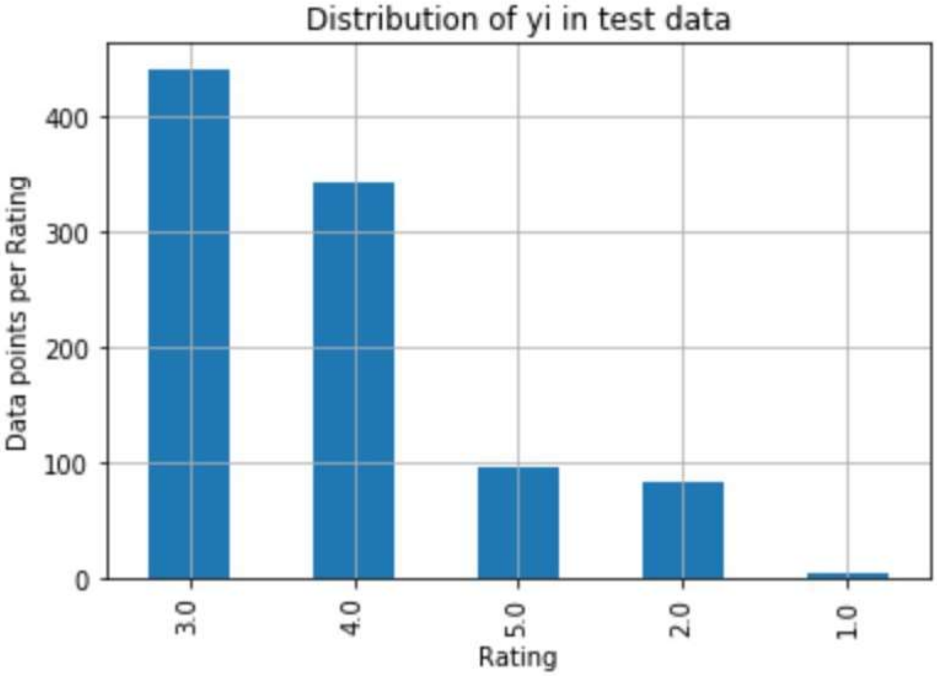
Modeling

Before modeling, we split data into train and test with the distribution of 80% and 20%. As our dataset is imbalanced with respect to target variable ‘Rating’, we used stratified train-test split so we get a similar distribution of data among train set and test set. The figure below describes train data and test data distributions.





```
Number of data points in each Rating : 1761 ( 45.634 %)
Number of data points in each Rating : 1369 ( 35.476 %)
Number of data points in each Rating : 381 ( 9.873 %)
Number of data points in each Rating : 333 ( 8.629 %)
Number of data points in each Rating : 15 ( 0.389 %)
```



```
Number of data points in each Rating : 441 ( 45.699 %)
Number of data points in each Rating : 342 ( 35.44 %)
Number of data points in each Rating : 95 ( 9.845 %)
Number of data points in each Rating : 83 ( 8.601 %)
Number of data points in each Rating : 4 ( 0.415 %)
```

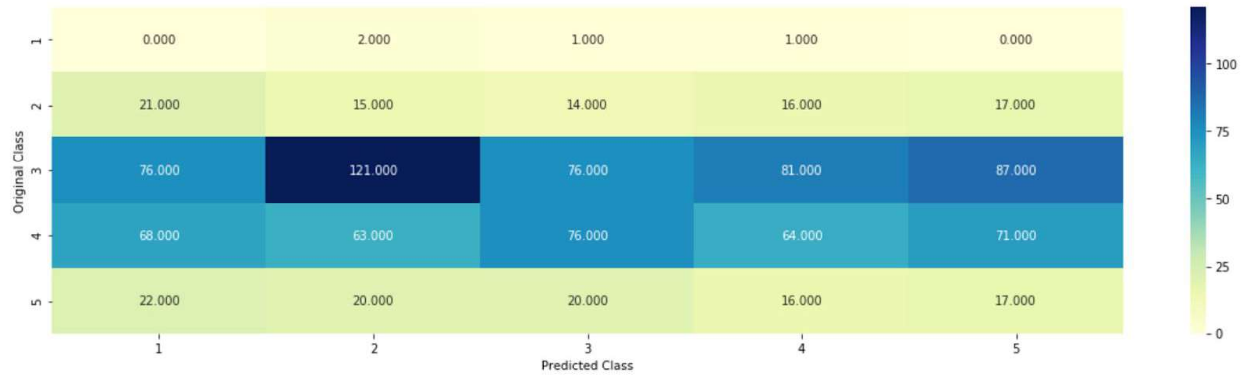
As we can see the distribution of ratings in the data is pretty imbalanced.

Random Model

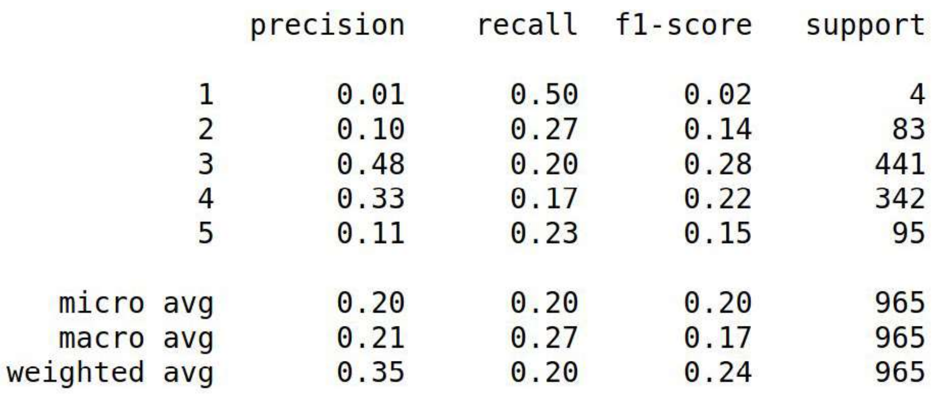
Before we apply the model to our data, we need to have some randomly predicted results. On the basis of the randomly predicted results, we can evaluate the model performance of other models like Logistic Regression, Naive Bayes and others.

We generated random ‘rating’ values between 1 to 5 for each ICO coin and classified all the coins. Below are the results of random model:

```
Confusion matrix --------------------
```



Original Class \ Predicted Class	1	2	3	4	5
1	0.000	2.000	1.000	1.000	0.000
2	21.000	15.000	14.000	16.000	17.000
3	76.000	121.000	76.000	81.000	87.000
4	68.000	63.000	76.000	64.000	71.000
5	22.000	20.000	20.000	16.000	17.000



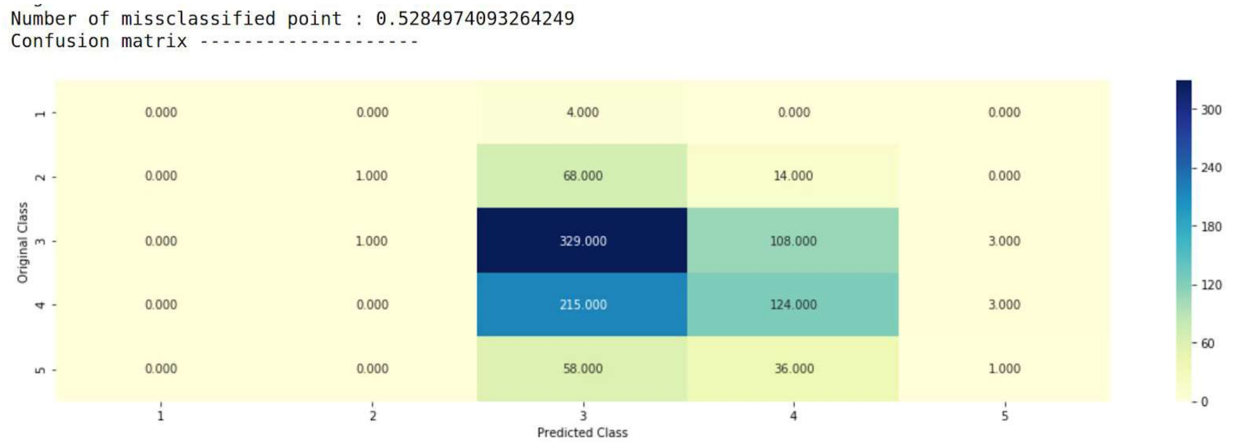
```
             precision    recall  f1-score   support

           1       0.01      0.50      0.02         4
           2       0.10      0.27      0.14        83
           3       0.48      0.20      0.28       441
           4       0.33      0.17      0.22       342
           5       0.11      0.23      0.15        95

   micro avg       0.20      0.20      0.20       965
   macro avg       0.21      0.27      0.17       965
weighted avg       0.35      0.20      0.24       965
```

As we took Tf-IDF weights, many features were created and when there are so many features and ours is a classification problem, therefore, we thought of applying logistic regression and Multinomial Naive Bayes models.

Classification

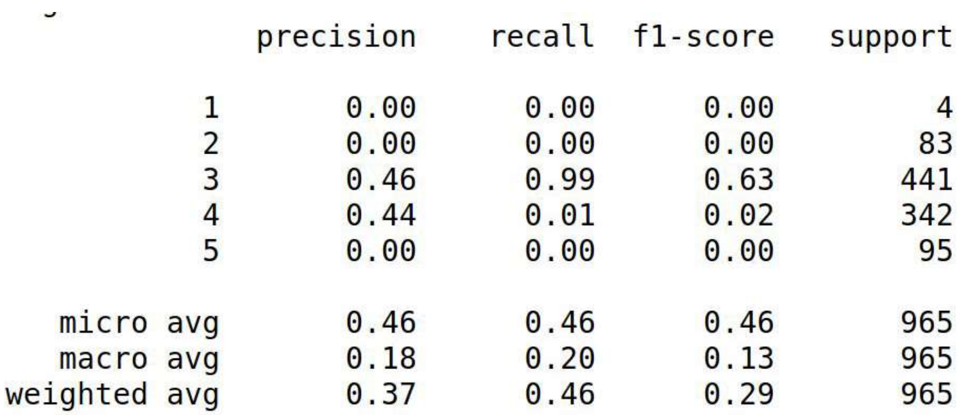
Logistic Regression

We vectorized the word tokens of whitepaper text to generate Tf-IDF weights.
We used Hyperparameter tuning to get the best Alpha value to train the logistic regression model.
Then we trained a logistic regression model using these Tf-If weights on train dataset and after applying it to test dataset we got the below result.



```
Number of missclassified point : 0.5284974093264249
Confusion matrix --------------------
```

Original Class \ Predicted Class	1	2	3	4	5
1	0.000	0.000	4.000	0.000	0.000
2	0.000	1.000	68.000	14.000	0.000
3	0.000	1.000	329.000	108.000	3.000
4	0.000	0.000	215.000	124.000	3.000
5	0.000	0.000	58.000	36.000	1.000



```
               precision    recall  f1-score   support

           1       0.00      0.00      0.00         4
           2       0.00      0.00      0.00        83
           3       0.46      0.99      0.63       441
           4       0.44      0.01      0.02       342
           5       0.00      0.00      0.00        95

   micro avg       0.46      0.46      0.46       965
   macro avg       0.18      0.20      0.13       965
weighted avg       0.37      0.46      0.29       965
```

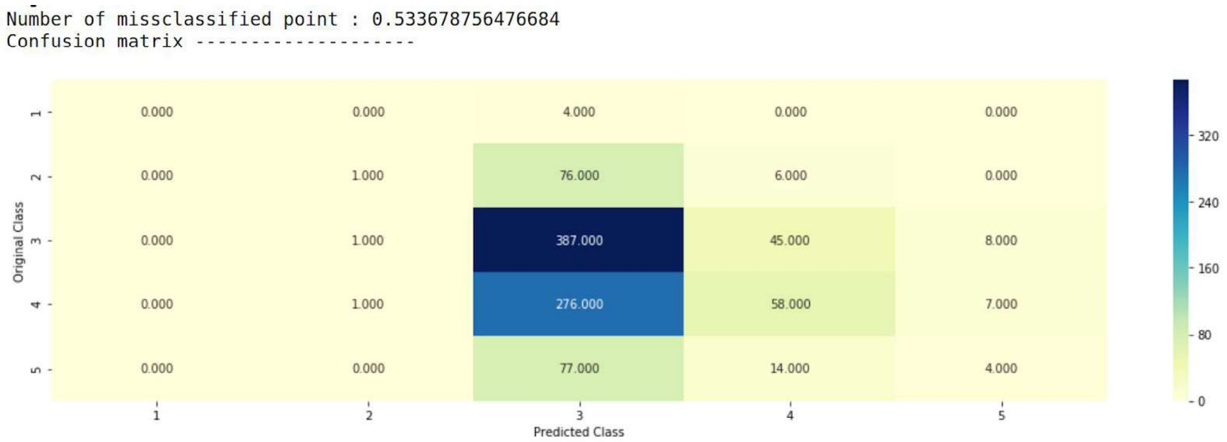
This above result is not appropriate as the f1 score is very less and it gives an **accuracy** of only **48%** and as shown in the confusion matrix, it does not predict any of the coins with rating = 1 and very few for rating = 2.

Naive Bayes

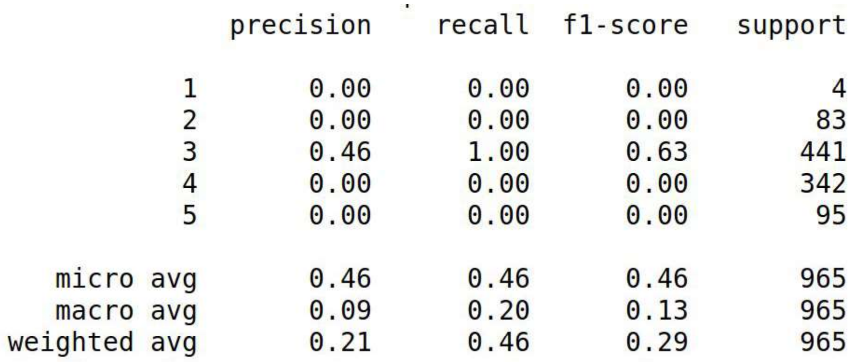
We vectorized the word tokens of whitepaper text to generate Tf-IDF weights.

We used Hyperparameter tuning to get the best Alpha value to train the logistic regression model.

We trained the Multinomial Naive Bayes model using the train dataset and then applied the model on test data to get below result.



```
Number of missclassified point : 0.533678756476684
Confusion matrix --------------------
```



```
              precision    recall  f1-score   support

           1       0.00      0.00      0.00         4
           2       0.00      0.00      0.00        83
           3       0.46      1.00      0.63       441
           4       0.00      0.00      0.00       342
           5       0.00      0.00      0.00        95

   micro avg       0.46      0.46      0.46       965
   macro avg       0.09      0.20      0.13       965
weighted avg       0.21      0.46      0.29       965
```

The above result performed with **46.7 % accuracy and** also the f1 score is very less and even zero for some class so it performs poorly similar to logistic regression.

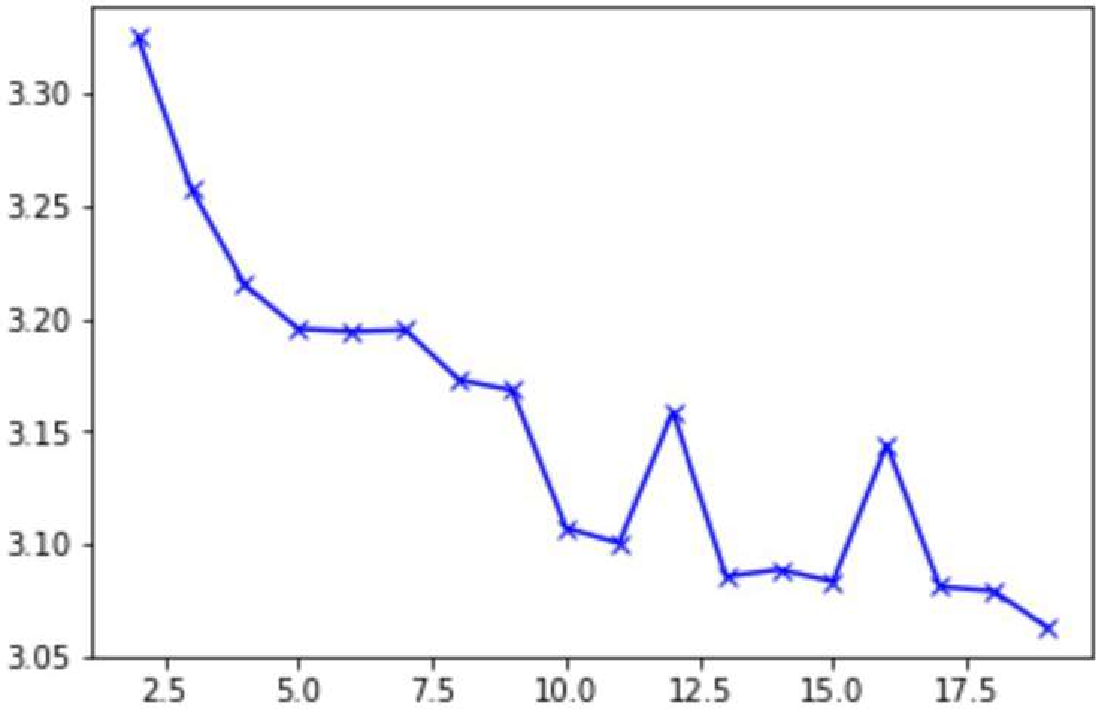
After trying classification models we thought of applying clustering models to see if the whitepaper text of similarly rated coins is grouping into the same clusters or not.

Clustering

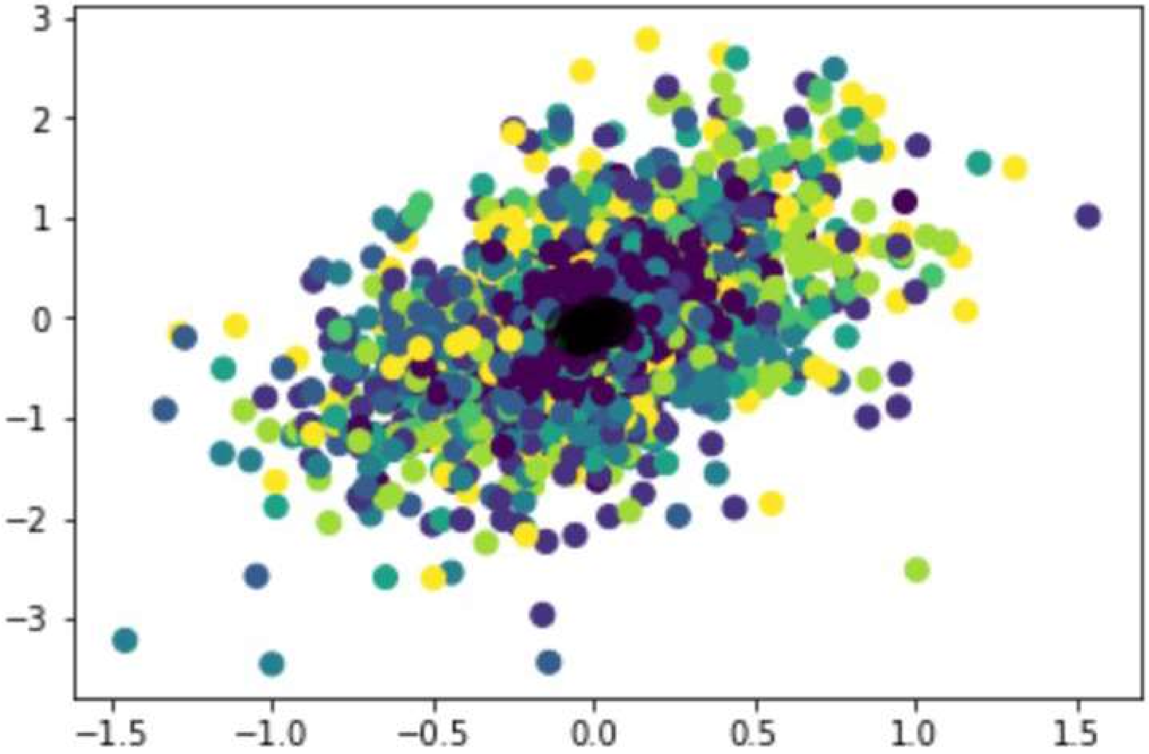
We trained the word2vec model using the whitepaper word tokens with a size of 300(each word gets converted to 1 x 300 dimension) and word occurring less than 5 times were ignored. Then we extracted only the “Noun” vectors using POS(Part-Of-Speech) tagging using spaCy. We got about 20,000 unique word vectors and we applied various clustering methods to cluster each Noun word present in Whitepaper text.

K-means

We applied Kmeans clustering to the Noun vectors for a number of clusters K = 2 to K = 19 and calculated silhouette score and plotted to apply elbow method to decide the optimal number of clusters for K-means clustering.



As we can see in the above graph the elbow is created when clustering number K = 4, K = 9. We decided to go with K = 9 and applied Kmeans clustering to get below the distribution of clusters.

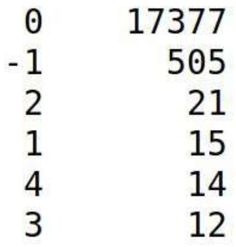


We can see that the clusters are overlapping and most of them have the centroid(Black color dots) in the middle of the distribution. Thus we are unable to get proper insights from applying K-means clustering.

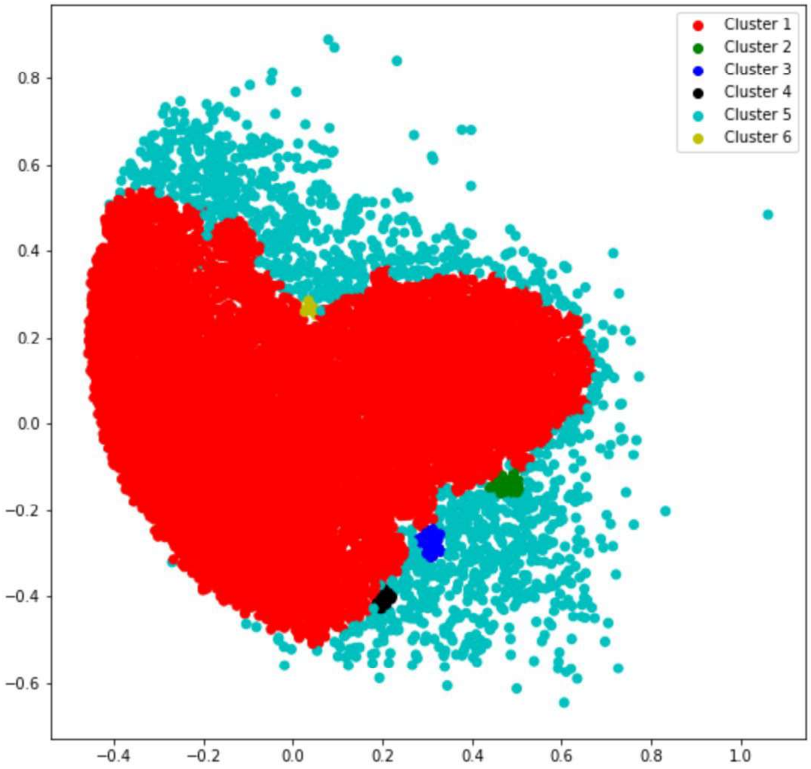
Maybe we can try to train Noun chunks or adjective + noun chunks to create vectors and then apply K-means clustering to hopefully get better results and insights.

DBSCAN

We also tried to cluster all the unique word vectors. With epsilon set to 0.03 and min_samples to 10. The algorithm created 6 clusters and assigned each data point to specific cluster.



```
 0    17377
-1      505
 2       21
 1       15
 4       14
 3       12
```

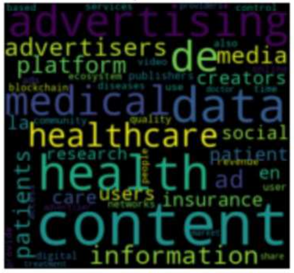


From the above plot, we can conclude that clusters 0 has almost all the data points assigned to it, which represents that this algorithm did not perform well with the data.

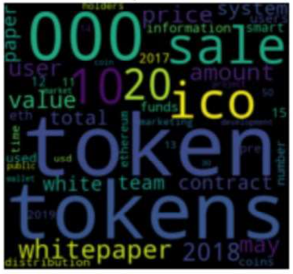
Topic Modeling with LDA

We also tried topic modeling with LDA to categorize white paper text into 10 topics. We used CountVectorizer to convert text data into vectors with max_df = 0.9 and min_df = 5. We also removed stopwords from the text. LDA performed somewhat well in the topic assignment as we can see 10 different categories from the results below.

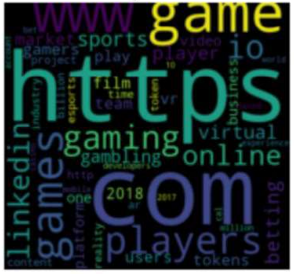
Topic: 0



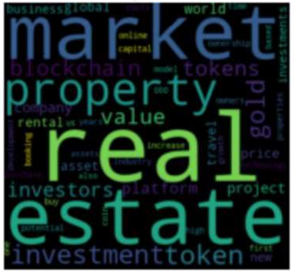
Topic: 1



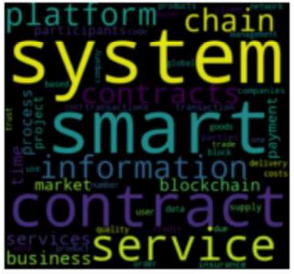
Topic: 2



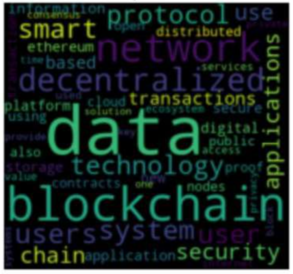
Topic: 3



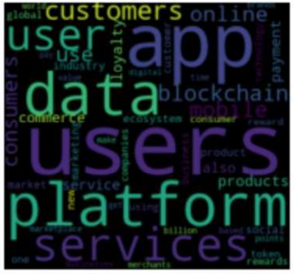
Topic: 4



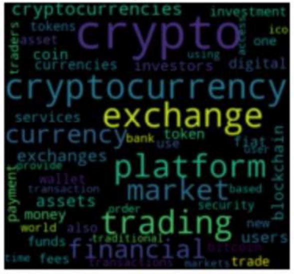
Topic: 5



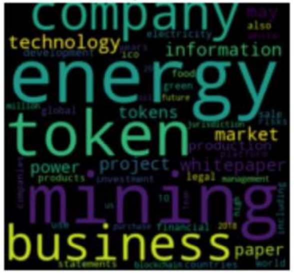
Topic: 6



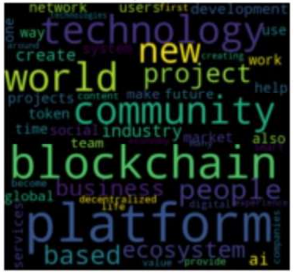
Topic: 7



Topic: 8



Topic: 9

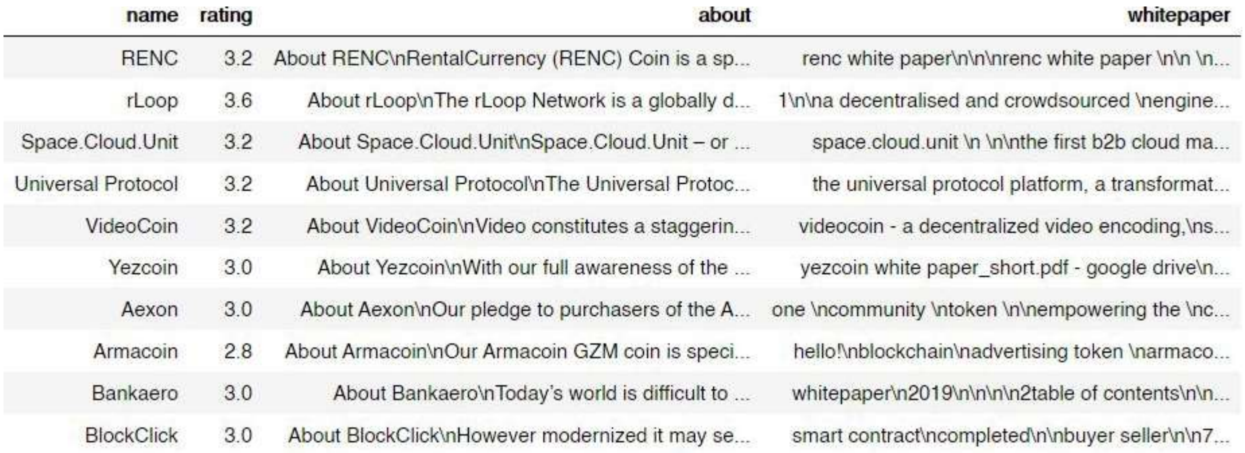


Methodology: convolutional neural network

algorithm: we tried to use a convolutional neural network to analyze the white papers

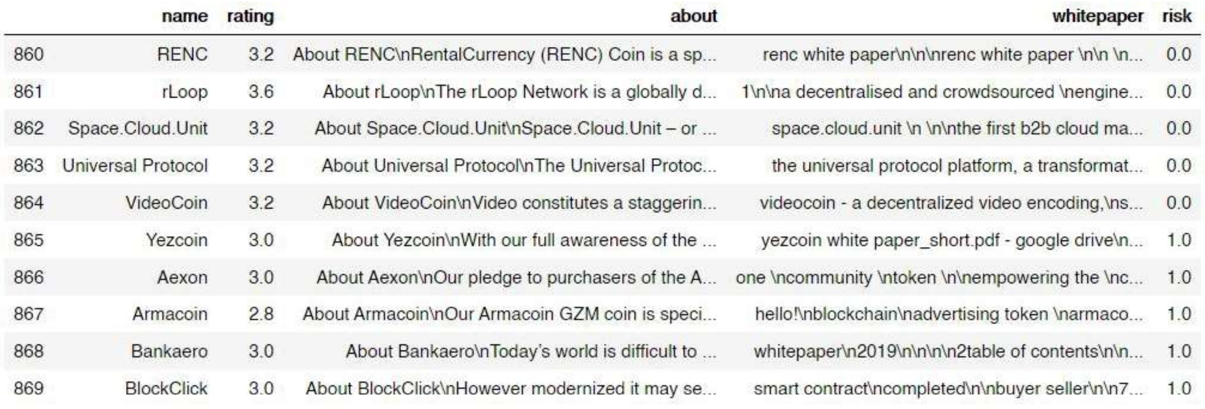
Input: Dataframe consisting of the following columns: Coin name, Rating, about and whitepaper.

The coins that did not have a whitepaper were dropped from the data frame By analyzing it was found that the coins consisting of ratings less than 3 were poorly documented and had much-missing information.



name	rating	about	whitepaper
RENC	3.2	About RENC\nRentalCurrency (RENC) Coin is a sp...	renc white paper\n\n\nrenc white paper \n\n \n...
rLoop	3.6	About rLoop\nThe rLoop Network is a globally d...	1\n\na decentralised and crowdsourced \nengine...
Space.Cloud.Unit	3.2	About Space.Cloud.Unit\nSpace.Cloud.Unit – or ...	space.cloud.unit \n \n\nthe first b2b cloud ma...
Universal Protocol	3.2	About Universal Protocol\nThe Universal Protoc...	the universal protocol platform, a transformat...
VideoCoin	3.2	About VideoCoin\nVideo constitutes a staggerin...	videocoin - a decentralized video encoding,\ns...
Yezcoin	3.0	About Yezcoin\nWith our full awareness of the ...	yezcoin white paper_short.pdf - google drive\n...
Aexon	3.0	About Aexon\nOur pledge to purchasers of the A...	one \ncommunity \ntoken \n\nempowering the \nc...
Armacoin	2.8	About Armacoin\nOur Armacoin GZM coin is speci...	hello!\nblockchain\nadvertising token \narmaco...
Bankaero	3.0	About Bankaero\nToday's world is difficult to ...	whitepaper\n2019\n\n\n\n2table of contents\n\n...
BlockClick	3.0	About BlockClick\nHowever modernized it may se...	smart contract\ncompleted\n\nbuyer seller\n\n7...

Therefore a column called risk was created for each coin, and given value 0 if the rating of the coin is greater than 3 and value 1 of the coin rating is 3 or lesser.



	name	rating	about	whitepaper	risk
860	RENC	3.2	About RENC\nRentalCurrency (RENC) Coin is a sp...	renc white paper\n\n\nrenc white paper \n\n \n...	0.0
861	rLoop	3.6	About rLoop\nThe rLoop Network is a globally d...	1\n\na decentralised and crowdsourced \nengine...	0.0
862	Space.Cloud.Unit	3.2	About Space.Cloud.Unit\nSpace.Cloud.Unit – or ...	space.cloud.unit \n \n\nthe first b2b cloud ma...	0.0
863	Universal Protocol	3.2	About Universal Protocol\nThe Universal Protoc...	the universal protocol platform, a transformat...	0.0
864	VideoCoin	3.2	About VideoCoin\nVideo constitutes a staggerin...	videocoin - a decentralized video encoding,\ns...	0.0
865	Yezcoin	3.0	About Yezcoin\nWith our full awareness of the ...	yezcoin white paper_short.pdf - google drive\n...	1.0
866	Aexon	3.0	About Aexon\nOur pledge to purchasers of the A...	one \ncommunity \ntoken \n\nempowering the \nc...	1.0
867	Armacoin	2.8	About Armacoin\nOur Armacoin GZM coin is speci...	hello!\nblockchain\nadvertising token \narmaco...	1.0
868	Bankaero	3.0	About Bankaero\nToday's world is difficult to ...	whitepaper\n2019\n\n\n\n2table of contents\n\n...	1.0
869	BlockClick	3.0	About BlockClick\nHowever modernized it may se...	smart contract\ncompleted\n\nbuyer seller\n\n7...	1.0

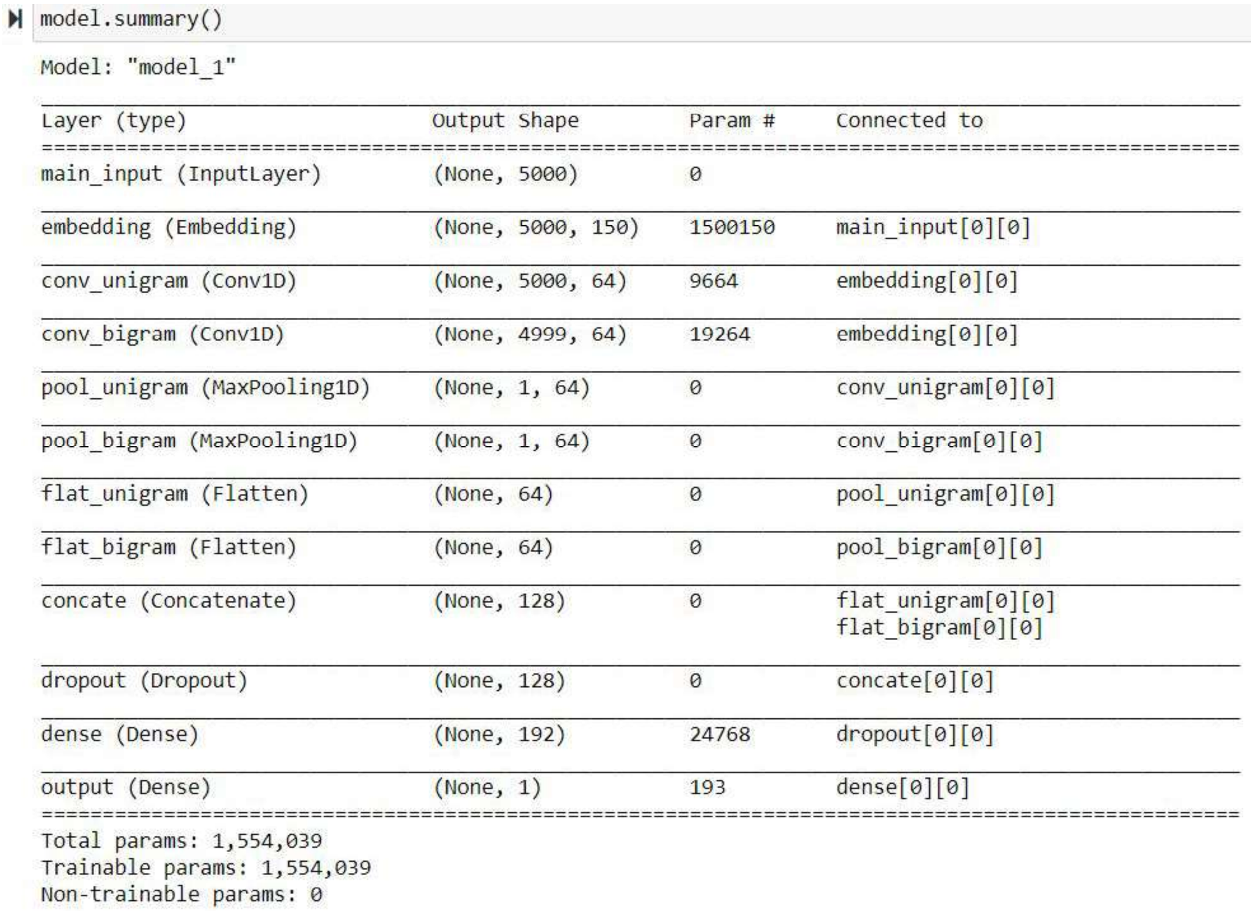
Keras tokenizer was then used to tokenize all the whitepapers in the dataset.

MAX_DOC_LEN was set to 5000 as few of the whitepapers we analyzed were to be between 3000 and 4000 words.

All whitepaper sequences were padded and truncated towards the right to achieve a length of 5000.

Then the data then was split into testing and training set by setting X as the padded sequence of whitepapers and Y as the new column risk obtained from the rating score. The training set consisted of 90 percent of the data and the test set consisted of 10 percent of the data.

Then the following CNN model was constructed to analyze the whitepapers

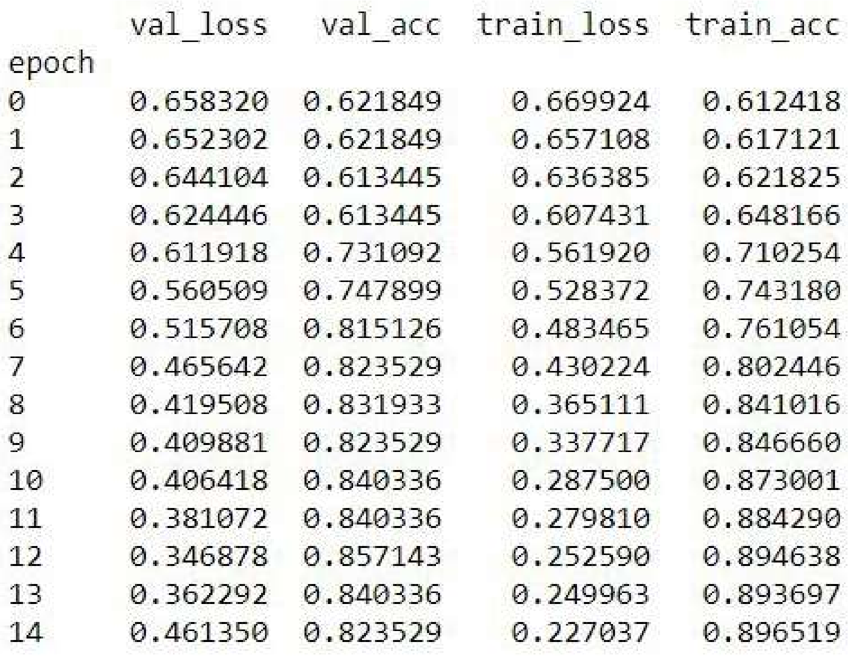


```
model.summary()
```

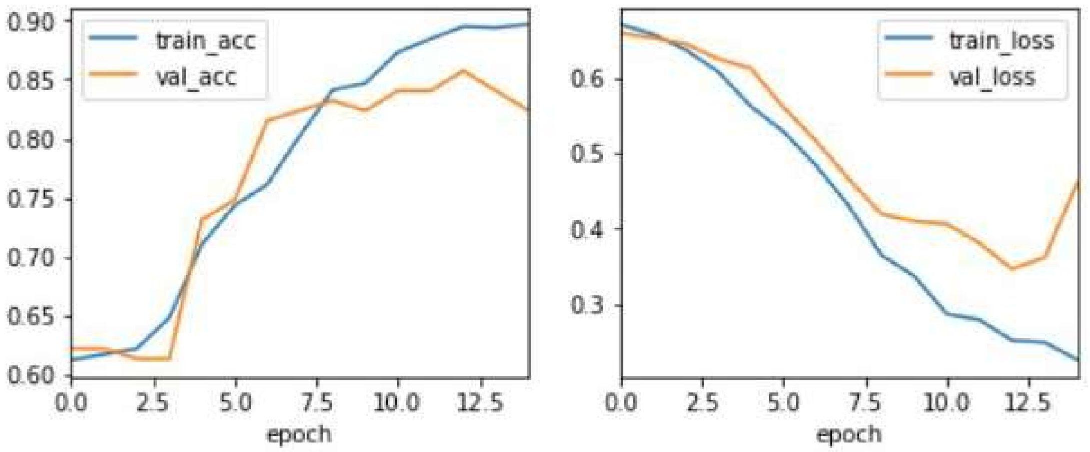
```
Model: "model_1"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
main_input (InputLayer)         (None, 5000)         0
__________________________________________________________________________________________________
embedding (Embedding)           (None, 5000, 150)    1500150     main_input[0][0]
__________________________________________________________________________________________________
conv_unigram (Conv1D)           (None, 5000, 64)     9664        embedding[0][0]
__________________________________________________________________________________________________
conv_bigram (Conv1D)            (None, 4999, 64)     19264       embedding[0][0]
__________________________________________________________________________________________________
pool_unigram (MaxPooling1D)     (None, 1, 64)        0           conv_unigram[0][0]
__________________________________________________________________________________________________
pool_bigram (MaxPooling1D)      (None, 1, 64)        0           conv_bigram[0][0]
__________________________________________________________________________________________________
flat_unigram (Flatten)          (None, 64)           0           pool_unigram[0][0]
__________________________________________________________________________________________________
flat_bigram (Flatten)           (None, 64)           0           pool_bigram[0][0]
__________________________________________________________________________________________________
concate (Concatenate)           (None, 128)          0           flat_unigram[0][0]
                                                                 flat_bigram[0][0]
__________________________________________________________________________________________________
dropout (Dropout)               (None, 128)          0           concate[0][0]
__________________________________________________________________________________________________
dense (Dense)                   (None, 192)          24768       dropout[0][0]
__________________________________________________________________________________________________
output (Dense)                  (None, 1)            193         dense[0][0]
==================================================================================================
Total params: 1,554,039
Trainable params: 1,554,039
Non-trainable params: 0
```

Result of the CNN model:

We trained the model on the training set data and tested it on the test set data for each epoch. Early stopping to stop the training the model when the validation loss decreased for 2 successive epochs and we got the following results.



```
         val_loss   val_acc  train_loss  train_acc
epoch
0        0.658320  0.621849    0.669924   0.612418
1        0.652302  0.621849    0.657108   0.617121
2        0.644104  0.613445    0.636385   0.621825
3        0.624446  0.613445    0.607431   0.648166
4        0.611918  0.731092    0.561920   0.710254
5        0.560509  0.747899    0.528372   0.743180
6        0.515708  0.815126    0.483465   0.761054
7        0.465642  0.823529    0.430224   0.802446
8        0.419508  0.831933    0.365111   0.841016
9        0.409881  0.823529    0.337717   0.846660
10       0.406418  0.840336    0.287500   0.873001
11       0.381072  0.840336    0.279810   0.884290
12       0.346878  0.857143    0.252590   0.894638
13       0.362292  0.840336    0.249963   0.893697
14       0.461350  0.823529    0.227037   0.896519
```



Here we can see that at epoch 12(as it starts from 0) the loss is minimum on the validation/test set which is 0.346878 and accuracy is maximum at 0.857143 or 85.71%.

Analysis of Results:

MNB and Logistic regression did not perform well. This might be because we tried using Tf-IDF vector weights for all the text, maybe it can improve if all the whitepapers are well structured and we can extract specific information like how the coins will be allocated after crowdfunding, their sales, and marketing strategy as well as their technical implementation plan.

In clustering, the word2vec model is maybe forming word vectors with similar weights because most of the words in the text are similar and therefore appropriate clusters are not forming and are overlapping.

LDA performed somewhat well in the topic assignment as we can see 10 different categories in the result.

In CNN at epoch 12 the loss is minimum on the validation/test set which is 0.346878 and accuracy is maximum at 0.857143 or 85.71% which means it performed well.

How to improve?

We can improve the results by collecting more data about the whitepapers of different coins.

We can improve by cleaning the data more appropriately and extracting useful and important features from whitepaper text before feeding it into the models.

Increasing the number of hidden layers and tuning the hyperparameters accordingly.

Trying other different deep learning techniques such as a recurrent neural network.

Building and training a deep learning model that has different parameters(X_values) such as market soft cap, hard cap, exchange rate, etc along with the whitepaper.

Conclusion: We found that there was a relation between the whitepaper and the rating of an Initial Coin Offering.

Therefore it may be possible to detect a fake ICO to very high accuracy by training a model with a more large amount of data and parameters.

Future Work

- Collect more whitepapers and data by scraping from other websites.
- Collect more info about the teams behind ICO coins and price change data for each coin.
- Try different deep learning models with a higher number of hidden layers.

References

- https://www.investopedia.com/news/what-ico/
- https://en.wikipedia.org/wiki/Initial_coin_offering
- https://en.wikipedia.org/wiki/White_paper
- https://blockgeeks.com/guides/crypto-whitepaper/
- https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148
- https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html
- https://www.jasondavies.com/wordcloud/
- Lecture Notes