Project Report

Bitcoin Tweets Analysis

Mining Tweets to Predict Future Bitcoin Price

CSCI-B565 Data Mining, Fall 2022

Group Members:

Ashutosh Hathidara (ashuhath@iu.edu)

Gaurav Atavale (gatavale@iu.edu)

Suyash Chaudhary (suschaud@iu.edu)

Abstract

Bitcoin has increased investment interests in people during the last decade. We have seen an increase in the number of posts on social media platforms about cryptocurrency, especially Bitcoin. This project focuses on analyzing user tweet data in combination with Bitcoin price data to see the relevance between price fluctuations and the conversation between millions of people on Twitter. This study also exploits this relationship between user tweets and bitcoin prices to predict the future bitcoin price. We are utilizing novel techniques and methods to analyze the data and make price predictions.

Keywords: Bitcoin tweets analysis, Bitcoin price prediction, Cryptocurrency price prediction

Contents

1. Introduction

Cryptocurrency has been a constant source of attention for people in the last decade. Especially bitcoin, one of the cryptocurrencies which have increased investment interest significantly. Social media platforms like Twitter were flooded with tweets related to bitcoin and other cryptocurrencies. People have become more focused on identifying investment opportunities in crypto firms. But due to this, crypto markets have also become too volatile and fluctuating. The price for specific cryptocurrencies changes when a renowned celebrity or politician tweets about it. Positive or negative fluctuations depend upon the sentiments of tweets and the market conditions. Market conditions still affect very less as compared to people's sentiments.

In this project, we are analyzing tweets related to Bitcoin extracted from the Twitter feed. We are using the Bitcoin Tweets dataset [2] on Kaggle. The dataset contains 16M tweets made by people around the world. Due to this, the dataset contains tweets from multiple languages and regions. The dataset has the most number of tweets in duration from 2016-01-01 to 2019-03-29. Although the dataset also contains tweets before and after this time interval, the data for that is very sparse. Because of this, we are only analyzing the tweets in the time duration mentioned above.

	user	timestamp	replies	likes	retweets	text
0	KamdemAbdiel	2019-05-27 11:49:14+00	0.0	0.0	0.0	È appena uscito un nuovo video! LES CRYPTOMONN...
1	bitcointe	2019-05-27 11:49:18+00	0.0	0.0	0.0	Cardano: Digitize Currencies; EOS https://t.co...
2	3eyedbran	2019-05-27 11:49:06+00	0.0	2.0	1.0	Another Test tweet that wasn't caught in the s...
3	DetroitCrypto	2019-05-27 11:49:22+00	0.0	0.0	0.0	Current Crypto Prices! \n\nBTC: $8721.99 USD\n...
4	mmursaleen72	2019-05-27 11:49:23+00	0.0	0.0	0.0	Spiv (Nosar Baz): BITCOIN Is An Asset & NO...

Figure 1: Raw data extracted from Kaggle

Figure 1 illustrates the features in the dataset. Each row in the dataset consists of a single tweet. The metadata of the tweet is also given. Metadata consists of the username of the user who made the tweet, the timestamp when the tweet was made, and the traffic (replies, likes, retweets) on the tweet after it was made. The usernames provided are not fake or artificial. These are the real usernames of people at the time of data extraction.

This report is structured in a way that first we will illustrate the methods we used for data mining-related tasks in the next section. Later in the results section, we will reveal the results we observed from the methods we employed.

2. Methods

Beginning with the raw dataset, we employed various preprocessing and analysis techniques. Later, we also utilized prediction techniques to validate whether the analysis we performed as part of the data mining tasks is actually helpful in real-time predictions. The below subsections explain the project setup and each of the methods we used for processing, analyzing, and predicting the data.

2.1 Exploratory Data Analysis

As the first step of the data mining process, we applied some preprocessing and analysis techniques to the dataset. As we mentioned earlier, the dataset contains tweets from many languages. We first applied language detection on the tweets using the python package langdetect [5]. We observed that 80% of the tweets in the dataset are English-language tweets. For flexibility, we filtered only English-language tweets for further analysis.

For analyzing this filtered dataset, we aggregated various features on a day level. Features like replies, likes, and retweets can be aggregated by adding them after grouping them on the day level. For tweet text, we have performed sentiment analysis [22] [11] to find the number of positive, negative, and neutral tweets every day. More about sentiment analysis is described in the next subsection. We also compared the trend for each of these features' tweet volume, likes, replies, and retweets from 2016 to 2019. We also segregated tweet volume w.r.t. the number of likes (> 0, > 10, > 100, > 1000) and observed the trend for that. We performed analysis on tweet volume by aggregating it to the hour of the day and day of the week independently to see if there is any particular day or hour when people are tweeting more about bitcoin.

As part of this step, we also preprocessed the data and made it ready for prediction purposes. We aggregated the data at a day level as described above and included all the features like day-level tweet volume, number of tweets with likes (> 0, > 10, > 100, > 1000), number of tweets with retweets (> 0, > 100), number tweets with the particular sentiment (positive, negative, neutral), one hot encoding the day of the week, one hot encoding hour of the day, price of bitcoin on the previous day, etc.

2.2 Sentiment Analysis

To understand the mood or opinion of the tweet, we performed sentiment analysis [10] on our dataset. Tweets are generally noisy due to the inclusion of mentions, URLs, and emoticons. The tweet-preprocessor [3] package was used to clean the data before sentiment analysis in order to counteract the influence of this noise in tweets. For sentiment analysis, we used two separate libraries: vaderSentiment [4] and textblob [1]. When compared to vaderSentiment, textblob labeled tweets more 'accurately'. We performed the sentiment analysis on the entire 'English' Bitcoin tweets for the selected duration.

2.3 Clustering Analysis

Before we actually went ahead with clustering, we performed quite a bit of preprocessing on our data, we started with dropping the text feature from the dataset as it was irrelevant to the clustering task followed by removing data that dated back to beyond 2016 and finally implementing another variation of clustering on the data i.e. we aggregated all the data into user level which gave us the total count of data features such as likes, comments, and retweets.

After this preprocessing, we experimented with three different approaches to Clustering namely K-means clustering [15], Hierarchical Clustering [20] and DBSCAN

Clustering [14]. The data worked for K-means clustering without a hitch but for Hierarchical clustering and DBSCAN Clustering, we had to work with a fraction of our original data as the algorithms were running out of allocated memory.

2.4 Regression

As part of this step, we want to predict the bitcoin price of the next day based on various features we have after preprocessing the data. We will also include current-day bitcoin prices as part of this step. We have almost 3 years of data for tweets and we have aggregated it to day level. So, we have 1100 (1 for each day) examples in our processed dataset. We have split this dataset into training and testing datasets where the testing split is 10% of the total dataset. The data is split such that all the training data contains consecutive 90% of the days and the last 10% of days goes into the testing dataset. For standardizing the dataset, we used StandardScalar from sklearn [16].

We employed different types of models for regression experiments. We experimented with simple Linear regression [17], Ridge regression [9], and Lasso regression [18] models as part of classical regression models. We also experimented with tree-based regression models like Decision-Tree regression [8] and Random Forest regression [21]. Moreover, we also employed an artificial neural network [6] approach for training the dataset. Some of the models described above require parameter tuning. We have used the K-fold cross-validation technique in combination with GridSearch for parameter tuning. We performed a comparative analysis of these models considering a common metric of comparison.

2.5 Classification

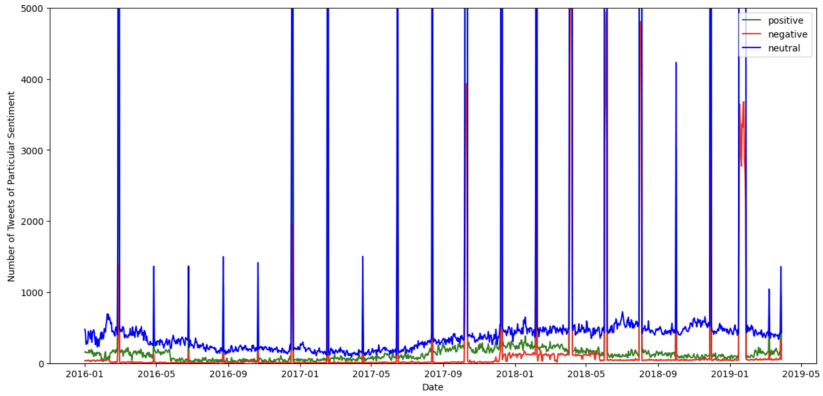
To predict if the price of bitcoin will dip or rise compared to the previous day, we have used various classification frameworks like KNN [13], Logistic Regression model [7], Naive Bayes model [23], Kernel SVM [19], Decision Tree Classification model, Decision Tree Classification model, Random Forest Classification, XGBoost [12], and Light GBM.

The objective was to determine the direction of movement of the BTC price. The data and features remain the same as our Regression model. We have added a new classification Target Variable which takes the values of 0 and 1 based on price going down or up respectively. Here as well we kept train to test ratio as 90:10. We have performed k-fold cross-validation and grid search for all the frameworks. Accuracy, Recall, Precision and F1 scores are the metrics of measurement we have used here. Finally we have plotted a ROC curve to compare the results of each model.

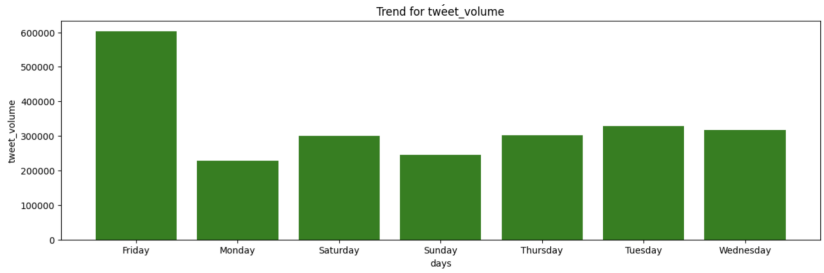
3. Results

We have described the data processing and analysis processes in detail in the previous section. From the basic analysis of the day-level aggregation of the dataset, we have understood that the daily tweet volume and other tweet-related metrics (likes, replies, retweets) have increased from 2016 to 2019. Also, we have seen some regular impulses

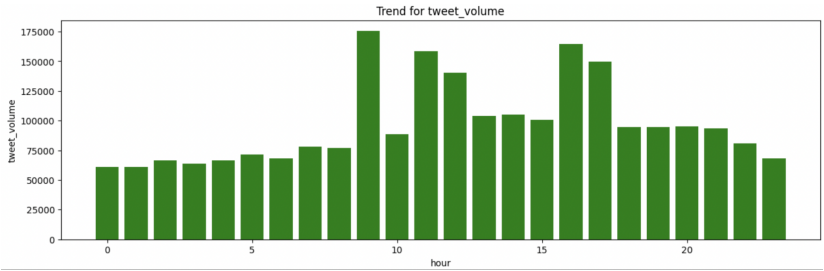
in the metric statistics where the metric values shoot up the graph. The reason behind that might be extraordinary fluctuation in the bitcoin price or other unusual hype in the cryptocurrency market. When we analyzed the number of tweets of certain sentiments (positive, negative, and neutral), we found out that the trend for each of them has been almost constant throughout the time (2016-2019). Comparatively, people have made more neutral tweets than positive and negative tweets as shown in figure 2 (a). We also performed analysis by observing the number of tweets on a certain day of the week. As shown in figure 2 (b), we can see that people are tweeting more about bitcoin on Fridays than any other day in the week. Additionally, we performed analysis to observe similar statistics about the hour of the day. We can observe in figure 2 (c) that the plot has a higher number of tweets from 9 am to 5 pm. This is usually work time and people are tweeting more about bitcoin during working hours.



(a) Sentiment Trend



(b) Day-level Tweet Volume

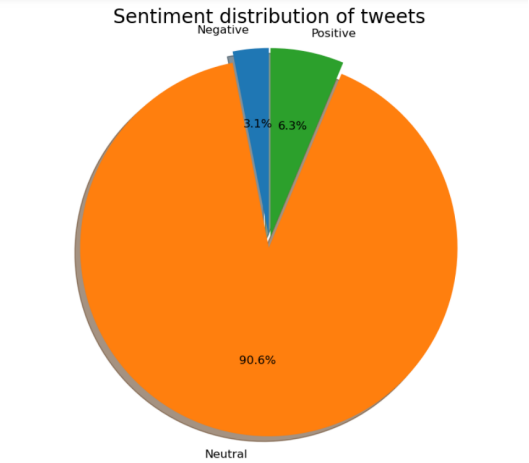


(c) Hour-level Tweet Volume

Figure 2: Results extracted from EDA

With respect to Sentiment Analysis, we could see that majority of the tweets are unbiased and informational. They don't share a specific emotion and are not biased towards positive or negative. More than 90% of tweets fall into this category. Nearly

7% of tweets are positive and 3% tweets are negative(distribution is shown in the figure below figure 3 (a)). We didn't find any correlation between sentiment and movement of price. To validate that the categorization was done right, we printed out a word cloud and we could visualize positive words dominating in the positive sentiment category and similarly negative words in the negative sentiment category. We see unbiased words in the Neutral category as shown in figure 3 (b).



(a) Distribution of Sentiments

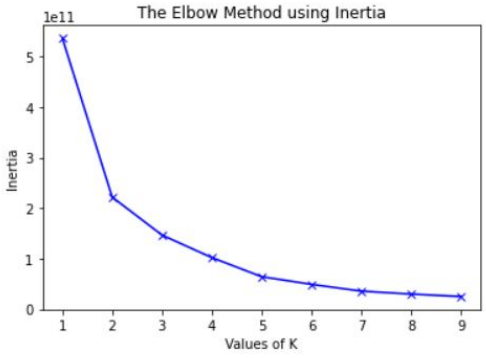
Sentiment	Corr with Bitcoin price
Positive Sentiment	0.13
Negative Sentiment	0.15
Neutral Sentiment	0.07

(b) Correlation with Bitcoin price

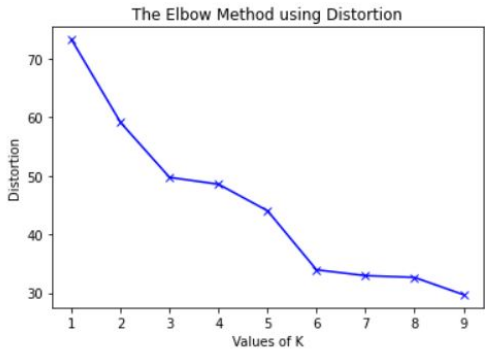
Figure 3: Results extracted from Sentiment Analysis

If we talk about clustering, K-means Clustering yielded different results compared to what Hierarchical and DBSCAN Clustering produced, using evaluation metrics like distortion and inertia, we were able to plot the elbow curve and have k=3 as evident as shown in figure 4 (a) and (b) using K-means Clustering but for Hierarchical clustering and DBSCAN we received k=1 as the ideal k value based on the dendrogram we got for hierarchical clustering and the epsilon value we got from the elbow curve for DBSCAN. Hierarchical and DSCAN clustering might have produced different results because they weren't really working with the entirety of data, the results might have been different if the algorithms supported computing the entire data but then again when you take the the original tweet text out of context, the remaining data might not produce desirable results when it comes to grouping users into separate groups.

As part of regression analysis, we performed the next-day bitcoin price prediction. We have used different types of models as described in the previous section. Using the simple regression models, we have found out that all 3 models linear regression, ridge regression, and lasso regression perform really well in predicting bitcoin price. Figure 5 (a) illustrates the results of ridge regression. The yellow scatter plot in the figure describes the original price data. The green line is the predicted price for training data and the red line is the predicted price for testing data. We are getting very similar results in the case of linear regression and lasso regression. In the case of tree-based models, we tried decision tree regression and random forest regression. Decision tree regression performs really well on the training dataset but performs very poorly on the test dataset. Random forest performs relatively better at testing datasets as compared to decision tree regression but both of these tree-based models perform worse than regression models. The reason behind the poor performance of tree based models



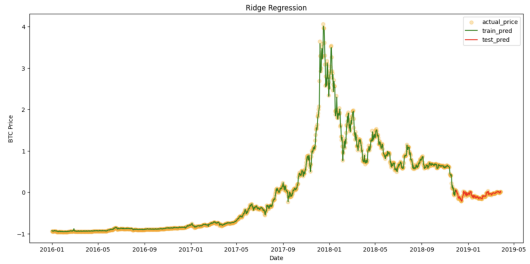
(a) Elbow curve for inertia



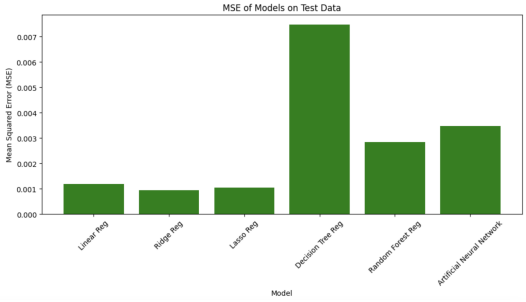
(b) Elbow curve for distortion

Figure 4: K-Means clustering analysis

can be because of splitting datasets and creating hard rules. These rules will not be generalizable for testing dataset. We also performed experiments on Artificial Neural Networks (ANN). We found out that it performs better than decision trees for the test dataset but it still performs poorly as compared to regression models. Figure 5 (b) illustrates the comparison of Mean Squared Error (MSE) on the test dataset of each of the models. We can see that the decision tree regression model performs the worst and the ridge regression performs the best.



(a) Actual BTC price & Ridge regression predicted price



(b) Comparison of MSE between different regression models

Figure 5: Results extracted from Regression Analysis

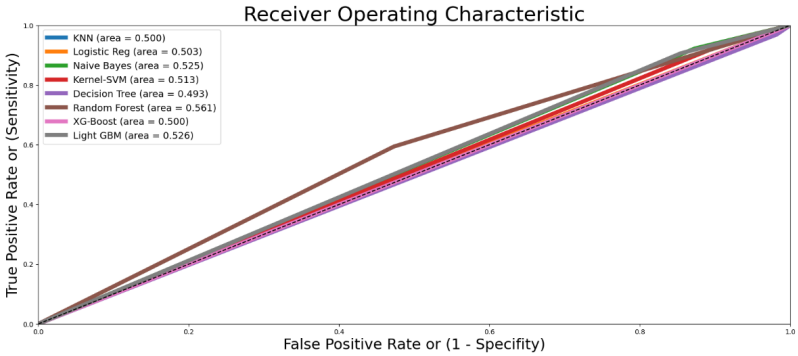


Figure 6: Comparing ROC curve of different classification models

Talking about classification analysis, We achieved the best results from Random forest classifier, which gave us accuracy of 62% and F1 score of 75%. The parameters with which we achieved these numbers are class_weight:balanced, criterion:gini, max_features:log2, n_estimators: 100 and min_samples_split:2. The table 1 shows the scores for each model and figure 6 shows the Receiver operating characteristic (ROC) curve.

	Accuracy	**Precision**	**Recall**	**F1-score**
Logistic Regression	0.49	0.54	0.41	0.46
KNN	0.54	0.54	1	0.7
Naive Bayes	0.56	0.55	0.92	0.69
Kernel SVM	0.55	0.54	0.95	0.69
Decision Tree	0.53	0.53	0.97	0.69
Random Forest	0.62	0.64	0.86	0.75
XGBoost	0.54	0.54	1	0.7
Light GBM	0.55	0.55	0.91	0.69

Table 1: Scores for models involved in Classification Analysis

4. Conclusion & Discussion

As we can see from the data, the BTC price field shows extremely high variation for this timeframe. Lowest value is $364 and highest is $19497 exhibiting a range of $19K. This variation in data makes it hard to predict the exact price of BTC the next day through regression techniques. Also, as the fluctuations are huge and frequent, this poses a serious problem to classify the movement to see if price is going up or down. After an extensive EDA and understanding the data, we tried several techniques for regression and classification to achieve these objectives and have achieved decent results. We could further improve them by collecting more data and expanding the feature set. Some of the features which would help us could be- verified flag, google trends data, authentic news channels/accounts flag etc.

We could also deploy a similar methodology to predict prices for other cryptocurrencies as well. Some of the cryptocurrencies have few variants of hashtags which has to be taken into consideration while collecting tweets data.

We observed through clustering that the users could potentially be grouped into three categories namely, users who positively affect the price, users who negatively affect the price and users who have no effect on the price whatsoever. But the results of Clustering are something to be skeptical about as each algorithm produced different results due to lack of computation ability on the clustering algorithms among other reasons.

Lastly, we want to make a strong argument with this study that the user related tweets affect the price fluctuations of cryptocurrencies like Bitcoin and we can predict the price variation and even exact price using some of the features extracted from day-level aggregation of tweets.

5. References

Bibliography

[1] textblob, 9 2014. URL https://pypi.org/project/textblob/0.9.0/.

[2] Bitcoin tweets - 16m tweets, 11 2019. URL https://www.kaggle.com/datasets/alaix14/bitcoin-tweets-20160101-to-20190329.

[3] tweet-preprocessor, 5 2020. URL https://pypi.org/project/tweet-preprocessor/.

[4] vaderSentiment, 5 2020. URL https://pypi.org/project/vaderSentiment/.

[5] langdetect, 5 2021. URL https://pypi.org/project/langdetect/.

[6] Amini Pishro A., Zhang S., Huang D., Xiong F., Li W., and Yang Q. Application of artificial neural networks and multiple linear regression on local bond stress equation of UHPC and reinforcing steel bars. *Scientific Reports*, 11(1), 7 2021. doi: 10.1038/s41598-021-94480-2. URL http://dx.doi.org/10.1038/s41598-021-94480-2.

[7] Amini Pishro A., Zhang S., Huang D., Xiong F., Li W., and Yang Q. Application of artificial neural networks and multiple linear regression on local bond stress equation of UHPC and reinforcing steel bars. *Scientific Reports*, 11(1), 7 2021. doi: 10.1038/s41598-021-94480-2. URL http://dx.doi.org/10.1038/s41598-021-94480-2.

[8] Gordon A. D., Breiman L., Friedman J. H., Olshen R. A., and Stone C. J. Classification and Regression Trees. *Biometrics*, 40(3):874, 9 1984. doi: 10.2307/2530946. URL http://dx.doi.org/10.2307/2530946.

[9] Hoerl A. E. and Kennard R. W. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, 12(1):55–67, 2 1970. doi: 10.1080/00401706.1970.10488634. URL http://dx.doi.org/10.1080/00401706.1970.10488634.

[10] Jethin Abraham, Daniel Higdon, John Nelson, and Juan Ibarra. Cryptocurrency Price Prediction Using Tweet Volumes and Sentiment Analysis. *SMU Data Science Review*, 1(3):1–, 12 2017.

[11] J. Johan Bollen, H. Huina Mao, and X. Xiaojun Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8, 3 2011. doi: 10.1016/j.jocs.2010.12.007. URL http://dx.doi.org/10.1016/j.jocs.2010.12.007.

[12] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. *CoRR*, abs/1603.02754, 2016. URL http://arxiv.org/abs/1603.02754.

[13] Padraig Cunningham and Sarah Jane Delany. k-nearest neighbour classifiers: 2nd edition (with python examples). *CoRR*, abs/2004.04523, 2020. URL https://arxiv.org/abs/2004.04523.

[14] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. *Knowledge Discovery and Data Mining*, pages 226–231, 8 1996.

[15] Hartigan J. A. and Wong M. A. Algorithm AS 136: A K-Means Clustering Algorithm. *Applied Statistics*, 28(1):100, 1979. doi: 10.2307/2346830. URL http://dx.doi.org/10.2307/2346830.

[16] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. Scikit-learn: Machine Learning in Python. *Le Centre pour la Communication Scientifique Directe - HAL - memSIC*, 1 2011.

[17] Sampson R., Assuncao andP. D., Montgomery D. C., and Peck E. A. Introduction to Linear Regression Analysis. *Journal of the American Statistical Association*, 88(421):383, 3 1993. doi: 10.2307/2290746. URL http://dx.doi.org/10.2307/2290746.

[18] Tibshirani R. Regression Shrinkage and Selection Via the Lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1 1996. doi: 10.1111/j.2517-6161.1996.tb02080.x. URL http://dx.doi.org/10.1111/j.2517-6161.1996.tb02080.x.

[19] Jiang S., Hartley R., and Fernando B. Kernel Support Vector Machines and Convolutional Neural Networks. *2018 Digital Image Computing: Techniques and Applications (DICTA)*, 12 2018. doi: 10.1109/dicta.2018.8615840. URL http://dx.doi.org/10.1109/dicta.2018.8615840.

[20] Cohen V., Kanade V., Mallmann-Trenn F., and Mathieu C. Hierarchical Clustering: Objective Functions and Algorithms. *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 378–397, 1 2018. doi: 10.1137/1.9781611975031.26. URL http://dx.doi.org/10.1137/1.9781611975031.26.

[21] Svetnik V., Liaw A., Tong C., Culberson J. C., Sheridan R. P., and Feuston B. P. Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling. *Journal of Chemical Information and Computer Sciences*, 43(6):1947–1958, 11 2003. doi: 10.1021/ci034160g. URL http://dx.doi.org/10.1021/ci034160g.

[22] Patodkar Vaibhavi N and I.R S. Sheikh. Twitter as a Corpus for Sentiment Analysis and Opinion Mining. *IJARCCE*, 5(12):320–322, 12 2016. doi: 10.17148/ijarcce.2016.51274. URL http://dx.doi.org/10.17148/ijarcce.2016.51274.

[23] Vikramkumar, Vijaykumar B, and Trilochan. Bayes and naive bayes classifier. *CoRR*, abs/1404.0933, 2014. URL http://arxiv.org/abs/1404.0933.