410255: LP-V. DL 4 22-23 (2nd Sem)					
Student Name :					
BE Computer Division		Roll No		Batch	
Experiment No: 4	Recurrent neural network (RNN) - Design a time series analysis and prediction system using RNN.				

Aim : Recurrent neural network (RNN) - Use the Google stock prices dataset and design a time series analysis and prediction system using RNN.

Theory

Stock price analysis has been a critical area of research and is one of the top applications of machine learning. This tutorial will teach you how to perform stock price prediction using machine learning and deep learning techniques. Here, you will use an LSTM network to train your model with Google stocks data.

what is the Stock Market?

A stock market is a public market where you can buy and sell shares for publicly listed companies. The stocks, also known as equities, represent ownership in the company. The stock exchange is the mediator that allows the buying and selling of shares.

Importance of Stock Market

- Stock markets help companies to raise capital.
- It helps generate personal wealth.
- Stock markets serve as an indicator of the state of the economy.
- It is a widely used source for people to invest money in companies with high growth potential.

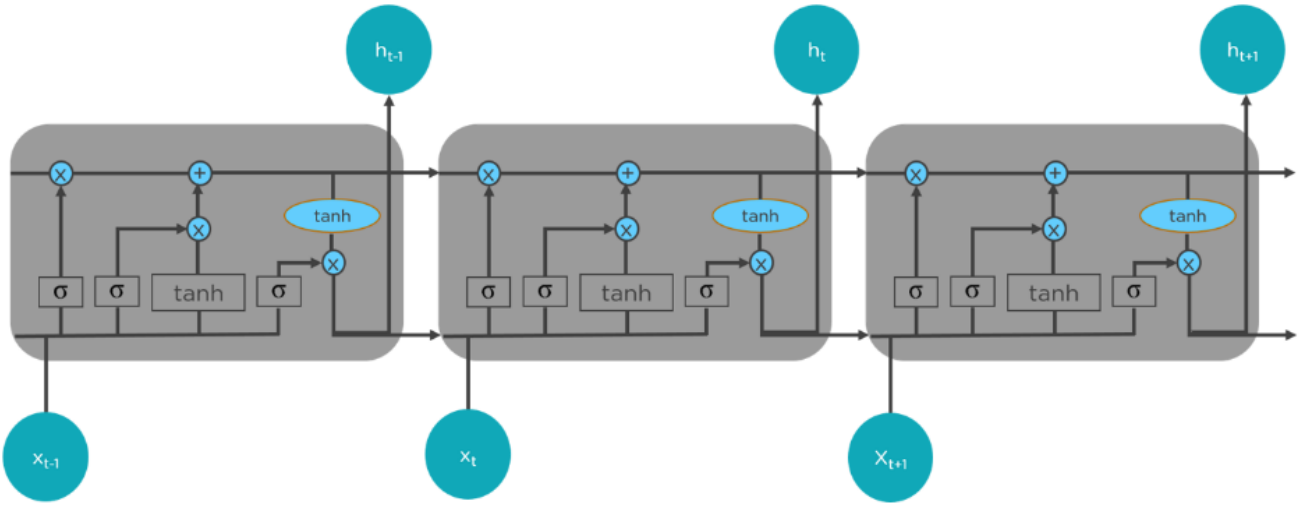
Stock Price Prediction

Stock Price Prediction using machine learning helps you discover the future value of company stock and other financial assets traded on an exchange. The entire idea of predicting stock prices is to gain significant profits. Predicting how the stock market will perform is a hard task to do. There are other factors involved in the prediction, such as physical and psychological factors, rational and irrational behavior, and so on. All these factors combine to make share prices dynamic and volatile. This makes it very difficult to predict stock prices with high accuracy.

Understanding Long Short Term Memory Network

Here, you will use a Long Short Term Memory Network (LSTM) for building your model to predict the stock prices of Google.

LTSMs are a type of Recurrent Neural Network for learning long-term dependencies. It is commonly used for processing and predicting time-series data.



From the image on the top, you can see LSTMs have a chain-like structure. General RNNs have a single neural network layer. LSTMs, on the other hand, have four interacting layers communicating extraordinarily.

LSTMs work in a three-step process.

- The first step in LSTM is to decide which information to be omitted from the cell in that particular time step. It is decided with the help of a sigmoid function. It looks at the previous state (ht-1) and the current input xt and computes the function.
- There are two functions in the second layer. The first is the sigmoid function, and the second is the tanh function. The sigmoid function decides which values to let through (0 or 1). The tanh function gives the weightage to the values passed, deciding their level of importance from -1 to 1.
- The third step is to decide what will be the final output. First, you need to run a sigmoid layer which determines what parts of the cell state make it to the output. Then, you must put the cell state through the tanh function to push the values between -1 and 1 and multiply it by the output of the sigmoid gate.

With this basic understanding of LSTM, you can dive into the hands-on demonstration part of this tutorial regarding stock price prediction using machine learning.

Google Stock Price Prediction Using LSTM

1. Import the Libraries.

```
#Import libraries
import os
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
```

2. Load the Training Dataset.

The Google training data has information from 3 Jan 2012 to 30 Dec 2016. There are five columns. The Open column tells the price at which a stock started trading when the market opened on a particular day. The Close column refers to the price of an individual stock when the stock exchange closed the market for the day. The High column depicts the highest price at which a stock traded during a period. The Low column tells the lowest price of the period. Volume is the total amount of trading activity during a period of time.

```
dataset_train = pd.read_csv("Google_Stock_Price_Train.csv")
dataset_train.head()
```

	Date	Open	High	Low	Close	Volume
0	1/3/2012	325.25	332.83	324.97	663.59	7,380,500
1	1/4/2012	331.27	333.87	329.08	666.45	5,749,400
2	1/5/2012	329.83	330.75	326.89	657.21	6,590,300
3	1/6/2012	328.34	328.77	323.68	648.24	5,405,900
4	1/9/2012	322.04	322.29	309.46	620.76	11,688,800

3. Use the Open Stock Price Column to Train Your Model.

```
training_set = dataset_train.iloc[:,1:2].values

print(training_set)
print(training_set.shape)
```

```
[[325.25]
 [331.27]
 [329.83]
 ...
 [793.7 ]
 [783.33]
 [782.75]]
(1258, 1)
```

4. Normalizing the Dataset.

```
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler(feature_range = (0,1))
scaled_training_set = scaler.fit_transform(training_set)

scaled_training_set
```

```
array([[0.08581368],
       [0.09701243],
       [0.09433366],
       ...,
       [0.95725128],
       [0.93796041],
       [0.93688146]])
```

5. Creating X_train and y_train Data Structures.

```
X_train = []
y_train = []
for i in range(60,1258):
    X_train.append(scaled_training_set[i-60:i, 0])
    y_train.append(scaled_training_set[i, 0])
X_train = np.array(X_train)
y_train = np.array(y_train)
```

```
print(X_train.shape)
print(y_train.shape)
```

```
(1198, 60)
(1198,)
```

6. Reshape the Data.

```
X_train = np.reshape(X_train,(X_train.shape[0], X_train.shape[1], 1))

X_train.shape
```

```
(1198, 60, 1)
```

7. Building the Model by Importing the Crucial Libraries and Adding Different Layers to LSTM.

```
from keras.models import Sequential
from keras.layers import LSTM
from keras.layers import Dense
from keras.layers import Dropout
```

```
regressor = Sequential()

regressor.add(LSTM(units = 50, return_sequences= True, input_shape = (X_train.shape[1], 1)))
regressor.add(Dropout(0.2))

regressor.add(LSTM(units = 50, return_sequences= True))
regressor.add(Dropout(0.2))

regressor.add(LSTM(units = 50, return_sequences= True))
regressor.add(Dropout(0.2))

regressor.add(LSTM(units = 50))
regressor.add(Dropout(0.2))

regressor.add(Dense(units=1))
```

8. Fitting the Model.

```
regressor.compile(optimizer = 'adam', loss = 'mean_squared_error')
regressor.fit(X_train, y_train, epochs=100, batch_size=32)
```

```
Epoch 1/100
38/38 [==============================] - 11s 114ms/step - loss: 0.1011
Epoch 2/100
38/38 [==============================] - 4s 117ms/step - loss: 0.0061
Epoch 3/100
38/38 [==============================] - 4s 118ms/step - loss: 0.0063
Epoch 4/100
```

9. Extracting the Actual Stock Prices of Jan-2017.

```
dataset_test = pd.read_csv("Google_Stock_Price_Test.csv")
actual_stock_price = dataset_test.iloc[:,1:2].values
```

10. Preparing the Input for the Model.

```
dataset_total = pd.concat((dataset_train['Open'], dataset_test['Open']), axis = 0)
inputs = dataset_total[len(dataset_total)- len(dataset_test)-60:].values

inputs = inputs.reshape(-1,1)
inputs = scaler.transform(inputs)

X_test = []
for i in range(60,80):
    X_test.append(inputs[i-60:i, 0])
X_test = np.array(X_test)
X_test = np.reshape(X_test,(X_test.shape[0], X_test.shape[1], 1))
```

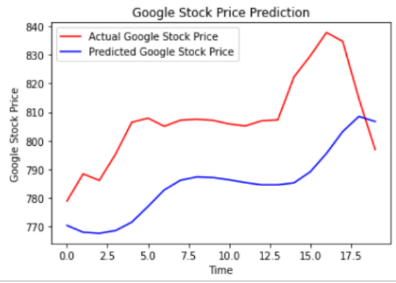
11. Predicting the Values for Jan 2017 Stock Prices.

```
predicted_stock_price = regressor.predict(X_test)
predicted_stock_price = scaler.inverse_transform(predicted_stock_price)
```

12. Plotting the Actual and Predicted Prices for Google Stocks.

```
plt.plot(actual_stock_price, color = 'red', label = 'Actual Google Stock Price')
plt.plot(predicted_stock_price, color = 'blue', label = 'Predicted Google Stock Price')
plt.title('Google Stock Price Prediction')
plt.xlabel('Time')
plt.ylabel('Google Stock Price')
plt.legend()
```

```
<matplotlib.legend.Legend at 0x7fbf71eb1b90>
```



As you can see above, the model can predict the trend of the actual stock prices very closely. The accuracy of the model can be enhanced by training with more data and increasing the LSTM layers.

Conclusion

The stock market plays a remarkable role in our daily lives. It is a significant factor in a country's GDP growth. In this tutorial, you learned the basics of the stock market and how to perform stock price prediction using machine learning.

Questions

1. Discuss RNN i.e. LSTM
2. Dicuss time series analysis and prediction system
3. Discuss time series analysis for forecasting system

Attendance (5 Marks)	Conduction (5 Marks)	Oral (5 Marks)	Total (15 Marks)	Sign of Teacher