Development of a dataFrame and a Bot to predict NFT-collection performance

Marc Durban, Joaquim Gabarró
Universitat Politècnica de Catalunya, España, marc.durban@gmail.com
Universitat Politècnica de Catalunya, ALBCOM. CS Dept, España, gabarro@cs.upc.edu

ABSTRACT

There has been an enormous growth of blockchain technologies at the end of 2021. Particularly one of its assets called NFTs, an asset that is a new form of digital scarcity. This work aims to study their still highly volatile market and develop both a dataFrame and a bot, capable of predicting a NFT collection performance in the near future [1].

We focus on the NFT main marketplace OpenSea and specifically the collections in the Ethereum ecosystem, which has the 80% of the market as of early 2022. Because of its novelty, few applications and studies have been done on this field to date. However, this has not stopped this new asset to rapidly rise in popularity from a small 10k users to more than 1.5M only in the last year. It is true that as interest grows more and more information and projects are being published, but none similar to this one (as of starting date February 2022). Despite this, other predictive applications developed for economic fields such as the stock market, might be a helpful reference.

1. WHAT ARE A BLOCKCHAIN, A NFT AND A BOT?

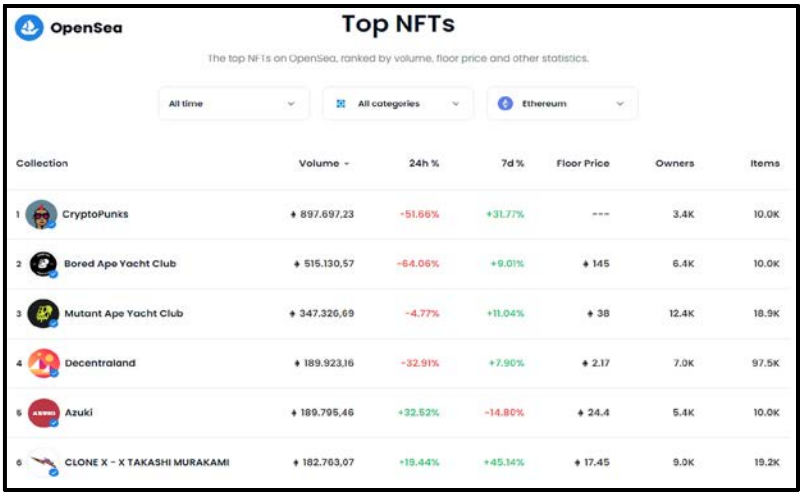
A *blockchain* is a list of blocks linked using cryptography [1]. They are used to make decentralized transactions between users. Multiple transactions get included into a block, verified and afterwards added to the ledger. The ledger is a decentralized, shared and immutable list of verified blocks that serves as reference for everyone in the network.

A *Non Fungible Token* or NFT, is a token/asset linked to a blockchain which is unique, indivisible and tradeable. NFTs aim to create verifiable digital scarcity and are commonly grouped, under a same brand, in what is known as *collections*. Usually they are traded in *marketplaces* such as OpenSea, looksRare, Solanart or Magic Eden.

A *bot* is a software application that runs automated tasks. Often they are used to perform tasks that are simple and repetitive.

2. ON DATA SELECTION

When trying to perform a data analysis, the size of the sample is a key factor [2]. Do not forget that data is scarce when we are talking about NFT transactions. In our case, we make use of the OpenSea ranking, (look at https://opensea.io/). In which, as we can see in Figure 1, we can consult the top best-selling NFT collections in the platform.



OpenSea

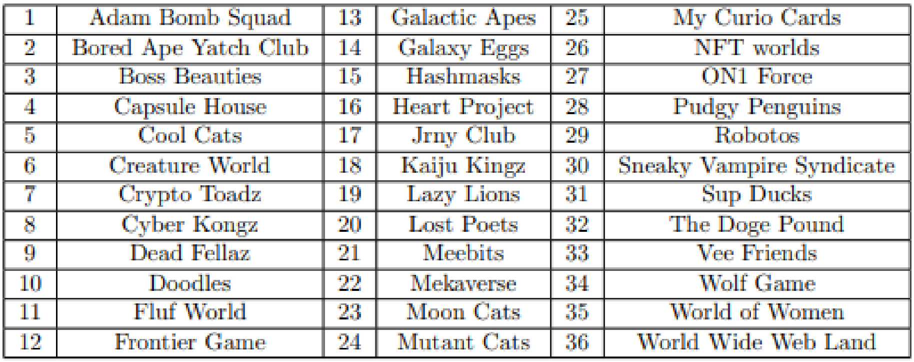
Top NFTs

The top NFTs on OpenSea, ranked by volume, floor price and other statistics.

Collection		Volume	24h %	7d %	Floor Price	Owners	Items
1	CryptoPunks	♦ 897.697,23	-51.66%	+31.77%	---	3.4K	10.0K
2	Bored Ape Yacht Club	♦ 515.130,57	-64.06%	+9.01%	♦ 145	6.4K	10.0K
3	Mutant Ape Yacht Club	♦ 347.326,69	-4.77%	+11.04%	♦ 38	12.4K	18.9K
4	Decentraland	♦ 189.923,16	-32.91%	+7.90%	♦ 2.17	7.0K	97.5K
5	Azuki	♦ 189.795,46	+32.52%	-14.80%	♦ 24.4	5.4K	10.0K
6	CLONE X - X TAKASHI MURAKAMI	♦ 182.763,07	+19.44%	+45.14%	♦ 17.45	9.0K	19.2K

Figure 1. Top Ethereum collections on OpenSea, look at https://opensea.io/rankings

To realize this project 36 collections from this ranking have been chosen. Ensuring with this a more than enough volume of transactions for each of the collections.



1	Adam Bomb Squad	13	Galactic Apes	25	My Curio Cards
2	Bored Ape Yatch Club	14	Galaxy Eggs	26	NFT worlds
3	Boss Beauties	15	Hashmasks	27	ON1 Force
4	Capsule House	16	Heart Project	28	Pudgy Penguins
5	Cool Cats	17	Jrny Club	29	Robotos
6	Creature World	18	Kaiju Kingz	30	Sneaky Vampire Syndicate
7	Crypto Toadz	19	Lazy Lions	31	Sup Ducks
8	Cyber Kongz	20	Lost Poets	32	The Doge Pound
9	Dead Fellaz	21	Meebits	33	Vee Friends
10	Doodles	22	Mekaverse	34	Wolf Game
11	Fluf World	23	Moon Cats	35	World of Women
12	Frontier Game	24	Mutant Cats	36	World Wide Web Land

Figure 2. Selected collections [1].

Another thing to consider is the homogeneity of the dataset. Reducing the variables to the minimum, to obtain reliable results. Because of this, secondary collections from already established brands has been discarded (Figure 3).

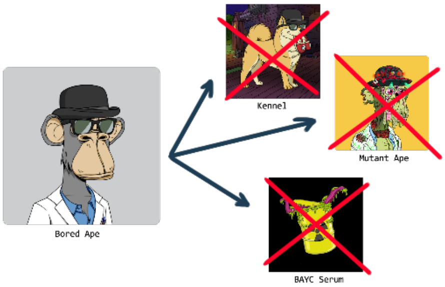


Figure 3. Full Bored Ape Yatch Club project. Main collection plus three derived collections. All the derived collections are discarded.

Last but not least, a limited time frame has been decided to compare all dataFrames in the same timeline. Going from a starting date of December 1st, 2021 to April 30, 2022.

3. THE DATAFRAME

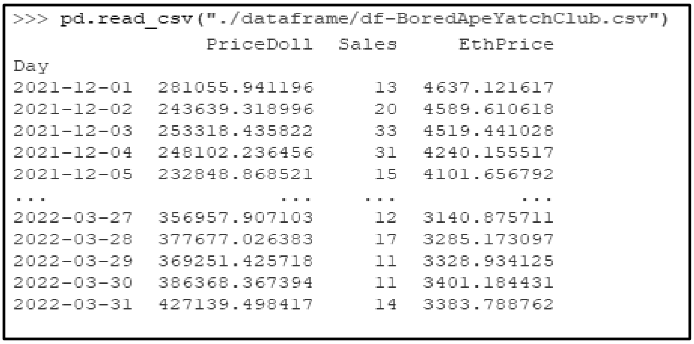
Once the collections of interest are selected, we start with the gathering and construction of a dataFrame, in order to deal with NFT collections. The objective of the dataFrame, from Python's data analysis library Pandas, is to store relevant information for future evaluation and analysis of a given NFT collection.

The data is extracted directly from the blockchain transactions using the API of main NFT marketplace OpenSea. Thousands of transactions are processed and key information such as the average selling price and the number of units sold on any given date are obtained.

It is important to note that there are some rare cases where transactions have been stored differently, because of a registration error or having other particularities. As less than 0.1% of the 212.729 analyzed where affected, they are ignored.

On the other hand, it has been also necessary to take care of data particularities. As one of the most important metrics for this project to track NFT collection performance has been the *average selling price on a given date*. There have been two scenarios where issues have presented for this metric. The first being when there have been 0 transactions in a given date. Making it impossible to calculate an average price and therefore setting it to 0$. The later being the case where few weird pricing sales have taken place on a day. Making the average selling price of that day to be radically different from the ones on the previous and next dates. These two cases create breaks in the continuity of the evolution of prices overtime, hurting the ability of the bot to better train and predict them. Because of this reason it has been decided to copy the average selling price of the previous day when a situation like the ones described is encountered in a dataFrame. This solution only had to be applied in 34 out of the 4356 day-prices in the 36 final dataFrames, which equals to less than 0.008% of the cases.

To the data obtained using the OpenSea API we add the average price of the Ethereum token in each date, currency with which it is exchanged in the analyzed market. This information is obtained using the API of the cryptocurrency stats provider CoinGecko.

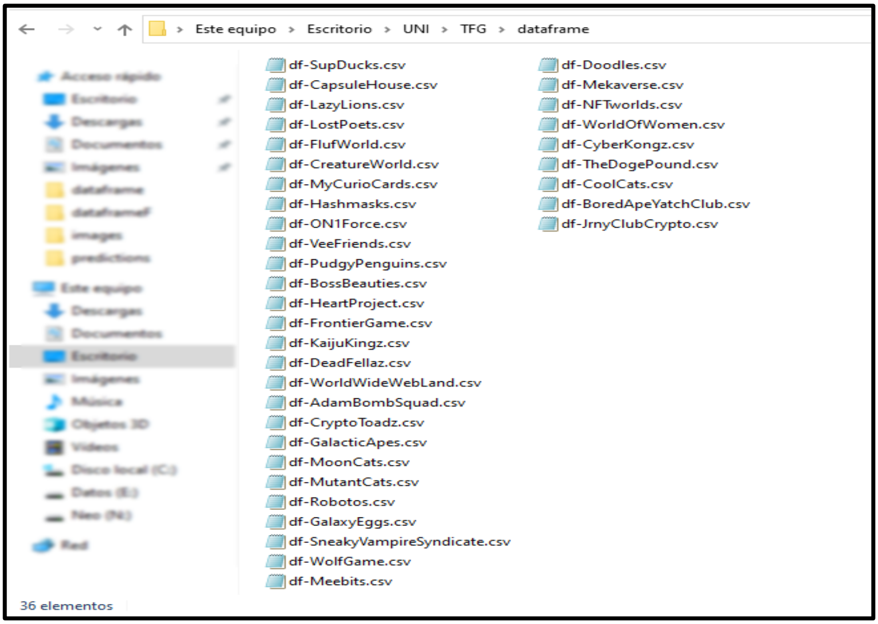


```
>>> pd.read_csv("./dataframe/df-BoredApeYatchClub.csv")
                PriceDoll  Sales     EthPrice
Day
2021-12-01  281055.941196     13  4637.121617
2021-12-02  243639.318996     20  4589.610618
2021-12-03  253318.435822     33  4519.441028
2021-12-04  248102.236456     31  4240.155517
2021-12-05  232848.868521     15  4101.656792
...                   ...    ...          ...
2022-03-27  356957.907103     12  3140.875711
2022-03-28  377677.026383     17  3285.173097
2022-03-29  369251.425718     11  3328.934125
2022-03-30  386368.367394     11  3401.184431
2022-03-31  427139.498417     14  3383.788762
```

Figure 4. DataFrame example of the Bored Ape Yatch Club NFT collection.

As we see in the image above (Figure 4). In the case of our dataFrames, dates are used as index. Using a format of the type YYYY-MM-DD, different than the one in seconds that will be needed to carry out operations, in order to improve readability.

For any given date compressed in our limited timeframe, the average selling price of a NFT collection and its number of sales can be obtained. The first is transformed into the dollar currency, as it is a more stable measure than the Ethereum currency. This conversion is made using the average price of the Ethereum cryptocurrency on a given date and is also included into our dataFrames as the last column value.



df-SupDucks.csv
df-CapsuleHouse.csv
df-LazyLions.csv
df-LostPoets.csv
df-FlufWorld.csv
df-CreatureWorld.csv
df-MyCurioCards.csv
df-Hashmasks.csv
df-ON1Force.csv
df-VeeFriends.csv
df-PudgyPenguins.csv
df-BossBeauties.csv
df-HeartProject.csv
df-FrontierGame.csv
df-KaijuKingz.csv
df-DeadFellaz.csv
df-WorldWideWebLand.csv
df-AdamBombSquad.csv
df-CryptoToadz.csv
df-GalacticApes.csv
df-MoonCats.csv
df-MutantCats.csv
df-Robotos.csv
df-GalaxyEggs.csv
df-SneakyVampireSyndicate.csv
df-WolfGame.csv
df-Meebits.csv
df-Doodles.csv
df-Mekaverse.csv
df-NFTworlds.csv
df-WorldOfWomen.csv
df-CyberKongz.csv
df-TheDogePound.csv
df-CoolCats.csv
df-BoredApeYatchClub.csv
df-JrnyClubCrypto.csv

Fig 5. Directory with all 36 NFT-Collection dataFrames.

Data from every single NFT collection selected gets collected and stored in a directory called ./dataframe. A screenshot of the folder appears in Figure 5.

4. PREDICTING BOT

Remind that ./dataframe, referenced as the storage, now contains 36 .csv files. These files hold the data from each of the selected collections dataFrames, with the respective information from December 1st, 2021 to April 30, 2022.

Given a specific collection from our storage, the bot can finally start the predicting process. Information from the chosen NFT collection gets read from storage and imported into a Pandas object for better manipulation. Data then gets sliced, to obtain the amount of information we want the bot to work with. Being 4 months for the execution by default in this work. The remaining data then gets splited, using a proportion of 70 to 30, into two sets for the training and testing of the bot's model (Figure 6).

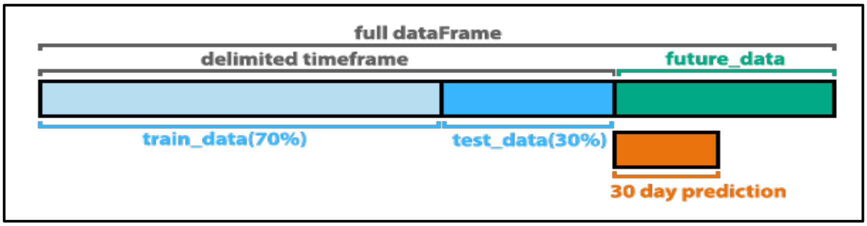


Figure 6. Dataframe partitioning for bot usage.

The algorithm in the bot is the LSTM (Long Short Term Memory) algorithm [2, 3]. Thanks to its ability to store key information about the past in order to make better evaluations and predictions in the future. On top of its great affinity with date-based data. This algorithm has been commonly used by many in stock predicting applications [4, 5, 6]. This is a research domain that seems to share some similarities to the NFT market analyzed on this paper.

Using the before mentioned algorithm and the necessary training and testing data, the LSTM stacked model is then built and trained. Providing the bot with a model capable of realizing predictions for the following date, given the required data. Here is when a process named forecasting takes place. Storing the predictions of the following date as part of the dataFrame data and this way being able to realise further predictions into the future.

An example result of the bot execution is seen in Figure 7. This execution has been made over the VeeFriends NFT collection, using 4 months of data for the model training and building. Predictions are presented by the bot using an overlapped plot with the data highlighted in different colors. In the Figure 7, colors are replaced by grey tones. First, the bot prints in blue the data used to train and build the model. In the Figure 7, this corresponds to data beginning on Dec and ending before Apr. Second, the bot highlights in green the data of our dataFrame that has not been used and will be used to compare with the prediction. In the case of the Figure 7 this corresponds to the data starting from Apr. Last but not least, the bot prints in orange the prediction. In the Figure 7, this corresponds to the more or less straight line starting at Apr.

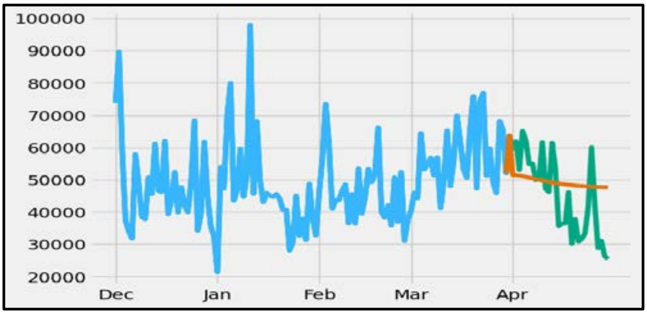
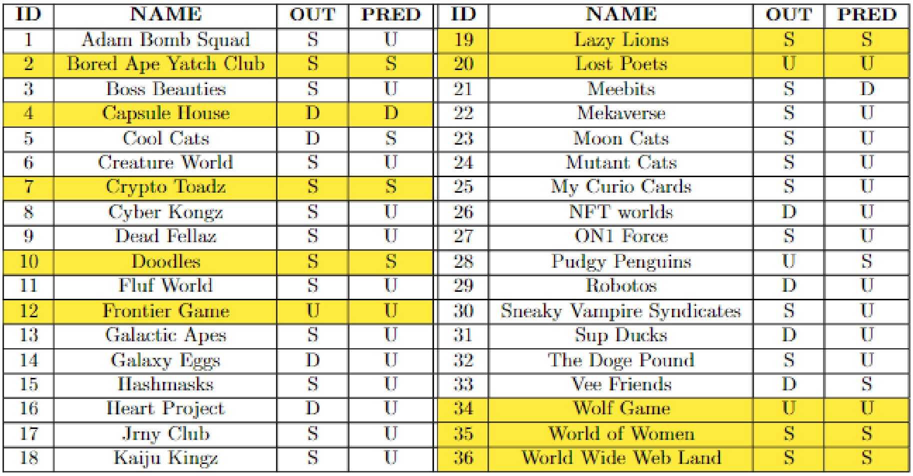


Figure 7. Bot prediction resulting plot.

Taking in consideration that the usual data sizes for LSTM stock prediction projects, such as the ones presented in [5, 6], have at least 3 years of data. We need to take into account this fact when we consider the precision of our model's prediction for a dataset with a size of less than 6 months.

It has seemed reasonable to give in this project more weight to the predicted trend than the adjustment of these prediction to the actual values. For this reason, the average price value of both the prediction and the real data during the same period have been used for evaluation. We divide the different outcomes into three categories, this being: up-trending, U, sideways moving, S, or down-trending, D. We have considered as a successful prediction one that is in the same category as the real outcome (Figure 8).



ID	NAME	OUT	PRED	ID	NAME	OUT	PRED
1	Adam Bomb Squad	S	U	19	Lazy Lions	S	S
2	Bored Ape Yatch Club	S	S	20	Lost Poets	U	U
3	Boss Beauties	S	U	21	Meebits	S	D
4	Capsule House	D	D	22	Mekaverse	S	U
5	Cool Cats	D	S	23	Moon Cats	S	U
6	Creature World	S	U	24	Mutant Cats	S	U
7	Crypto Toadz	S	S	25	My Curio Cards	S	U
8	Cyber Kongz	S	U	26	NFT worlds	D	U
9	Dead Fellaz	S	U	27	ON1 Force	S	U
10	Doodles	S	S	28	Pudgy Penguins	U	S
11	Fluf World	S	U	29	Robotos	D	U
12	Frontier Game	U	U	30	Sneaky Vampire Syndicates	S	U
13	Galactic Apes	S	U	31	Sup Ducks	D	U
14	Galaxy Eggs	D	U	32	The Doge Pound	S	U
15	Hashmasks	S	U	33	Vee Friends	D	S
16	Heart Project	D	U	34	Wolf Game	U	U
17	Jrny Club	S	U	35	World of Women	S	S
18	Kaiju Kingz	S	U	36	World Wide Web Land	S	S

Figure 8. Trending categories

As we see in Figure 9, predictions change when we pass from 4 months to 5 months.

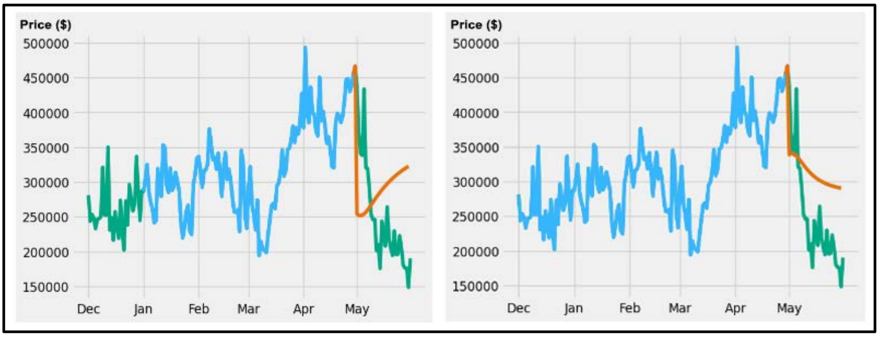


Figure 9. Experiment result comparison between 4 months and 5 months of data used by the bot.

Finally let us consider bot's graphic interface. In order to improve data processing and automatize some actions like result extraction. A simplified interface with some key functionalities have been developed for this project to better interact with the bot and the dataFrame files. The Python library Tkinter has been used to develop this task.The minimalistic appearance of this interface is the result of prioritizing its functionality rather than its aesthetics. An image of how the mentioned layout looks could be seen in the following screenshot (Figure 10).

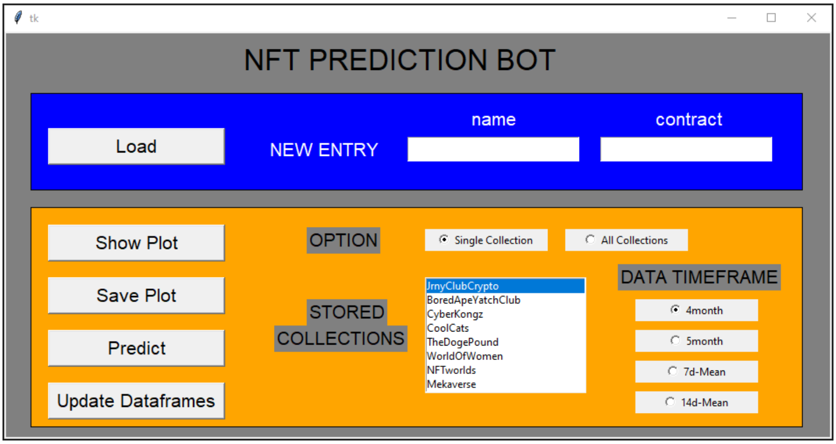


Figure 10. Screenshot of the graphic interface designed for the bot.

5. CONCLUSIONS

Focusing only on the Ethereum market and more specifically on OpenSea, seems to have been the right decision. For instance a direct competitor as looksRare marketplace, which was rapidly growing at the start of the year, ended up dropping in popularity after a few months. On the contrary, OpenSea remained as the most important NFT market in the space. This together with the gained access to the OpenSea developer's API, has helped to gather all the needed data from the desired collections.

On the other hand, implementing a bot that makes predictive analysis using neural networks has been challenging. When it comes to its accuracy and optimistic predictions, found mismatches could be a product of two main factors. First, the unexpected general economic crash produced by historical facts such as the

Ukraine and Russian conflict (escalating in April 18) and the increasing interest rates fruit of the covid (communicated in many instances during the 2nd Quarter of 2022). Events the model developed for this application does not have in consideration and affected all economic markets, the NFT one included. Second, the lack of more than 6 months of data for the NFT asset class. Being maybe too new to be properly analyzed with machine learning algorithms like the LSTM stacked model presented on this project.

6. ACKNOWLEDGMENTS

Joaquim Gabarró has been financed by: Proyecto PID2020-112581GB-C21 financed by MCIN/AEI/10.13039/501100011033.

REFERENCES

[1] Marc Durban (2022). “Development of a dataFrame and a Bot to predict NFT-collection performance”. Degree Final Project in Informatics and Engineering, Facultat de Informàtica de Catalunya, Universitat Politècnica de Catalunya, June 2022.

[2] Andriy Burkov (2019). “The Hundred-Page Machine Learning Book”. Neural Networks and Deep Learning, 61-76.

[3] Sandro Skansi (2018). “Introduction to Deep Learning - From Logical Calculus to Artificial Intelligence”. Recurrent Neural Networks, 135-152.

[4] Hum Nath Bhandari, Binod Rimal, Nawa Raj Pokhrel, Ramchandra Rimal, Rajendra K.C. Khatri (2022). Machine Learning with Applications,”Predicting stock market index using LSTM”. Article 100320.

[5] Roshan Adusumili (2019). “Artificial Intelligence and its Applications in Finance - Utilizing a Keras LSTM model to forecast stock trends”.

[6] Luke Sun (2020). “LSTM for stock price prediction - Technical Walk-through on LSTM-based Recurrent Neural Network Creation for Google Stock Price Prediction”