

QUANT ALGO-TRADING USING DEEP LEARNING NEURAL NETWORK

Sathyabama Institute of Science and Technology

(Deemed to be University)

Submitted in partial fulfilment of the requirements for the award of

Master of Engineering Degree

in

Computer Science and Engineering

By

M JEEVARAMANAN

(Reg. No. 41310001)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI – 600119,

TAMILNADU.

APRIL 2023



SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)



Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Phase-2 Report is the bonafide work of **JEEVARAMANAN M (Reg no:41310001)** who out the project entitled "**QUANT ALGO-TRADING USING DEEP LEARNING NEURAL NETWORK**" under your supervision from December 2022 to April 2023.

Internal Guide

Dr. G. Mathivanan M.E., Ph.D.,

Head of the Department

Dr. L. Lakshmanan M.E., Ph.D.,



Submitted for Viva voce Examination held on **25.4.2023**

Internal Examiner

External Examiner

DECLARATION

I, **M JEEVA RAMANAN (41310001)** hereby declare that the Project Phase-2 Report entitled “**QUANT ALGO-TRADING DEEP LEARNING NEURAL NETWORK**” done by me under the Guidance of **Dr. G. Mathivanan M.E. Ph.D.**, at Sathyabama Institute of Science and Technology is submitted in partial fulfilment of the requirements for the award of Master of Engineering degree in Computer Science and Engineering.

DATE:25.4.2023

PLACE: Chennai

A rectangular box containing a handwritten signature in blue ink, which appears to be 'Jeeva'.

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph. D, Dean**, School of Computing, **Dr. L. Lakshmanan M.E., Ph.D.**, Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. G. Mathivanan M.E., Ph.D.**, for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project phase-1 work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project

ABSTRACT

This work shows the forex data in financial market need to be technically analyzed to increase the efficiency of forex data accuracy, to further increase our odds in making a stable economic decision. we also have implemented Genetic algorithm in feature engineering to find best possible parameter. Using Technical analysis and Deep learning Algorithm we have obtained an optimal Profit Percentage. Implementing Algo-trading in Forex currency market removes human error and Emotional error and the result are analyzed by Algorithm. The algo-trading uses a computer program to follow a set of instruction to place a Buy or Sell order, and in this instruction, we have implemented Deep Learning algorithm to make it more Efficiently and we have automated this to work as per Forex market Duration and to work all day throughout the weekdays without interruption.

Key Words: Forex Market, Algo-trading, Machine Learning, Technical Analysis, Forecasting.

TABLE OF CONTENT

Chapter No:	Title	PAGE NO
	ABSTRACT	V
	LIST OF FIGURES	VII
1	INTRODUCTION	1
	1.1 Aim of the Project	1
	1.2 About Algo-Trading	1
2	OBJECTIVES	3
	2.1 Reduce Human Error	3
	2.2 Trade at Best Possible Price	3
	2.3 Reduce Transaction Cost	4
	2.4 More Accuracy	4
3	LITERATURE SURVERY	5
4	PROPOSED SYSTEM	10
5	SYSTEM ARCHITECTURE	11
6	MODULE	12
	6.1 Pre-Processing	12
	6.2 Feature Engineering	13
	6.2.1 Simple Moving Average (SMA)	14
	6.2.2 Relative Strength Index (RSI)	15
	6.2.3 Deep Learning Implementation	16
7	RESULT AND DISCUSSION	18
8	CONCLUSION	20
9	REFERENCE	21

LIST OF FIGURES

Chapter No:	Title	Page No:
1.3	Algo-Trading Outline	2
5.1	System Architecture	11
6.1	Historical Price	12
6.2	Data before Pre-Processing	13
6.3	Data after Pre-Processing	13
6.4	Simple Moving Average	14
6.5	Numerical Data of SMA	15
6.6	Relative Strength Index	15
6.7	Numerical Data of RSI	16
6.8	RSI Data	16
6.9	Sigmoid Function	17
7.1	RNN Model	18
7.2	Achieved Result	18
7.3	Values in Chart	19
7.4	Data from Live Market	19

Chapter 1

INTRODUCTION

Algorithmic trading (also called automated trading, black-box trading, or algo-trading) uses a computer program that follows a defined set of instructions (an algorithm) to place a trade. The trade, in theory, can generate profits at a speed and frequency that is impossible for a human trader.

The defined sets of instructions are based on timing, price, quantity, or any mathematical model. Apart from profit opportunities for the trader, algo-trading renders markets more liquid and trading more systematic by ruling out the impact of human emotions on trading activities.

1.1 AIM OF THE PROJECT

Trading Manually take time and experience and even with all this we make errors because we have human emotions and psychological factors and, we are slow to make a decision.

But by using Algo-trading we can solve this problem we can use this to reduce the possibility of mistakes be human traders based on emotional and psychological factors.

We can also reduce transaction cost, we can also simultaneously check on multiple market conditions and we can execute trade order at best possible price available.

1.2 ABOUT ALGO-TRADING

An algorithm is a specific set of clearly defined instructions aimed to carry out a task or process. And there are instances when a human trader is not able to handle enormous numbers of trading, and that's when you need intervention of an intelligent algorithm.

Algorithms have gained popularity in the online trading landscape and many big clients demand it. These mathematical algorithms analyse every quote and trade in the stock market, identify liquidity opportunities, and turn the information into intelligent trading decisions. Algorithmic trading, or computer-directed trading, cuts

down transaction costs, and allows investment managers to take control of their own trading processes. Algorithm innovation continues to offer returns for firms with the scale to absorb the costs and to reap the benefits.

Algorithmic trading (automated trading, black-box trading, or simply algo trading) is the process of using computers programmed to follow a defined set of instructions for placing a trade in order to generate profits at a speed and frequency that is impossible for a human trader. Any strategy for algorithmic trading requires an identified opportunity, which is profitable in terms of improved earnings or cost reduction. The algorithmic trading strategies follow defined sets of rules, and are based on timing, price, quantity, or any mathematical model. Apart from profit opportunities for the trader, algorithmic-trading makes markets more liquid and makes trading more systematic by ruling out emotional human impacts on trading activities.

FIG 1.3 ALGO-TRADING OUTLINE

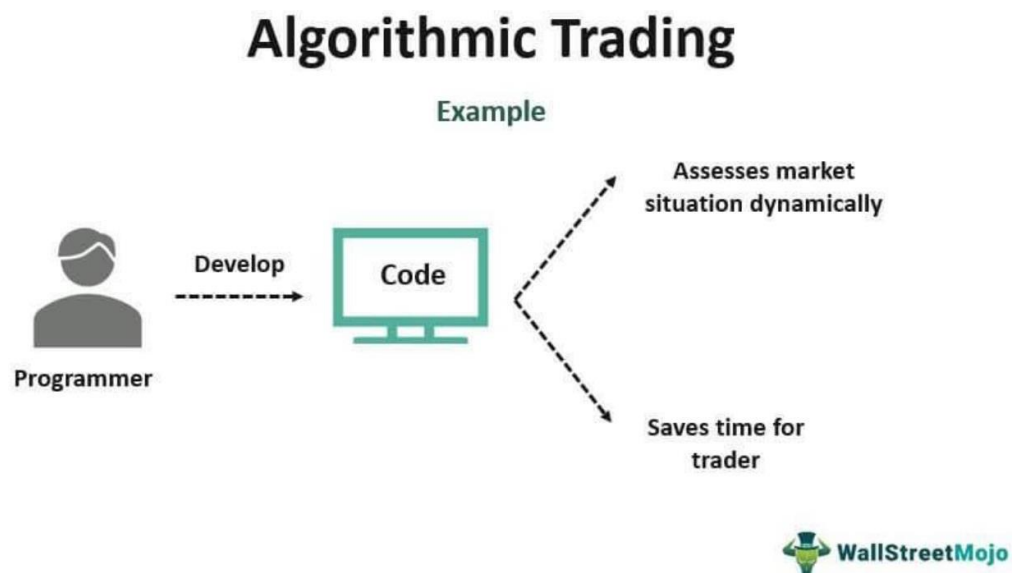


FIG 1.3 Algo-trading outline

Chapter 2

OBJECTIVES

2.1 REDUCE HUMAN ERRORS

By controlling emotions, traders usually have a relaxed time following the plan. The algo trading system minimizes emotions during forex trading in India. As orders are completed by the program automatically as soon as trading conditions are fulfilled, the trader will not have to change questions about trade.

Along with helping traders who lack confidence in making decisions, algo trading can restrain those who have a tendency to overtrade and buy and sell at any perceived opportunity.

Algorithmic trading is trading on the stock exchange with the help of computer programs according to a certain algorithm.

Algo trading programs allow you to get high profits when trading on the exchange, while the “human factor” in making decisions on the conclusion of a transaction is minimized – this approach helps to get rid of emotional transactions. The money management system helps to determine the level of acceptable risk in a trade.

As per the experts, the main reason why most traders have been attracted to Algo trading in shares is the fact that they eliminate human error. Being able to control emotions such as greed and fear is the biggest task for any trader, an issue that algo trading helps avoid.

2.2 TRADE AT BEST POSSIBLE PRICE

Algo-trading allow traders to enter trade at best possible price, because we use set of instruction which make the code to trade instantly at best possible price.

It is impossible to traders to enter a trade manually always in live market at best possible time.

Algorithmic trading uses computer algorithms to assess market data and make orders. A human trader may take seconds or minutes to answer, whereas an

algorithm will instantly make a decision. This is particularly helpful in turbulent markets when prices might move fast, and your broker may not react soon enough.

2.3 REDUCED TRANSACTION COST

Technology has been included in trading activities through algo trading, significantly lowering transaction costs. Traders do not have to be hooked to their gadgets all the time to keep track of trades. Instead, a collection of instructions that track trade activity are used to programme the system. Without constant supervision, trades can be made, saving money and time that would have been spent continuously watching trading activity.

Lowering the time between submitting an order and executing it decreases expenses since there's less danger in holding assets until they're sold at a profit.

If you are acquainted with Walmart or Target, you know how costly it is to stock shelves and pay people. Automated traders are not paid unless they're successful; thus, there are no additional costs (for example, taxes).

2.4 MORE ACCURACY

There is less human interaction, which is a big advantage of algo trading. This implies a considerable reduction in the likelihood of errors.

In addition, the algorithms are double-checked to ensure that human errors do not affect them.

Although a trader can misjudge the technical indications and analyse them improperly, computer systems do not make such mistakes.

Chapter 3

LITERATURE SURVEY

Literature survey is the most important step in software development process. Before developing the tool, it is necessary to determine the time factor, economy, and company strength. Once these things are satisfied, then the next step is to determine which operating system and language can be used for developing the tool. Once the programmers start building the tool the programmers need lot of external support. This support can be obtained from senior programmers, from book or from websites. Before building the system, the above consideration is taken into account for developing the proposed system. The major part of the project development sector considers and fully survey all the required needs for developing the project. For every project Literature survey is the most important sector in software development process. Before developing the tools and the associated designing it is necessary to determine and survey the time factor, resource requirement, man power, economy, and company strength. Once these things are satisfied and fully surveyed, then the next step is to determine about the software specifications in the respective system such as what type of operating system the project would require, and what are all the necessary software are needed to proceed with the next step such as developing the tools, and the associated operations.

Novel Data-Driven Fuzzy Algorithmic Volatility Forecasting Models with Applications to Algorithmic Trading

Proposed by A. Thavaneswaran Department of Statistics University of Manitoba Winnipeg, Canada thavane@umanitoba.ca You Liang Department of Mathematics Ryerson University Toronto, Canada you.liang@ryerson.ca Zimo Zhu Department of Statistics University of Manitoba Winnipeg, Canada zhuz3456@myumanitoba.ca Ruppia K. Thulasiram Department of Computer Science University of Manitoba Winnipeg, Canada tulsi@cs.umanitoba.ca

The explosion of algorithmic trading has been one of the most prominent trends in the finance industry. In this paper, two strategies for algorithmic trading such as Bollinger bands and the simple moving average (SMA) crossover strategy are studied in the fuzzy settings.

The commonly used Bollinger bands trading strategy assumes that the difference between an asset's price and its SMA is normally distributed. However, it is shown that a datadriven t distribution is more appropriate to model the difference between an asset's price and its SMA. A novel data-driven fuzzy Bollinger bands strategy is proposed for algo trading. A good strategy should have a good algo return on investment with low algo volatility. Therefore, forecasting algo volatility and identifying an appropriate distribution of algo returns play a crucial role in algo trading.

Exploring the financial indicators to improve the pattern recognition of economic data based on machine learning

Proposed By Xiaohui Wei, Wanling Chen, Xiao Li

Various economic data in the financial market need to be pattern-recognized to improve the efficiency of economic data pattern recognition, further improve the accuracy of economic-related decisions, and promote stable economic development. Based on machine learning technology, this study establishes a statistical model by establishing a multiple regression model to extract financial indicators that have significant effects on the financing trade of listed companies. Moreover, this study provides a preliminary empirical model for judging whether a company conducts financing trade based on some company's financial indicators

and uses data to verify the consistency of the model. In addition, this study conducts research and demonstration of the algorithm model of this research through empirical research.

The research results show that the model shows high reliability and validity in accurately identifying whether the enterprise has the characteristics of conducting financing trade.

Portfolio Trading System of Digital Currencies: A Deep Reinforcement Learning with Multidimensional Attention Gating Mechanism

Proposed by Liguu Wenga , Xudong Suna , Min Xia*,a, Jia Liua , Yiqing Xub

As a hot topic in the financial engineering, the portfolio optimization aims to increase investors' wealth. In this paper, a portfolio management system based on deep-reinforcement learning is proposed. In contrast to inflexible traditional methods, the proposed system achieves a better trading strategy through Reinforcement learning. The reward signal of Reinforcement learning is updated by action weights from Deep learning networks.

Low price, high price and close price constitute the inputs, but the importance of these three features is quite different. Traditional methods and the classical CNN can't deal with these three features separately, but in our method, a designed depth convolution is proposed to deal with these three features separately. In a virtual currency market, the price rise only occurs in a flash. Traditional methods and CNN networks can't accurately judge the critical time. In order to solve this problem, a three-dimensional attention gating network is proposed and it gives higher weights on rising moments and assets.

Deep reinforcement learning for portfolio management of markets with a dynamic number of assets

Proposed by Carlos Betancourt , Wen-Hui Chen

This work proposes a novel portfolio management method using deep reinforcement learning on markets with a dynamic number of assets. This problem is especially important in cryptocurrency markets, which already support the trading of hundreds of assets with new ones being added every month.

A novel neural network architecture is proposed, which is trained using deep reinforcement learning. Our architecture considers all assets in the market, and automatically adapts when new ones are suddenly introduced, making our method more general and sample-efficient than previous methods.

Further, transaction cost minimization is considered when formulating the problem. For this purpose, a novel algorithm to compute optimal transactions given a desired portfolio is integrated into the architecture. The proposed method was tested on a dataset of one of the largest cryptocurrency markets in the world, outperforming state-of-the-art methods, achieving average daily returns of over 24%.

A Survey on Stock Market Prediction using Machine Learning Approaches

Proposed by Shraddha Jain ,Megha Gupta

Stock Market prediction is a category of time series prediction which extremely challenging due to the dependence of stock prices on several financial, socioeconomic, and political parameters etc. Moreover, small inaccuracies in stock market price predictions may result in huge losses to firms which use stock market price prediction results for financial analysis and investments. Off late, soft computing techniques are being used widely for stock market prediction due to relatively higher accuracy compared to conventional statistical techniques.

The proposed work investigates the necessity of soft computing approaches for stock market prediction, the various soft computing approaches relevant to the area, previous work in the area with their salient features and the major challenges or the problem domain pertaining to the area. The main focus is on neural networks and fuzzy logic for developing a predictive model. Stock market is highly volatile and is unarguably very difficult to predict accurately based on certain parameters. This paper presents a comprehensive survey on the various techniques explored for high accuracy prediction emphasizing on the salient features.

Dynamic Data Science Applications in Optimal Profit Algorithmic Trading

Proposed by: You Liang Department of Mathematics Ryerson University Toronto, Canada you.liang@ryerson.ca Aerambamoorthy Thavaneswaran Department of Statistics University of Manitoba Winnipeg, Canada thavane@umanitoba.ca Na Yu Department of Statistics Ryerson University Toronto, Canada nayu@ryerson.ca Md.

Erfanul Hoque Department of Statistics University of Manitoba Winnipeg, Canada
hoqueme@myumanitoba.ca Ruppa K. Thulasiram Department of Computer
Science University of Manitoba Winnipeg, Canada tulsi@cs.umanitoba.ca

Many of the challenges and opportunities of data science in finance involve recursive smoothing, forecasting, filtering and pattern mining. Recently there has been a growing interest in using filtered estimates for dynamic hedge ratios for pairs trading. Moreover, rolling estimates and forecasts are used for pattern mining in technical analysis. Kalman filtering algorithms were successfully applied in pairs trading with only two co-integrated assets.

In this paper, pairs trading strategy is extended to multiple trading (with more than two assets) strategy. Recently proposed non-Gaussian maximum informative filtering algorithms for dynamic state space models are used to obtain the filtered estimates of hedge ratios and applied in multiple trading. It is shown that the proposed multiple trading strategy outperforms (with higher profits) the commonly used pairs trading strategy using real data.

Chapter 4

PROPOSED SYSTEM

Best Execution: Trades are often executed at the best possible prices. Low Latency: Trade order placement is instant and accurate (there is a high chance of execution at the desired levels).

Trades are timed correctly and instantly to avoid significant price changes. Reduced transaction costs. Simultaneous automated checks on multiple market conditions.

No Human Error: Reduced risk of manual errors or mistakes when placing trades. Also negates human traders; tendency to be swayed by emotional and psychological factors. Back testing: Algo-trading can be back tested using available historical and real-time data to see if it is a viable trading strategy.

Using these two simple instructions, a computer program will automatically monitor the stock price (and the moving average indicators) and place the buy and sell orders when the defined conditions are met. The trader no longer needs to monitor live prices and graphs or put in the orders manually. The algorithmic trading system does this automatically by correctly identifying the trading opportunity.

We also use Genetic algorithm to isolate and find best SMA (Simple Moving Average) parameter and using those parameter we will be able to find the market trend and be able to place trade

And then we use RSI (Relative Strength Index) to find the Overbought or Oversold level. And using this we will be able to make calculated decision

We also use RNN Recurrent Neural Networks are deep learning models, typically used to solve problems with sequential input data such as time series

Chapter 5

SYSTEM ARCHITECTURE

FIG 5.1 SYSTEM ARCHITECTURE

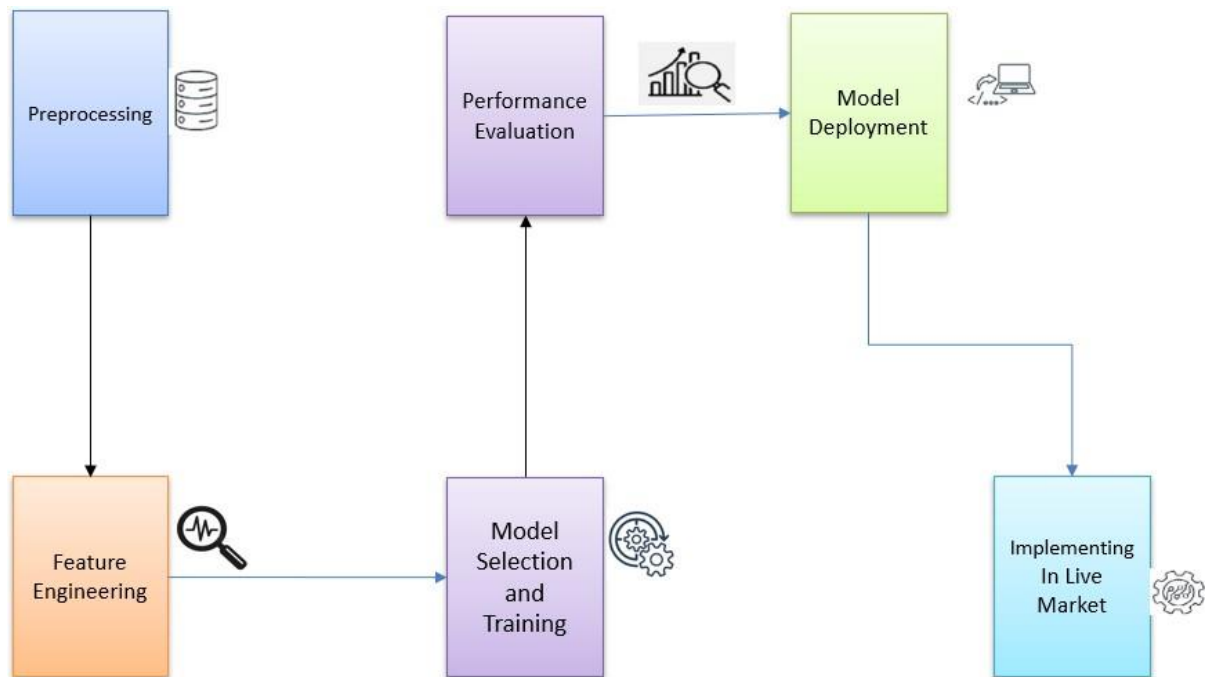


FIG 5.1 System Architecture

Chapter 6

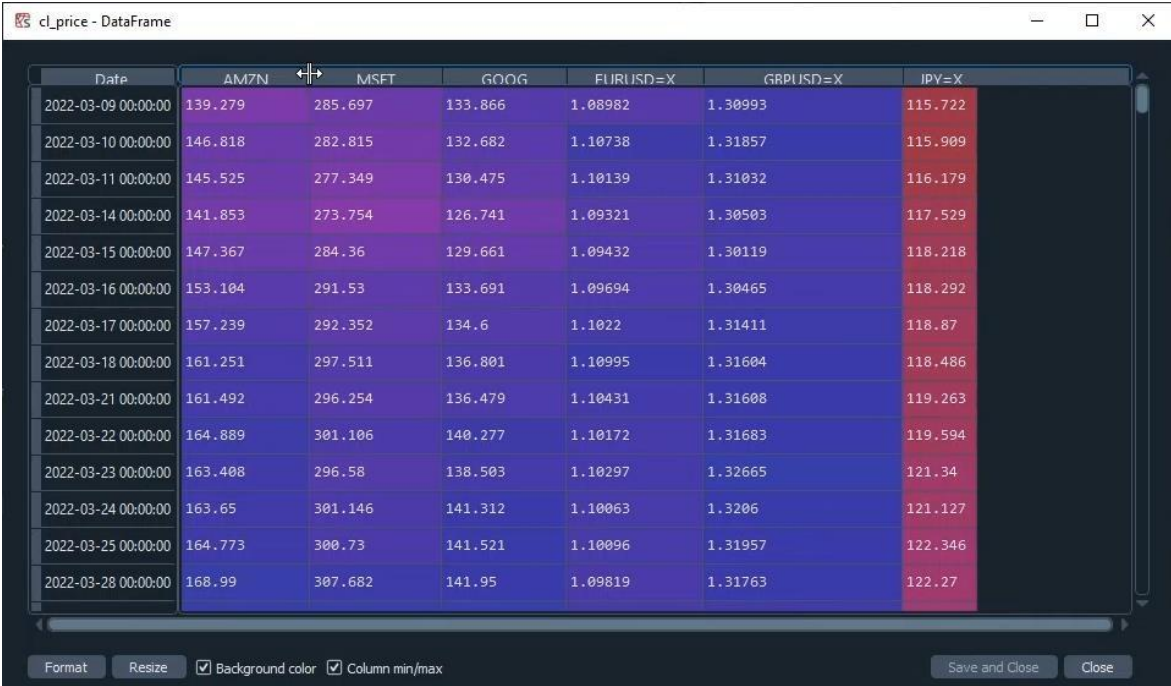
MODULE

6.1 PRE-PROCESSING

This module we collect historical data. We collect numerical data from Y-finance or from Brokage account. Applying matplotlib we convert numerical data into chart. We gather live or historical data from Y-finance (Yahoo Finance).

We loop over all the files and load them to data frames while also adding some trade time-based features.

Then we add more features using TA-Lib, a library for technical analysis in python.



Date	AMZN	MSFT	GOOG	EURUSD=X	GBPUSD=X	JPY=X
2022-03-09 00:00:00	139.279	285.697	133.866	1.08982	1.30993	115.722
2022-03-10 00:00:00	146.818	282.815	132.682	1.10738	1.31857	115.909
2022-03-11 00:00:00	145.525	277.349	130.475	1.10139	1.31032	116.179
2022-03-14 00:00:00	141.853	273.754	126.741	1.09321	1.30503	117.529
2022-03-15 00:00:00	147.367	284.36	129.661	1.09432	1.30119	118.218
2022-03-16 00:00:00	153.104	291.53	133.691	1.09694	1.30465	118.292
2022-03-17 00:00:00	157.239	292.352	134.6	1.1022	1.31411	118.87
2022-03-18 00:00:00	161.251	297.511	136.801	1.10995	1.31604	118.486
2022-03-21 00:00:00	161.492	296.254	136.479	1.10431	1.31608	119.263
2022-03-22 00:00:00	164.889	301.106	140.277	1.10172	1.31683	119.594
2022-03-23 00:00:00	163.408	296.58	138.503	1.10297	1.32665	121.34
2022-03-24 00:00:00	163.65	301.146	141.312	1.10063	1.3206	121.127
2022-03-25 00:00:00	164.773	300.73	141.521	1.10096	1.31957	122.346
2022-03-28 00:00:00	168.99	307.682	141.95	1.09819	1.31763	122.27

Fig 6.1 Historical Price

We must do quite a bit of cleaning up to do in order to get data in a nice format. Most of the work is on manipulating the data into the format you need for observing correlations and then making predictions.

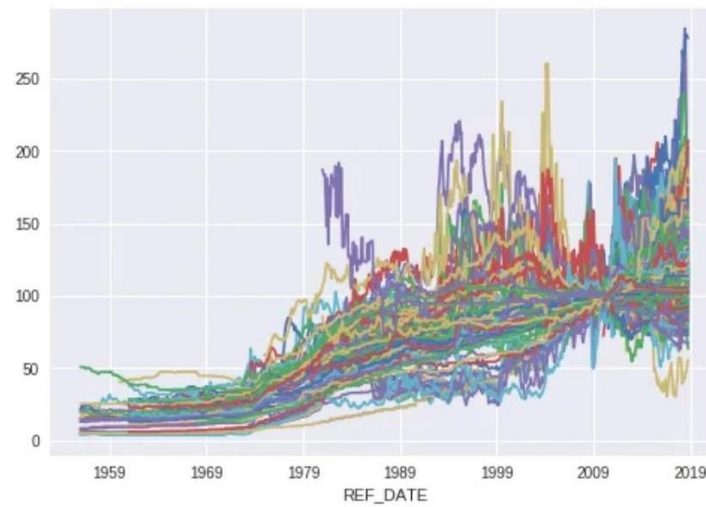


FIG 6.2 Date before Pre-processing

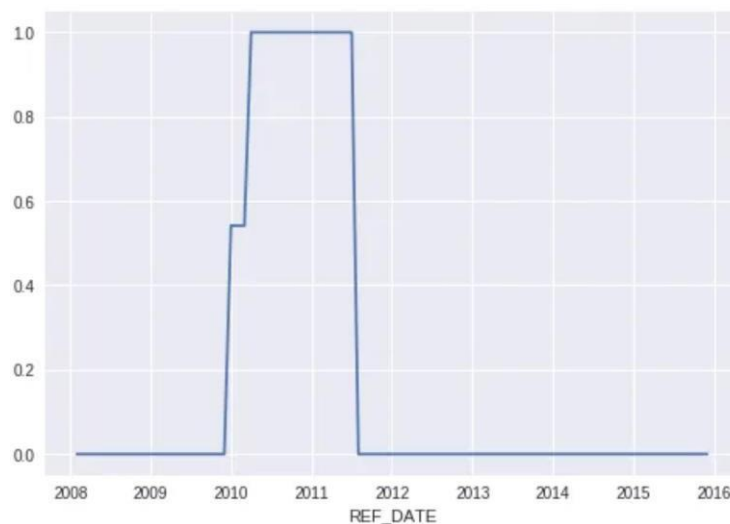


FIG 6.3 Data after Pre-processing

6.2 FEATURE ENGINEERING

This module consists of technical analysis calculations. SMA (Simple Moving Average) can be used to identify aspects of an asset's overall trend. MSD (Moving Standard Deviation) is a tool used in Feature Engineering to measure volatility.

The Relative Strength Index (RSI) is a well-known momentum-based oscillator that is used to measure the speed (velocity) as well as the change (magnitude) of directional price movements

6.2.1 SIMPLE MOVING AVERAGE (SMA)

Simple moving averages calculate the average of a range of prices by the number of periods in that range. A simple moving average is an indicator that can help determine whether an asset's price will continue or reverse a bullish or bearish trend.

SMA is an arithmetic moving average calculated by summing the last prices and then dividing this number by the number of time periods in the calculation average.

The formula for SMA is:

$$SMA = \frac{A_1 + A_2 + \dots + A_n}{n}$$

where:

A_n = the price of an asset at period n

n = the number of total periods



FIG 6.4 Simple Moving Average

	close	SMA 15	SMA 60
Date			
2004-02-23	1.257197	1.267882	1.251294
2004-02-24	1.267797	1.268808	1.252306
2004-02-25	1.249703	1.269761	1.253288
2004-02-26	1.244803	1.269475	1.253911
2004-02-27	1.249407	1.268649	1.254523
...	...	...	...
2023-04-03	1.080334	1.075897	1.074349
2023-04-04	1.091227	1.076695	1.074594
2023-04-05	1.096011	1.077943	1.074892
2023-04-06	1.090334	1.079493	1.075261
2023-04-07	1.090631	1.081626	1.075490

4961 rows x 3 columns

FIG 6.5 Numerical Data of SMA

6.2.2 RELATIVE STRENGTH INDEX

The Relative Strength Index (RSI) is a well-known momentum-based oscillator that is used to measure the speed (velocity) as well as the change (magnitude) of directional price movements.

Essentially, the RSI, when displayed on a chart, provides a visual means of tracking the current and historical strength and weakness of a particular market.



FIG 6.6 Relative Strength Index

Calculating RSI:

$$RSI = 100 - \left[\frac{100}{1 + \frac{\text{Average gain}}{\text{Average loss}}} \right]$$

Datetime	Open	High	Low	Close	Adj Close	Volume	RSI
2023-03-08 14:50:00-05:00	93.48	93.51	93.39	93.455	93.455	357917	57.676
2023-03-08 14:55:00-05:00	93.46	93.47	93.39	93.4671	93.4671	366064	58.0883
2023-03-08 15:00:00-05:00	93.4631	93.6	93.45	93.46	93.46	359507	57.7328
2023-03-08 15:05:00-05:00	93.455	93.46	93.22	93.28	93.28	353843	49.4703
2023-03-08 15:10:00-05:00	93.28	93.36	93.22	93.3388	93.3388	308614	51.0924
2023-03-08 15:15:00-05:00	93.335	93.39	93.28	93.335	93.335	271862	51.7199
2023-03-08 15:20:00-05:00	93.335	93.44	93.315	93.3455	93.3455	334706	52.1928
2023-03-08 15:25:00-05:00	93.3499	93.47	93.31	93.465	93.465	312171	57.318
2023-03-08 15:30:00-05:00	93.465	93.505	93.405	93.5	93.5	357490	58.7142
2023-03-08 15:35:00-05:00	93.51	93.755	93.48	93.69	93.69	671828	65.3415
2023-03-08 15:40:00-05:00	93.69	93.89	93.635	93.8601	93.8601	693309	69.9865
2023-03-08 15:45:00-05:00	93.86	93.99	93.86	93.9619	93.9619	768982	72.3729
2023-03-08 15:50:00-05:00	93.96	94.03	93.69	93.73	93.73	1040973	60.5603
2023-03-08 15:55:00-05:00	93.72	93.97	93.7	93.92	93.92	1505021	65.5252

FIG 6.7 Numerical data for RSI

SPY Close Prices & Relative Strength Index RSI(14) Strategy Indicator

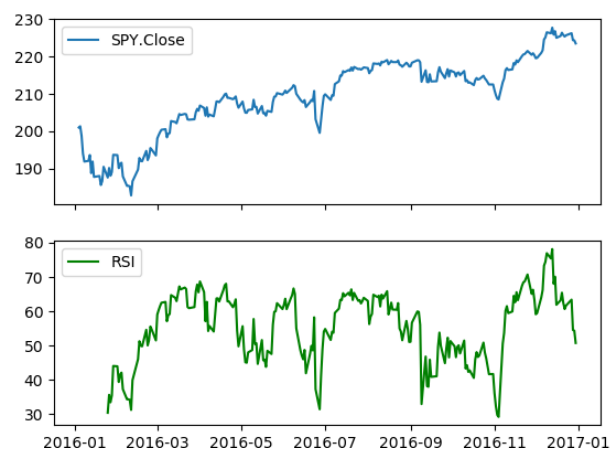


FIG 6.8 RSI Data

6.2.3 DEEP LEARNING IMPLEMENTATION

This module consist of Implementation of Deep Learning Neural Network. We implement RNN(Recurrent Neural Network), which is used to learn order dependence in sequence prediction problems.

Recurrent Neural Network are extremely powerful time-series model . RNN are the sigmoid activation functions they output a value between 0 or 1 and in every of the cases it is either 0 or 1

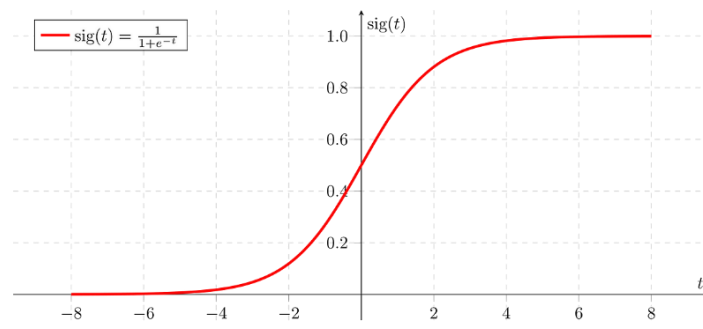


FIG 6.9 Sigmoid Function

Chapter 7

RESULT AND DISCUSSION

```
from tensorflow.keras.callbacks import EarlyStopping

# additional Library
import tensorflow
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, LSTM, Dropout

nb_hidden_layer = 3

# model Initialization
classifier = Sequential()

# adding LSTM layer
classifier.add(LSTM(units = 10, return_sequences = True,
                    input_shape = (X_train_3d.shape[1],X_train_3d.shape[2])))

# Loop to add LSTM layer
for _ in range(1):
    classifier.add(LSTM(units = 10, return_sequences = True))
    classifier.add(Dropout(0.15))

# Last lstm layer
classifier.add(LSTM(units = 10, return_sequences = False))

# Dense Output Layer
classifier.add(Dense(1, activation="sigmoid"))

# compiling the Model
classifier.compile(loss="binary_crossentropy", optimizer="adam")

# Training with early Stop
early_stop = EarlyStopping(verbose=1, patience = 5)

classifier.fit(X_train_3d, y_train_3d, validation_data=(X_test_3d, y_test_3d), epochs=100, callbacks=[ear
```

FIG 7.1 RNN Model

```
Epoch 1/100
132/132 [=====] - 10s 30ms/step - loss: 0.6950 - val_loss: 0.6925
Epoch 2/100
132/132 [=====] - 3s 26ms/step - loss: 0.6933 - val_loss: 0.6927
Epoch 3/100
132/132 [=====] - 3s 22ms/step - loss: 0.6934 - val_loss: 0.6926
Epoch 4/100
132/132 [=====] - 3s 20ms/step - loss: 0.6932 - val_loss: 0.6925
Epoch 5/100
132/132 [=====] - 3s 21ms/step - loss: 0.6929 - val_loss: 0.6924
Epoch 6/100
132/132 [=====] - 3s 22ms/step - loss: 0.6929 - val_loss: 0.6923
Epoch 7/100
132/132 [=====] - 3s 25ms/step - loss: 0.6931 - val_loss: 0.6923
Epoch 8/100
132/132 [=====] - 3s 20ms/step - loss: 0.6928 - val_loss: 0.6926
Epoch 9/100
132/132 [=====] - 3s 20ms/step - loss: 0.6926 - val_loss: 0.6922
Epoch 10/100
132/132 [=====] - 3s 20ms/step - loss: 0.6922 - val_loss: 0.6923
Epoch 11/100
132/132 [=====] - 3s 26ms/step - loss: 0.6928 - val_loss: 0.6925
Epoch 12/100
132/132 [=====] - 3s 21ms/step - loss: 0.6923 - val_loss: 0.6927
Epoch 13/100
132/132 [=====] - 3s 20ms/step - loss: 0.6924 - val_loss: 0.6925
Epoch 14/100
132/132 [=====] - 4s 28ms/step - loss: 0.6918 - val_loss: 0.6927
Epoch 14: early stopping
<keras.callbacks.History at 0x7feccc74f790>
```

FIG 7.2 Achieved Result

Here in RNN model we have use a function called early stopping as our desired output has been reached instead of running until the end this saves us some time.



FIG 7.3 Values in Chart

```

result = RNN("EURUSD=X", nb_model=15)
result

[*****100%*****] 1 of 1 completed
77/77 [*****] - 9s 47ms/step - loss: 0.6891 - val_loss: 0.6946
77/77 [*****] - 1s 5ms/step
19/19 [*****] - 0s 5ms/step
(Date
2021-08-19 -0.042775
2021-08-20 -0.056116
2021-08-21 0.008790
2021-08-22 -0.008509
2021-08-23 -0.004552
...
2022-06-02 -0.022430
2022-06-03 0.025046
2022-06-04 -0.004327
2022-06-05 -0.002472
2022-06-06 0.048953
Name: strategy, Length: 292, dtype: float64,
Date
2022-06-07 -0.006860
2022-06-08 -0.030207
2022-06-09 -0.003388
2022-06-10 -0.034146
2022-06-11 -0.024859
...
2023-04-05 0.000351
2023-04-06 -0.004750
2023-04-07 -0.004218
2023-04-08 0.000785
2023-04-09 -0.001562
Name: strategy, Length: 307, dtype: float64)

```

FIG 7.4 Data from Live Market

Conclusion

In this work we have made trading decision making workflow has been proposed to generate effective buy or sell signals. In this proposal, the trading rules with the predictive power of deep learning neural networks.

From among all the deep learning techniques tested we are using the ones that performed best.

The good performance of the DNN is well known is Forex market prediction and for that reason, has been extensively used in previous work.

The good performance suggests that for smaller period of time the linear model is suitable for prediction purpose.

Another possible improvement of the strategy is to incorporate trend information by addressing the problem as a classification task.

We also intend to address the exploration of more technical analysis strategies and new trading rules. Finally, in order to gain more insight about the predictive models we have the DNN model.

REFERENCE

[1] Explainable Machine Learning Exploiting News and Domain-Specific Lexicon for Stock Market Forecasting SALVATORE M. CARTA ¹, (Member, IEEE), SERGIO CONSOLI ², LUCA PIRAS¹, ALESSANDRO SEBASTIAN PODDA ¹, AND DIEGO REFORGIATO RECUPERO ¹ ¹Department of Mathematics and Computer Science, University of Cagliari, 09124 Cagliari, Italy ²European Commission, Joint Research Centre (DG-JRC), I-21027 Ispra, Italy

[2] Intelligent Algorithmic Trading Strategy Using Reinforcement Learning and Directional Change MONIRA ESSA ALOUD AND NORA ALKHAMEES Department of Management Information System, College of Business Administration, King Saud University, Riyadh 11451, Saudi Arabia Corresponding author: Monira Essa Aloud (mealoud@ksu.edu.sa)

[3] Dynamic Data Science Applications in Optimal Profit Algorithmic Trading You Liang Department of Mathematics Ryerson University Toronto, Canada you.liang@ryerson.ca Aerambamoorthy Thavaneswaran Department of Statistics University of Manitoba Winnipeg, Canada thavane@umanitoba.ca Na Yu Department of Statistics Ryerson University Toronto, Canada nayu@ryerson.ca Md. Erfanul Hoque Department of Statistics University of Manitoba Winnipeg, Canada hoqueme@myumanitoba.ca Ruppa K. Thulasiram Department of Computer Science University of Manitoba Winnipeg, Canada

[4] A Survey on Stock Market Prediction using Machine Learning Approaches Shraddha Jain¹, Megha Gupta² M.Tech Scholar (CSE)¹, Department of Computer Science and Engineering, PCST, Indore (India) Assistant Professor (CSE)² Department of Computer Science and Engineering, PCST, Indore (India)

[5] Time-driven feature-aware jointly deep reinforcement learning for financial signal representation and algorithmic trading Kai Lei ^{a,b}, Bing Zhanga, Yu Li ^a, 19 Min Yangc, Ying Shena,* ^a Shenzhen Key Lab for Information Centric Networking & Blockchain Technology (ICNLAB), School of Electronics and Computer Engineering (SECE), Peking University, Shenzhen 518055, China ^b PCL Research Center of Networks and Communications, Peng Cheng Laboratory, Shenzhen,

China c Shenzhen Institutes of Advanced Technology, University of the Chinese Academy of Sciences, China

[6] Deep reinforcement learning for portfolio management of markets with a dynamic number of assets Carlos Betancourt * , Wen-Hui Chen Graduate Institute of Automation Technology, National Taipei University of Technology, Taipei, Taiwan

[7] Technical analysis strategy optimization using a machine learning approach in stock market indices Jordan Ayala a , Miguel García-Torres b,* , José Luis Vázquez Noguera a , Francisco Gómez-Vela b , Federico Divina b a Polytechnic School, Universidad Nacional de Asunción, P.O.Box: 2111 SL, Paraguay b Division of Computer Science, Universidad Pablo de Olavide, ES-41013, Seville, Spain

[8] Portfolio Trading System of Digital Currencies: A Deep Reinforcement Learning with Multidimensional Attention Gating Mechanism Liguó Weng a , Xudong Sun a , Min Xia*,a, Jia Li a , Yiqing Xu a Jiangsu Key Laboratory of Big Data Analysis Technology, Nanjing University of Information Science and Technology, Nanjing, 210044, China bCollege of Information Science and Technology, Nanjing Forestry University, Nanjing, 210037, China

[9] Novel Data-Driven Fuzzy Algorithmic Volatility Forecasting Models with Applications to Algorithmic Trading A. Thavaneswaran Department of Statistics University of Manitoba Winnipeg, Canada thavane@umanitoba.ca You Liang Department of Mathematics Ryerson University Toronto, Canada you.liang@ryerson.ca Zimo Zhu Department of Statistics University of Manitoba Winnipeg, Canada zhuz3456@myumanitoba.ca Ruppa K. Thulasiram Department of Computer Science University of Manitoba Winnipeg, Canada tulsi@cs.umanitoba.ca

[10] Exploring the financial indicators to improve the pattern recognition of economic data based on machine learning Xiaohui Wei¹ • Wanling Chen² • Xiao Li^{3,4,5} Received: 10 March 2020 / Accepted: 4 June 2020 Springer-Verlag London Ltd., part of Springer Nature 2020

[11] Modelling Stock Markets by Multi-agent Reinforcement Learning Johann Lussange¹ Ivan Lazarevich^{1,2} Sacha Bourgeois-Gironde^{3,4} Stefano Palminteri⁵

· Boris Gutkin^{1,6} Accepted: 31 July 2020 © Springer Science+Business Media, LLC,
part of Springer Nature 2020