A SHORT-TERM INTERNSHIP REPORT ON

ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

BY

GURRAM SIRISHA

III DATA SCIENCE

Under the Esteemed Guidance of

Mr. G.V.S.S PRASANTH SIR

Tutor of Artificial Intelligence & Machine Learning

ADITYA DEGREE COLLEGE, TUNI

(Affiliated to ADIKAVI NANNAYA UNIVERSITY)

Tuni-533401, Kakinada District, ANDHRA PRADESH

2022-2025

ADITYA DEGREE COLLEGE

DECLARATION BY THE STUDENT

I hereby declare that the work described in this Short-term Internship, entitled "**Artificial Intelligence & Machine Learning**" which is being submitted by me in partial fulfilment of the requirements for the award of degree of **Bachelor of Science** from the Department of **Data Science** to Aditya Degree College,Tuni under the guidance of **Mr. G.V.S.S PRASANTH Sir** tutor of **Artificial Intelligence & Machine Learning** in Aditya Degree College , Tuni.

Place:TUNI **(GURRAM SIRISHA)**

Date:

ADITYA DEGREE COLLEGE

CERTIFICATE FROM THE SUPERVISOR

This is to certify that the Short Term Internship entitled, "**ARTIFICIAL INTELLIGENCE & MACHINE LEARNING",** that is being submitted by **GURRAM SIRISHA** bearing **221177156164** of **III Data Science** , which is being submitted by us in partial fulfilment of the requirements for the award of degree of **Bachelor of Science** from the Department of Data science to Aditya Degree College, bonified work carried out by him under my guidance and Supervision.

(Mr. G.V.S.S PRASANTH SIR)

ACKNOWLEDGEMENT

No endeavor is completed without the valuable support of others.

I would like to take this opportunity to extend my sincere gratitude to all those who have contributed to the successful completion of this Short-Term Internship Project Report.

I express my deep sense of gratitude to **SMT. M. DEEPTHI, Principal**, for his Efforts and for giving us permission for carrying out this Long-Term Internship.

I feel deeply honored in expressing my sincere thanks to Mr. G.V.S.S Prashanth Sir tutor of ULearn Visakhapatnam for making the resources available at right time and providing valuable insights leading to the successful completion of my short-Term Internship Project Report.

Finally, I thank all the faculty members of our department who contributed their valuable suggestions in completion of Short-Term Internship report and I also put my sincere thanks to My Parents who stood with me during the whole Short-Term Internship.

(G. SIRISHA)

INTRODUCTION

Stock Price Prediction is the task of forecasting future stock prices based on historical data and various market indicators. It involves using statistical models and machine learning algorithms to analyze financial data and make predictions about the future performance of a stock.

The goal of stock price prediction is to help investors make informed investment decisions by providing a forecast of future stock prices. Stock price prediction is important for value investments in the stock market. In particular, short-term prediction that exploits financial news articles is promising in recent years.

In this paper, we propose a novel deep neural network DP-LSTM for stock price prediction, which incorporates the news articles as hidden information and integrates difference news sources through the differential privacy mechanism. First, based on the autoregressive moving average model (ARMA), a sentiment-ARMA is formulated by taking into consideration the information of financial news articles in the model.Then, an LSTM-based deep neural network is designed, which consists of three components: LSTM, VADER model and differential privacy (DP) mechanism. The proposed DP-LSTM scheme can reduce prediction errors and increase the robustness. Extensive experiments on S&P 500 stocks show that the proposed DP-LSTM achieves 0.32% improvement in mean MPA of prediction result, and (ii) for the prediction of the market index S&P 500, we achieve up to 65.79% improvement in MSE. Investors may judge on the basis of

technical analysis, such as charts of a company, market indices, and on textual information such as news blogs or newspapers.

It is however difficult for investors to analyze and predict market trends according to all of these information . A lot of artificial intelligence approaches have been investigated to automatically predict those trends .

To solve the above problem, in this paper we use sentiment analysis to extract information from textual information. Sentiment analysis is the automated process of understanding an opinion about a given subject from news articles.

The analyzed data quantifies reactions or sentiments of the general public toward people, ideas or certain products and reveal the information’s contextual polarity. Sentiment analysis allows us to understand if newspapers are talking positively or negatively about the financial market, get key insights about the stock’s future trend market.

LEARNING OUTCOME OF SHORT TERM INTERNSHIP

ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

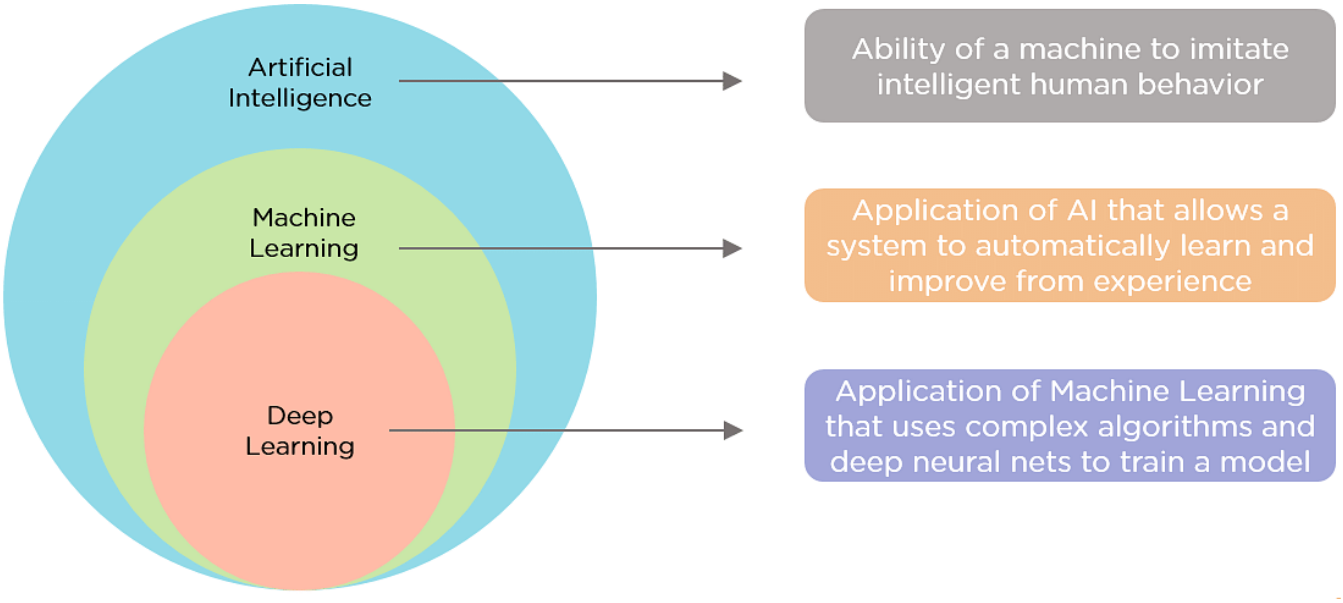
INTRODUCTION

What is AI?

- Ai is an branch of computer science it is the simulation of human intelligence process by machines is called AI
- Human intelligence which performs machines is called AI

In AI how many subsets are there?

- Machine learning
- Deep learning



Sub fields

- Artificial neural networks

- Cognative computing
- Natural language processing

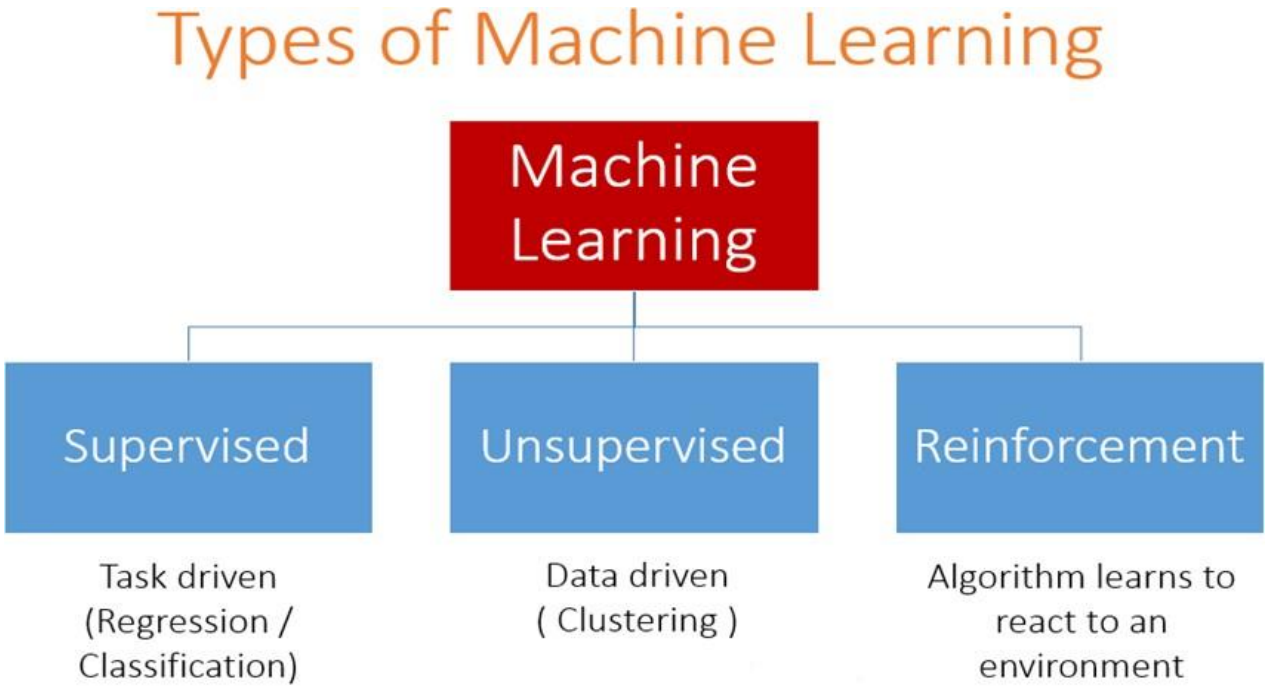
What is Machine learning?

- Its a subset of ai which focus on the use of data and algorithms imitate the way that human learn and fradually increasing its accuracy
- It learns from data & solve the problems

Types of machine learning:

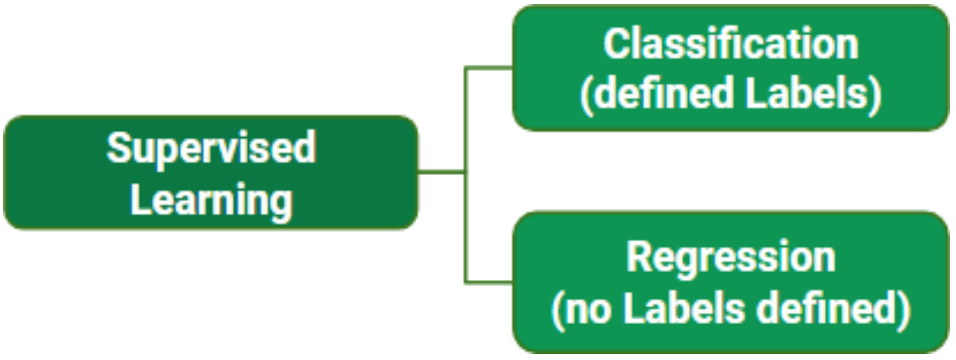
There are 3types of ML

- Supervised learning - its an labelled data or structered data
- Unsupervised learning - its an unlabelled data or unstructered data
- Reinforcement learning -it uses both structured data and unstructured data



SUPERVISERED LEARNING:

- Classification
- Regression



Classification:

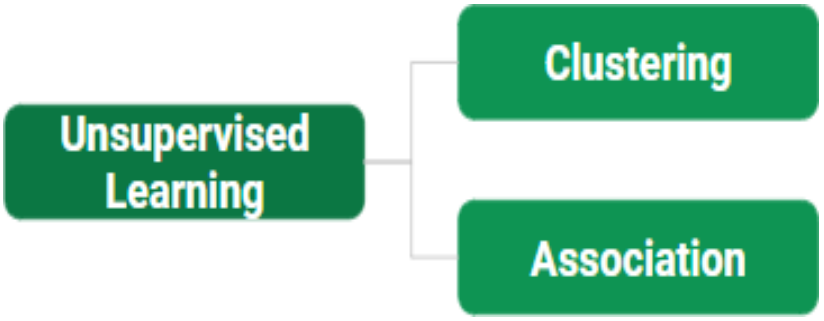
- KNN
- Support vector Machine
- Decision Tree
- Random Forest
- Navie Bays
- Neural Network
- Stochastic Gradient Descent

Regression:

- Linear regression
- Logistic Regression
- Polynomial Regression

UNSUPERVISED LEARNING:

- Clustering
- Association



Clustering:

- Principal Component Analysis
- Hierarchical Clustering
- Singular Value Decomposition
- Independent Component Analysis
- K-Means Clustering

Reinforcement Learning:

3 Ways of implementation

- Model based
- Policy based
- Value based

Models:

- Q-learning
- Markov Models

Programming Languages used in ML:

- Python
- R Language

Python:

Python is one of the easiest yet most useful programming languages which is widely used in the software industry. People use Python for Competitive Programming, Web Development, and creating software. Due to its easiest syntax, it is recommended for beginners who are new to the software engineering field. Its demand is growing at a very rapid pace due to its vast use cases in Modern Technological fields like Data Science, Machine learning, and Automation Tasks.

Mathematics:

- Linear Algebra
- Calculas
- Probability

Data Bases:

To store the structured or unstructured data in database to access the data by using Sql queries (structured query language)

- MangoDB
- My SQL

Visualization Tools:

- Qlick Sense
- Tableau
- PowerBI

Why we used this bi tools?

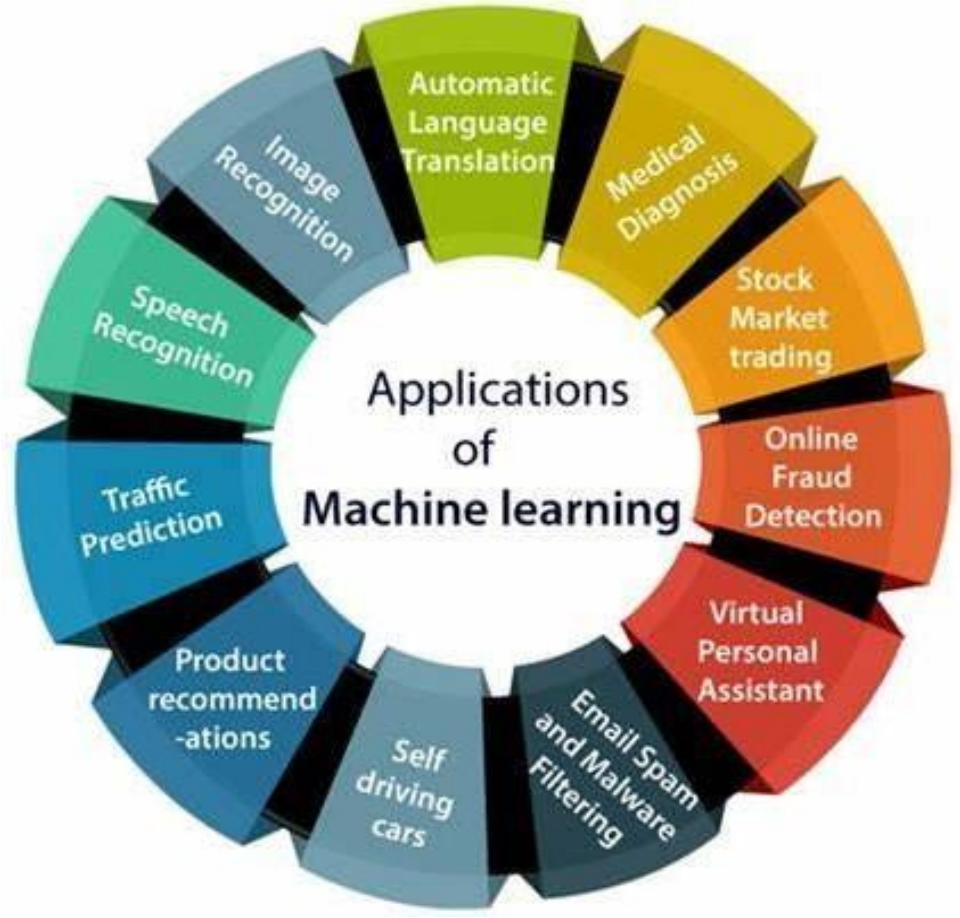
For data visulatation purpose
Bar Graphs
Pie charts
Histograms
Box plots
Scatterplot
Line charts

Steps required to train AI And ML models:

- Gathering Data
- Data prepration
- Choosing A model
- Training
- Evaluation
- Parameter tuning
- Prediction

Applications of Machine Learning:

- Self-Driving Cars
- Chatbot
- Image registration
- Speech Recognition
- Stock Market trading
- Email Spam Filtering
- Online fraud Detection



AI tools we used in our daily life:

1. Chatbots

A prominent example of this is **ChatGPT**. People started out using this chatbot just another online companion. However, you'll be surprised to know that, ChatGPT is actually an **artificial intelligence-powered chatbot.**

2. Microsoft Bing

While Google has always been the go-to search engine for almost everyone, Microsoft has now revamped Bing with artificial intelligence. The new AI Bing has been specially created to give the search engine the power to intelligently give nuanced responses by its AI. However, Bing also benefits from the new Chat mode.

3.Smart Compose, Quick Reply, and Grammar Check

If you use Gmail then you might have noticed a feature called Smart Compose. It suggests complete sentences based on the preceding line that you have written. It uses Artificial Intelligence to quickly compose your **email drafts with contextual accuracy** and correct grammar. I use it quite often and believe me, it's pretty helpful. There could be no better example of AI making life better and saving time on the other hand.

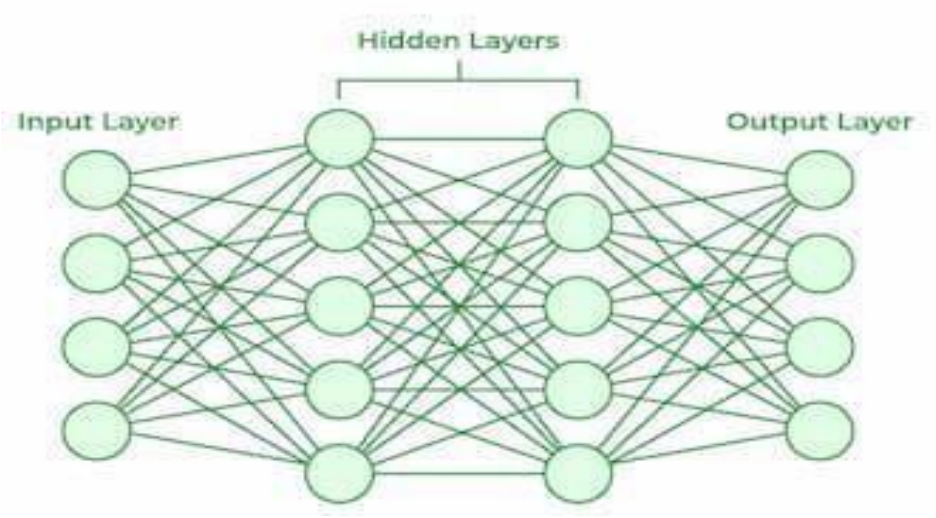
4.Google Lens and OCR

Google Lens is another Google service that is built on AI and has some great technology for fast and accurate optical recognition. It allows you to search for anything through images. Just **point the camera to a shoe or a plant** or an animal or

a text, and it can detect the type of subject and will provide precise information on that thing in just a few seconds.

What is neural Network?

Neural Networks are computational models that mimic the complex functions of the human brain. The neural networks consist of interconnected nodes or neurons that process and learn from data, enabling tasks such as pattern recognition and decision making in machine learning. The article explores more about neural networks, their working, architecture and more.



Input Layer: Each feature in the input layer is represented by a node on the network, which receives input data.
Weights and Connections: The weight of each neuronal connection indicates how strong the connection is. Throughout training, these weights are changed.
Hidden Layers: Each hidden layer neuron processes inputs by multiplying them by weights, adding them up, and then passing them through an activation function. By doing this, non-linearity is introduced, enabling the network to recognize intricate patterns.
Output: The final result is produced by repeating the process until the output layer is reached.

Back propagation:

- In machine learning, backpropagation is an effective algorithm used to trainartificial neural networks, especially in feed-forward neural networks.
- Backpropagation is an iterative algorithm, that helps to minimize the cost function by determining which weights and biases should be adjusted. During every epoch, the model learns by adapting the weights and biases tominimize the loss by moving down toward the gradient of the error. Thus, it involves the two most popular optimization algorithms, such as gradient descent or stochastic gradient descent.

Objection Detection:

Object detection is a technique that uses neural networks to localize and classify objects in images. This computer vision task has a wide range of applications, from medical imaging to self-driving cars.

Applications:

- Object reg
- Facial Recg
- Video Tracking
- Moment Detection
- Self Driving Cars
- Animal Detection
- Robotics

Steps involved in objection detection:

- Image preprocessing - resizing and normalization of image
- Feature extraction - it classifies the image
- Object localization - it locates the object
- Object classification - it identifies what type of object in image
-Post processing - it refining and eliminating the duplicate detections.

Gan - generative adversarial networks

-Gan was introduced by LAN goodfellow in 2014.
-Gan are algorithmic architecture that uses two neural networks pitting one against to other to generate new data that passes through the real data.

Applications:

- To generate photorealistic images
- Change facial expressions
- Create computer game scenes
- Visualize designs
- Create artwork

Diff b/w original image and generated images:
-Brightness
-Thickness
-Color
-Background
-Saturation
-unrealistic elements
-Quality
-Clarity
-Size
-Features

DEEP DREAM:

- Deep dream is one of the application of deep learning in computer vision.
- In this deep dream concept we used deep neural networks
- We are using CNN algorithm to find image patterns in images
- Deep dream software original image using deep CNN named as inception

PROCESS:

- If we take any image the deepdream will identify faces and there patterns inimage by using deep CNN algorithm to modify the images.
- once we trained this algorithm its reverse process takes place to changethe image patterns.
- This can be visualizations to understand the emergent structure of neuralnetwork and basis for the deep dream concept

DEEP FAKE:

It is digitally alterated image, video, or audio that replaces one person facewith another person face is called deep fake

Algorithms:

- Deep CNN
- GAN Model
- VGG16 and VGG19 - visual geometrical graphs

Training:
Jupyter notebook:

- required libraries
- importing datasets
- selecting input image
- targeted image
- training algorithm
- output

Examples:
Trump face -Putin Face Moffed
Girl Face – Celebrity Face.

DATA AUGMANTATION:

Data augmentation uses pre-existing data to create new data samples that can improve model optimization and generalizability.

- Data Augmantation is data analysis are techniques used to increased the amount of data by adding or slightly modifying copies in already existing dataor newly created synthetic data is called data augmantation.
- It is a set of techniques to artificially increase the amount of datagenerating new data points from existing data

WHY ITS IMPORTANT?

- It includes making small changes to data or using deep learning models to generate new data points.
- it is useful to improve the performance and outcomes of machine learning models by forming new and different examples to train datasets.

STEPS:

- Input data that feed to the data Augmantation pipeline
- The data Augmantation pipeline by sequential steps with different Augmantations
- Tf1-rotation
- Tf2-greyscale
  to rgbtf3-blur
- Tfn-flip
- The image is feed through the pipeline and processed through each step with different probablity
- After image is processed the human expert randomaly verifies the augmanted results and passes the feedback to the system
- After human expert verification the augmanted data is ready to train the ai training process

FOR IMAGE CLASSIFICATION AND SEGMENTATION:

Random
Rotatingre
Scalling
Vertical or horizontal flipping
ranslation
Cropping
zooming

PARAMETER SHARING AND TYPING:

It is an convolutional neural network model which is used to share the weights equally in neural networks is called parameter sharing and typing

It is an deep learning application

Parameter sharing is the method of sharing weights by all neurons in a particular feature map

ENSEMBLE METHODS:

- Ensemble methods are techniques that aim to improving the results in modelsby combining multiple model instead of using single model.
- The combined models increase the accuracy of the results.
- The most popular ensemble methods are bagging boosting

SENSEMBLE METHODS:

- BAGGING
- BOOSTING

BOOTSTACKING:

- The argumanted data is trained with multiple models in AI process the accuracyresults is more
- Ensemble methods are ideal for regression and classification where they reduce bias and variance to accuracy of models

BAYES THEORM:

A bayes theorm finds the probability of an event occuring given the probabilityof another event that has already occurs is called bayes theorm.

P(A/B) = P(B/A) P(A) / P(B)

Ex: if we toss a coin the probabilites = heads and tailsthe
probability of getting heads = 50%
The probability of getting tails = 50%
The occuring probability = 100%

We want to know that having alley when the text says
P(Y) = 1% * 80% + 99% +10%
= 10.7%

P(A/Y) = P(A) P(Y/A) / P(Y)
= 1% + 80% / 10.7%
= 7%

LSTM - LONG SHORT TERM MEMORY:

- LSTM network is a type of recurrent neural network architecture that is designed by the problem of vanishing expanding gradients in traditional RNN.

- LSTM are widely used in deeplearning for sequential data analyzing such asspeech recoginization, NLP, & time series analysis

- The architecture of lSTM network is similar to that trend RNN, but it includesmemory cell & three gates

- Input gates
- Forget gates
- Output gates

Restricted boltzman machine:

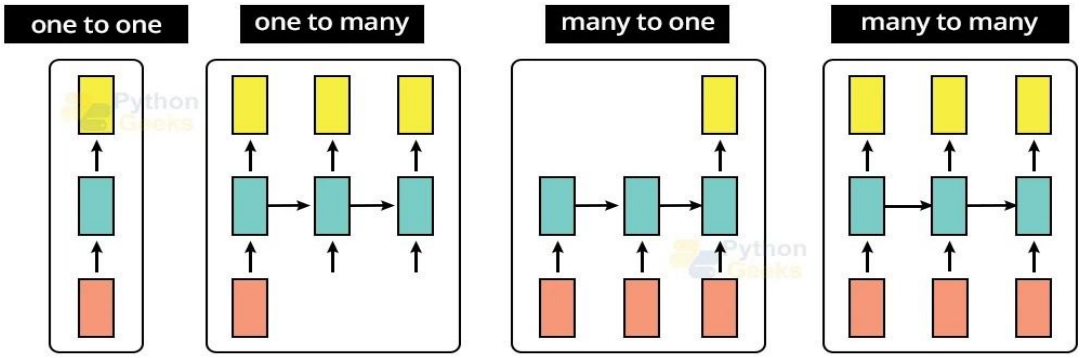
- A RBM is a type of artificial neural networks that commonly used in unsupervised machine learning tasks, such as dimensionally reduction, featurelearning, and collebrative learning to represent the probability distribution
- The RBM consist of 2 layers visible layer and hidden layer
- The visible layers represents the input data and the hidden layers represents a set of latent variables that captures the under lying because they dont have common connections

RNN-- recurrent neural networks

- In RNN we have seprate and indipendent input and output layers
- Which we inefficient the dealing with sequencial data
- Hence a new neural network called rnn was introdued to store of previous outputs in the internal memory
- These results are then fed into the neural network as input this allows it to used in applications like pattern detection speech recoginization, nlp, time series predection
- RNN has hidden layers that acts memory locations to store the output of a layer in a loop

There are 4types are there in recurrent neural network

- One to one
- One to many
- Many to one
- Many to many



1. one to one:

- In rnn is one to one which allows a single input & single output
- It has fired input and output sizes and acts as a traditional neural networks

applications:

Image classification

2. One to many:

- One to many is a type of RNN that gives multiple outputs which we given singleinput.
- It table a fixed size and give a sequential of data inputs and the main applications are found in music generation and image capturing

3. Many to one:

- Many-to-one is used when a single output is required from multiple inputs insequence
- It takes a sequence of inputs to display fixed output

4. Many to Many

- It is used to generate the sequence of output data from a sequence of inputdata

Auto encoders:

- An auto encoder AI a type of neural network architecture that is used inunsupervised learning
- The main goal of auto encoder is to learn a component representation of theoriginal data

Auto encoders consists of two parts
- Encode
- Decode

Types of auto encoders:

- Vanila autoencoders
- Convolutional auto encoders
- Recurrent auto encoders
- Variational auto encoders
- Denoising auto encoders
- Adversial auto encoders

Google net architecture:

Google net:
It is used in deep learning model which is developed by researchers of google and it consist of 22-layers and trained on the image net dataset. It can classifyobjects into 1000 different categories

SOURCE CODE

jupyter STOCK MARKET TEAM 5 DS Last Checkpoint: 12 minutes ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)

Run Code

In [1]:
```
import numpy as np
import pandas as pd
import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
```

In [2]:
```
df_stock = pd.read_csv('C:/Users/user/Desktop/1 DSST Datasets/stockmarket.csv')
df_stock.head()
```

Out[2]:

	date	open	high	low	close	adjclose	volume	ticker	RSIadjclose15	RSIvolume15	...	high-15	K-15	D-15	stochastic-k-15	stochastic-d-15	st
0	2022-01-03	17.799999	18.219000	17.500000	17.760000	17.760000	106600	ASLE	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	
1	2022-01-04	17.700001	18.309999	17.620001	17.660000	17.660000	128700	ASLE	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	
2	2022-01-05	17.580000	17.799999	16.910000	16.950001	16.950001	103100	ASLE	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	
3	2022-01-06	16.650000	16.879999	16.139999	16.170000	16.170000	173600	ASLE	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	
4	2022-01-07	16.219999	16.290001	15.630000	15.710000	15.710000	137800	ASLE	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	

5 rows × 1285 columns

jupyter STOCK MARKET TEAM 5 DS Last Checkpoint: 13 minutes ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)

Run Code

In [3]:
```
df_stock.shape
```

Out[3]: (7781, 1285)

In [4]:
```
df_stock.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7781 entries, 0 to 7780
Columns: 1285 entries, date to TARGET
dtypes: float64(1280), int64(3), object(2)
memory usage: 76.3+ MB
```

In [5]:
```
df_stock.describe()
```

Out[5]:

	open	high	low	close	adjclose	volume	RSIadjclose15	RSIvolume15	RSIadjclose25	RSIvolume25	...	high-
count	7781.000000	7781.000000	7781.000000	7781.000000	7781.000000	7.781000e+03	7316.000000	7316.000000	7006.000000	7006.000000	...	7347.0000
mean	34.990220	35.655999	34.301243	34.964414	34.483147	7.586022e+05	46.817434	49.814790	46.966016	49.898659	...	37.9472
std	99.841502	101.451058	98.073945	99.790823	98.603879	3.934491e+06	11.672838	5.002664	8.760961	3.420371	...	107.3402
min	0.410000	0.435000	0.405000	0.408000	0.408000	0.000000e+00	6.837461	35.303213	17.693637	39.520876	...	0.5100
25%	4.050000	4.130000	3.980000	4.030000	3.960000	1.080000e+04	38.946316	47.182234	40.954487	48.266978	...	4.5650
50%	10.080000	10.110000	10.005000	10.080000	10.061000	8.406000e+04	46.259711	48.356834	46.459477	48.961162	...	10.6400
75%	24.350000	24.500000	24.080000	24.250000	22.466007	6.724000e+05	54.061089	50.902284	52.289893	50.527067	...	25.1700
max	795.739990	799.359985	784.960022	797.489990	783.376221	1.615550e+08	96.365095	99.622735	91.023108	97.782293	...	799.3599

8 rows × 1283 columns

jupyter STOCK MARKET TEAM 5 DS Last Checkpoint: 13 minutes ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)

Run Code

In [6]:
```
df_stock.isnull().sum()
```

Out[6]:
```
date                0
open                0
high                0
low                 0
close               0
                   ...
stochastic-kd-15    587
volumenrelativo     215
diff                155
INCREMENTO          155
TARGET                0
Length: 1285, dtype: int64
```

jupyter STOCK MARKET TEAM 5 DS Last Checkpoint: 13 minutes ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)

Run Code

In [7]:
```
data_new = df_stock[['date','open', 'high', 'low', 'close']]
data_new
```

Out[7]:

	date	open	high	low	close
0	2022-01-03	17.799999	18.219000	17.500000	17.760000
1	2022-01-04	17.700001	18.309999	17.620001	17.660000
2	2022-01-05	17.580000	17.799999	16.910000	16.950001
3	2022-01-06	16.650000	16.879999	16.139999	16.170000
4	2022-01-07	16.219999	16.290001	15.630000	15.710000
...	...	...	...	...	...
7776	2022-12-23	23.250000	23.540001	23.250000	23.290001
7777	2022-12-27	23.350000	23.610001	23.250000	23.350000
7778	2022-12-28	23.450001	23.570000	23.219999	23.350000
7779	2022-12-29	23.330000	23.740000	23.330000	23.610001
7780	2022-12-30	23.680000	23.760000	23.610001	23.610001

7781 rows × 5 columns

In [8]:
```
data_new.isnull().sum()
```

Out[8]:
```
date     0
open     0
high     0
low      0
close    0
dtype: int64
```

jupyter STOCK MARKET TEAM 5 DS Last Checkpoint: 13 minutes ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)

Run Code

In [9]:
```
import seaborn as sns
import matplotlib.pyplot as plt
```

jupyter STOCK MARKET TEAM 5 DS Last Checkpoint: 13 minutes ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)

Run Code

In [10]:
```
data_new["tomorrow"]= data_new["close"].shift(-1)
data_new
```

```
C:\Users\user\AppData\Local\Temp\ipykernel_31772\3575239668.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data_new["tomorrow"]= data_new["close"].shift(-1)
```

Out[10]:

	date	open	high	low	close	tomorrow
0	2022-01-03	17.799999	18.219000	17.500000	17.760000	17.660000
1	2022-01-04	17.700001	18.309999	17.620001	17.660000	16.950001
2	2022-01-05	17.580000	17.799999	16.910000	16.950001	16.170000
3	2022-01-06	16.650000	16.879999	16.139999	16.170000	15.710000
4	2022-01-07	16.219999	16.290001	15.630000	15.710000	15.860000
...	...	...	...	...	...	...
7776	2022-12-23	23.250000	23.540001	23.250000	23.290001	23.350000
7777	2022-12-27	23.350000	23.610001	23.250000	23.350000	23.350000
7778	2022-12-28	23.450001	23.570000	23.219999	23.350000	23.610001
7779	2022-12-29	23.330000	23.740000	23.330000	23.610001	23.610001
7780	2022-12-30	23.680000	23.760000	23.610001	23.610001	NaN

7781 rows × 6 columns

jupyter STOCK MARKET TEAM 5 DS Last Checkpoint: 14 minutes ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

Run Code

In [11]:
```
data_new["target"] = data_new["tomorrow"] > data_new["close"].astype(float)
data_new
```

```
C:\Users\user\AppData\Local\Temp\ipykernel_31772\1825596126.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data_new["target"] = data_new["tomorrow"] > data_new["close"].astype(float)
```

Out[11]:

	date	open	high	low	close	tomorrow	target
0	2022-01-03	17.799999	18.219000	17.500000	17.760000	17.660000	False
1	2022-01-04	17.700001	18.309999	17.620001	17.660000	16.950001	False
2	2022-01-05	17.580000	17.799999	16.910000	16.950001	16.170000	False
3	2022-01-06	16.650000	16.879999	16.139999	16.170000	15.710000	False
4	2022-01-07	16.219999	16.290001	15.630000	15.710000	15.860000	True
...	...	...	...	...	...	...	...
7776	2022-12-23	23.250000	23.540001	23.250000	23.290001	23.350000	True
7777	2022-12-27	23.350000	23.610001	23.250000	23.350000	23.350000	False
7778	2022-12-28	23.450001	23.570000	23.219999	23.350000	23.610001	True
7779	2022-12-29	23.330000	23.740000	23.330000	23.610001	23.610001	False
7780	2022-12-30	23.680000	23.760000	23.610001	23.610001	NaN	False

7781 rows × 7 columns

jupyter STOCK MARKET TEAM 5 DS Last Checkpoint: 14 minutes ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)

Run Code

In [12]:
```
from sklearn.ensemble import RandomForestClassifier
```

In [13]:
```
model = RandomForestClassifier(n_estimators=200, min_samples_split= 100, random_state=1)
```

In [14]:
```
train = data_new.iloc[:-100]
test = data_new.iloc[-100:]
```

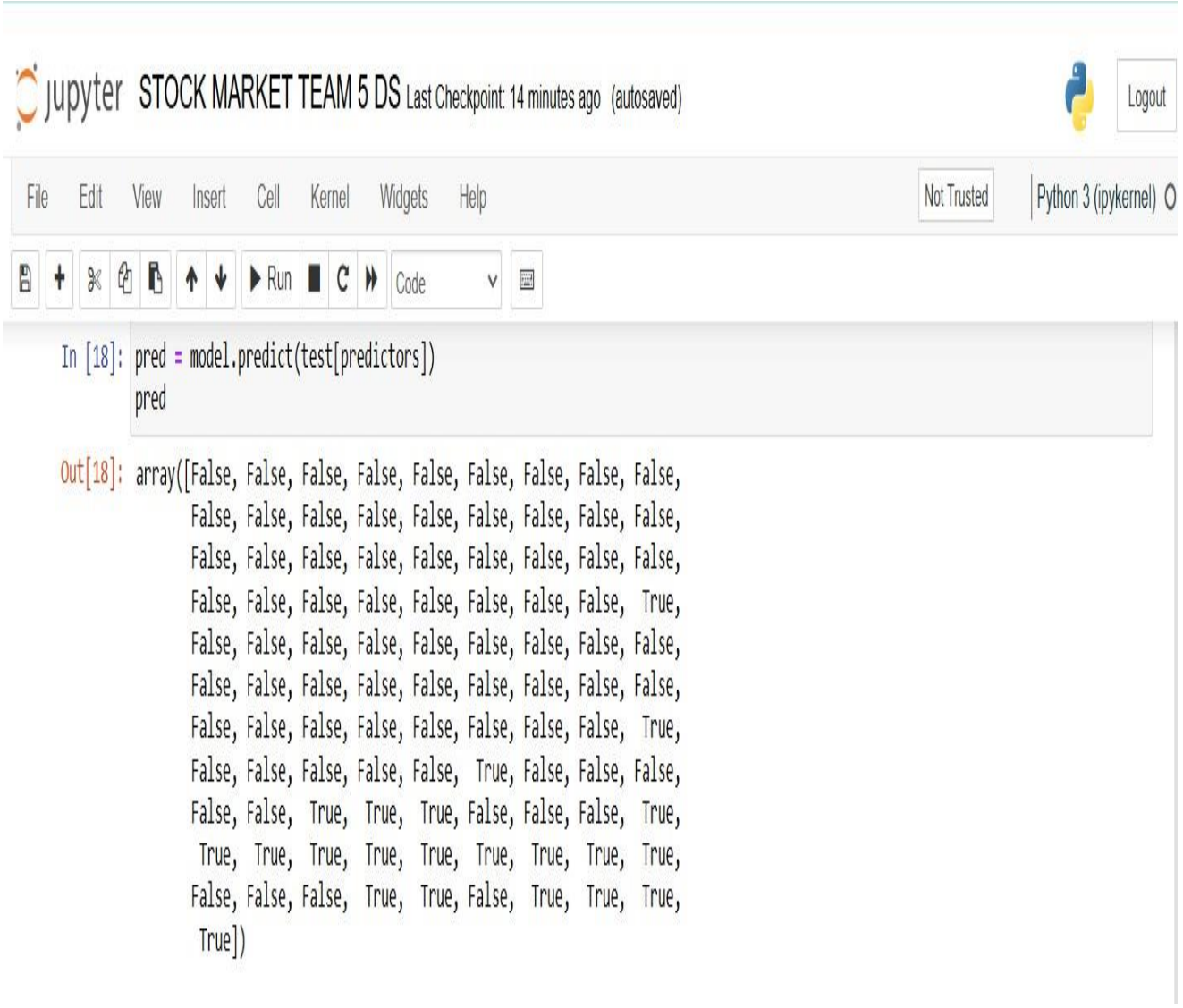
In [15]:
```
predictors = ["open", "high", "low", "close"]
model.fit(train[predictors], train["target"])
```

Out[15]:
```
RandomForestClassifier(min_samples_split=100, n_estimators=200, random_state=1)
```

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.

On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

In [16]:
```
from sklearn.metrics import precision_score
```


jupyter STOCK MARKET TEAM 5 DS Last Checkpoint: 14 minutes ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Code
In [18]: pred = model.predict(test[predictors])
pred
Out[18]: array([False, False, False, False, False, False, False, False, False,
False, False, False, False, False, False, False, False, False,
False, False, False, False, False, False, False, False, False,
False, False, False, False, False, False, False, False, True,
False, False, False, False, False, False, False, False, False,
False, False, False, False, False, False, False, False, False,
False, False, False, False, False, False, False, False, True,
False, False, False, False, False, True, False, False, False,
False, False, True, True, True, False, False, False, True,
True, True, True, True, True, True, True, True, True,
False, False, False, True, True, False, True, True, True,
True])

CONCLUSION

In conclusion, Machine Learning and Artificial Intelligence techniques implemented through Python offer a powerful and groundbreaking approach to **Stock Market Prediction**. Stock market prediction using AI and ML has yielded valuable insights into the feasibility and limitations of employing machine learning algorithms in financial forecasting. We found that while certain models demonstrated promising accuracy in predicting stock price movements, especially in stable market conditions, their performance varied significantly during periods of volatility or when influenced by unexpected external factors. The quality and relevance of data proved crucial, with features like historical prices, trading volumes, and macroeconomic indicators playing pivotal roles in model effectiveness.

Moving forward, further enhancements could focus on integrating more diverse data sources, refining model architectures, and exploring advanced AI techniques to better capture market complexities. Despite the challenges, our findings underscore the potential of AI-driven approaches in aiding investment decisions, albeit with careful consideration of ethical implications and the dynamic nature of financial market.