AID 552
PROJECT REPORT
Algorithmic Trading using LSTM (Group-2)

Basic Idea:

In this project, we basically try to influence buying and selling of stocks using an algorithmic model built using an ensemble of KNN, Decision Tree, Random Forest, SVM, and LSTM. The model depicts an ideal scenario for maximizing profits from a trade. Based on a set of rules using various technical indicators and the deep learning model, the trading signal (Buy or Sell) is predicted.

Dataset:

The dataset compromises 10-year Stock details of 5 stocks namely:

RELIANCE, HDFC, ITC, INFOSYS, and TCS. The result of the analysis is the predicted trend of the market index, which can be used to set out some trading rules:

1. If the next day's trend is Uptrend, then the decision is BUY
2. If the BUY decision already exists, then HOLD.
3. If the next day's trend is Downtrend, then the decision is SELL
4. If a SELL decision already exists, then HOLD

According to the result obtained with these rules, the return of strategy will be calculated.

Feature Extraction and Selection:

Feature selection is the process of selecting a subset of features that are most relevant for model construction which aids in creating an accurate predictive model. There is a wide range of feature selection algorithms, and these mainly fall into one of the three categories:

Filter method– selects features by assigning a score to them using some statistical measure. Wrapper method– evaluates a different subset of features, and determines the best subset. Embedded method – This method figures out which of the features give the best accuracy while the model is being trained.

A few ML models like KNN, SVM, Decision Tree Classifier and Random Forest have been compared so as to better extract the important features from our dataset. The Random Forest classifier is proved to be more accurate in depicting the underlying pattern in our data.

Thus, we used the filter method utilizing the **random.forest.importance** function. The "random.forest.importance" function rates the importance of each feature in the classification of the outcome, i.e. class variable. The function returns a data frame containing the name of each attribute and the importance value based on the mean decrease in accuracy. The top five features(indicators) according to the RF classifier are as follows -

Rate Of Change (ROC) Indicator:

- The Price Rate of Change (ROC) oscillator is an unbounded momentum indicator used in the technical analysis set against a zero-level midpoint.
- A rising ROC above zero typically confirms an uptrend while a falling ROC below zero indicates a downtrend.
- When the price is consolidating, the ROC will hover near zero.

$$\mathrm{ROC} = \left(\frac{\mathrm{Closing\ Price}_p - \mathrm{Closing\ Price}_{p-n}}{\mathrm{Closing\ Price}_{p-n}} \right) \times 100$$

Relative Strength Index (RSI):

The Relative Strength Index (RSI) is a measurement used by traders to assess the price momentum of a stock or other security. The basic idea behind the RSI is to measure how quickly traders are bidding the price of the security up or down. The RSI plots this result on a scale of 0 to 100. Readings below 30 generally indicate that the stock is oversold, while readings above 70 indicate that it is overbought.

α

$$RS = \frac{Avg.Gain}{Avg.Loss}$$

$$RSI = 100 - \frac{100}{1 + RS}$$

Momentum (MOM):

The Momentum (MOM) indicator compares the current price with the previous price from a selected number of periods ago. This indicator is similar to the "Rate of Change" indicator, but the MOM does not normalize the price, so different instruments can have different indicator values based on their point values.

$$MOM = \text{Price} - \text{Price of n periods ago}$$

Fast Stochastic Oscillator:

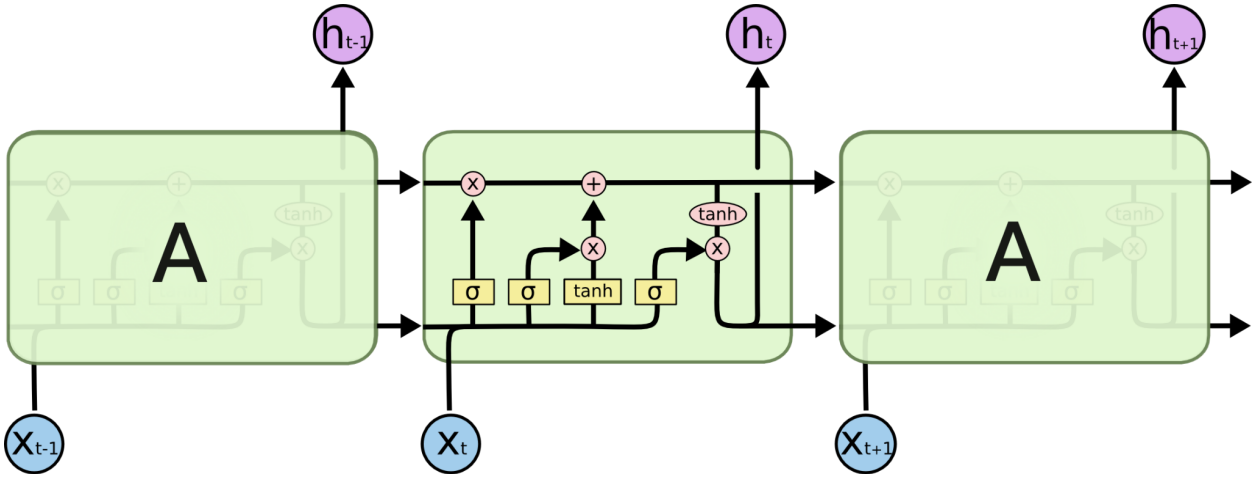
A stochastic indicator (%K) is a momentum technical indicator that aims to measure the trend in prices and identify trend reversals.

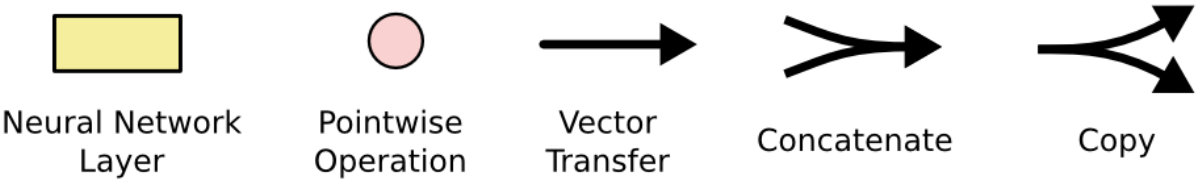
$$\%K = \frac{(C - L)}{(H - L)} * 100$$

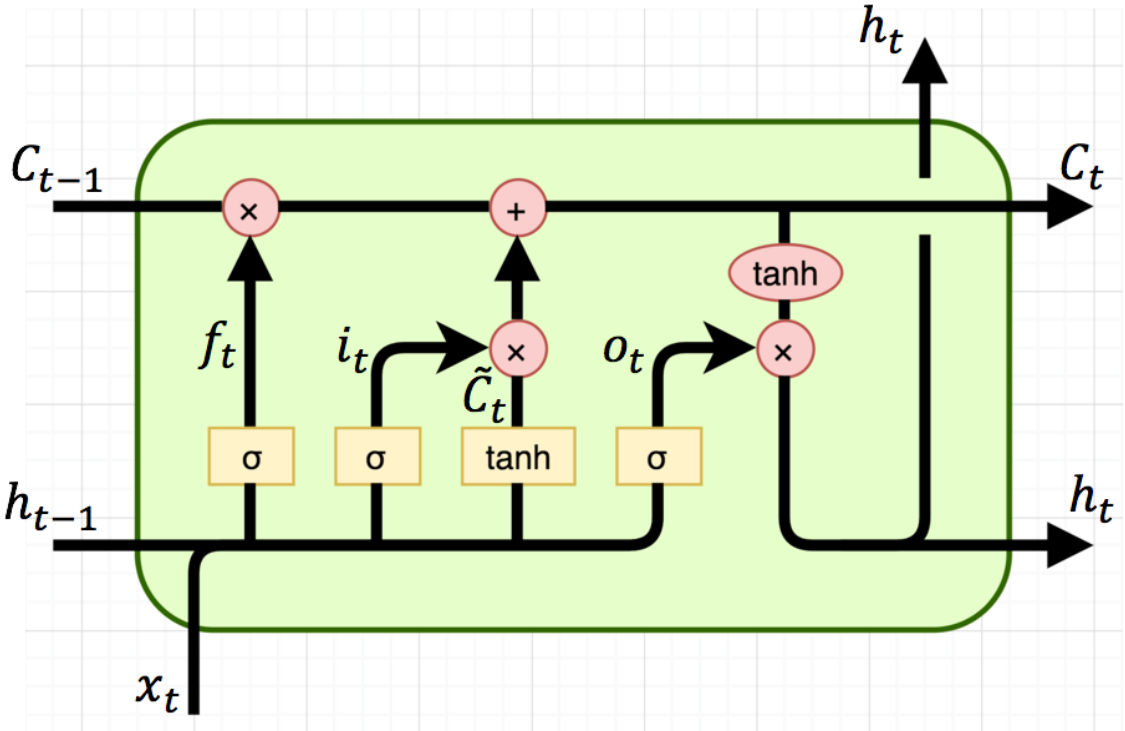
The fast stochastic oscillator (%K) is a momentum indicator, and it is used to identify the strength of trends in price movements. It can be used to generate overbought and oversold signals. Typically, a stock is considered overbought if the %K is above 80 and oversold if %K is below 20.

Long Short-Term Memory (LSTM) Networks:

Long Short-Term Memory networks – usually just called “LSTMs” – are a special kind of RNN, capable of learning long-term dependencies. LSTMs are explicitly designed to avoid the long-term dependency problem. Remembering information for long periods of time is practically their default behavior, not something they struggle to learn.


h_{t-1}
h_t
h_{t+1}
A
A
tanh
σ
σ
tanh
σ
X_{t-1}
X_t
X_{t+1}

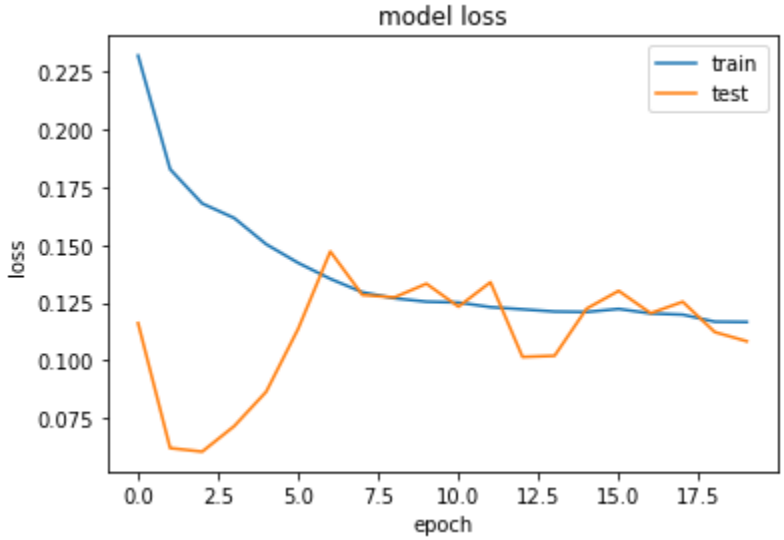

Neural Network Layer
Pointwise Operation
Vector Transfer
Concatenate
Copy


h_t
C_{t-1}
C_t
tanh
f_t
i_t
o_t
$\tilde{C}_t$
σ
σ
tanh
σ
h_{t-1}
h_t
x_t

In the above diagram, each line carries an entire vector, from the output of one node to the inputs of others. The pink circles represent pointwise operations, like vector addition, while the yellow boxes are learned neural network layers. Lines merging denote concatenation, while a line forking denotes its content being copied and the copies going to different locations.

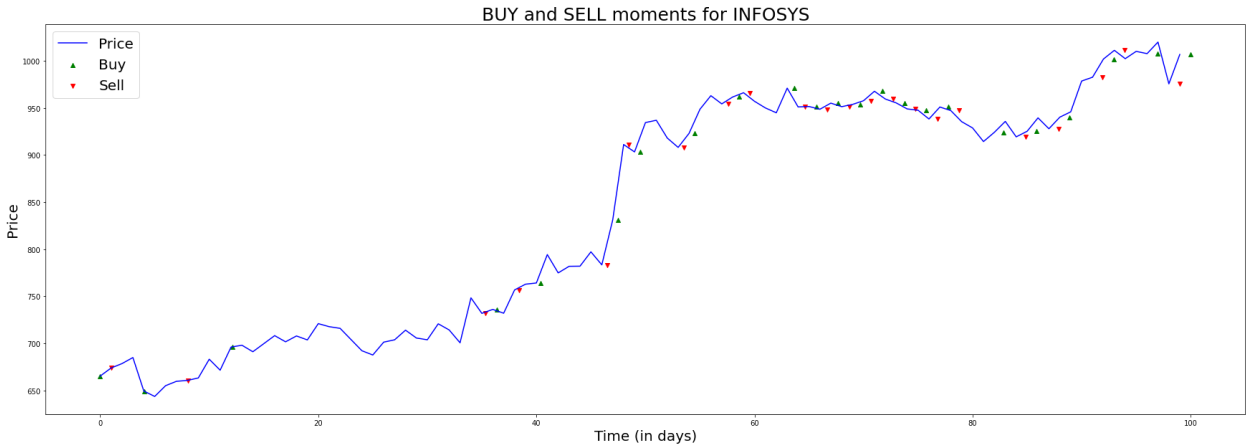
LSTMs use a series of 'gates' that control how the information in a sequence of data comes into, is stored in, and leaves the network. There are three gates in a typical LSTM; forget gate, input gate, and output gate. These gates can be thought of as filters and are each their own neural network.

The input matrix contains the top five technical indicators as features and the output is a vector of signals(buy or sell). The LSTM model is trained using the above-mentioned input and output for 20 epochs.



Results:

Our LSTM model predicts or indicates the buy and sell moments in the coming days. We can see that the model is performing well in a trendy market but lacks accuracy when the market is drifting sideways.


BUY and SELL moments for INFOSYS
Price
Buy
Sell
Price
Time (in days)
650
700
750
800
850
900
950
1000
0
20
40
60
80
100