stockDL: A Hybrid deep learning Library to predict the annual yields from stocks for an inexpensive computational environment.

ABSTRACT: Predicting and analyzing the stock market has been of primary interest to researchers, investors, and market experts. The technology has been evolving continuously from manual to automated collection, tuning, and data analysis to generate insights and predict the rise or fall of a stock. This work presents stockDL, a deep learning solution to analyze, understand the historical stock data and calculate the gross and annual yield for the chosen stock ticker. The proposed solution is comprehensive and user-friendly. It includes data collection and preprocessing and utilizes various mathematical and deep learning techniques for feature extraction combined with state-of-art neural network architectures to predict the market trends. The stockDL algorithm assimilates two traditional stock trading techniques, Buy and Hold strategy and Moving Average ribbon trading strategy, with two Deep Learning Models created using the state-of-art Long Short-Term Memory networks. The first model is a pure LSTM network, whereas the second network is a Mixture of Convolution Neural networks and LSTMs. stockDL uses the data of the past five years from the date of generating the predictions, making the model immune from any sudden fluctuations in the historical data. When evaluated on the four stock symbols (AAPL, GOOGL, HDFCBANK.NSE, RELIANCE.NSE), the model has attained state-of-art for deep learning backed algorithmic trading in a controlled computational environment. The novel solution introduced in this study is faster and more accurate than any existing deep-learning solutions available. It is immune from any sudden dramatic decline among significant sections of the stock market Market Crash). This work contributes to the stock analysis and research community in both the technical and financial domains.

1. INTRODUCTION

The stock market is an area of high profit and high risks, and this is considered while devising and generating a quantitative investment strategy to predict and judge the stock's future price using the historical stock data. The market governed by various financial and non-financial factors poses a new challenge to the researchers to develop a best-fit solution for predicting the stock prices or the annual yield by investing in a particular stock. The three significant factors measuring the outcome of investing in a company or business are Environment, Social, and Governance (ESG). Considering ESGs while making investing decisions leads to a favorable outcome in most cases[1].

The Environment of an investment market involves Assessment and Investment vehicles, Financial Markets, Market Structure, Market Intermediaries, Investment process, Regulation, and Economy. The Indian Financial markets include the Credit Market, Money Market, Foreign Exchange Market, Debt Market, and Capital Market. The Social factors include the interactions between neighbors, friends, colleagues, advice given by analysts, planners, bankers[2]. Social media and Interactions are prominent areas to analyze and predict the market trends based on recent activities of significant investors, executives, and influencers[3]. Security and Exchange Board India, established on April 12, 1992, handles India's stock market governance. Corporate Governance drafts the rules and regulations for the companies and the shareholders and assists in the smooth day-to-day functioning of trade, The Governance also handles illegal trade practices like Insider Trade, Security Frauds, et cetera[4].

This study follows the idea of conquering one step (or problem) at a time. It focuses on providing a solution to predict the fluctuations on a stock based on the Environment (the Historical Data). With the availability of a massive amount of data, the challenge is to use it and draw meaningful conclusions from it. We have analyzed the data from the date of the initial issue to the date of making predictions. The two deep learning algorithms introduced in this study include a Long Short Term Memory (LSTM), a recurrent neural network architecture for time-series prediction [5], and a Convolution Neural Network (CNN) and LSTM mix architecture for making the predictions. The baseline comparison of the deep-learning strategy is with two traditional stock trading strategies, the Buy and Hold system and the Moving Averages.

The deep learning model is trained end-to-end on new data (between a training window of the past six years from the day of making the predictions). This strategy is to prevent any sudden old fluctuations in the data from contaminating the projections. All outliers are scaled using a min-max scaler to stop them from dominating in the results.

Our results suggest a substantial promise in integrating the traditional and computer-generated strategies for developing an improved quantitative investment strategy. The algorithms outperform the baselines set by the conventional approach, and the enhanced LSTM-CNN mix model provides better performance with reduced computational cost. Integration of this model with other deep learning strategies to handle other governing factors like Governance and Social can bring a new revolution in algorithmic trading using Deep Learning.

2. RELATED WORK

Our work connects several relevant pieces of literature. Recent work highlights how Deep Learning can be in algorithmic trading. With researchers working on individual factors affecting the stock prices, as the social factor, Environment, and Governance [3, 4, 6, 7], a new vision is being added to the analysis and projection of stock prices based on the contributing factors. Reinforcement Learning libraries like finRL (Financial Reinforcement Learning) and TA-Lib (Technical Analysis Library in Python) have set a new benchmark in Artificial Intelligence for finance.

The use of machine learning algorithms like Support Vector Machine (SVM) [8] and a hybrid feature selection method provided a detailed parameter adjustment procedure with performance under different parameter values. The performance of this algorithm is significantly less than the state-of-art LSTMs.

Various other LSTM based models on long-term data fail to address the computational complexities and the efforts required to train it [9]. Our study focuses on the training aspect of the model and tries to reduce the training time even in a limited computational environment.

The limitation of these developments is that they fail to scale because of high computational requirements [10]. The creation of new replicas of the trading environment and dummy data generation fail to use the existing data. The current deep learning solutions using LSTMs and LSTM-CNN Mix are tested only on short-term predictions ranging from 1 day to 10 days [11].

The main contribution of our work to this problem is to:

i. Use existing, publically available historical data of stocks to train and test the model.
ii. Development of a computationally cheaper and simple deep-learning solution to provide enhanced performance and improved scalability.
iii. Comparing various investment strategies (Traditional and Deep Learning) makes it easier for the user to create a rational decision.

Our work uses the stock market to focus on a crucial issue, to align the human devised strategies with the computer-generated strategy for algorithmic trading and quantitative finance.

3. DATA AND BACKGROUND

3.1 Background

Our work leverages three radically different approaches. The moving average method is purely mathematical, and the Buy and Hold strategy is positional and confidence-based. In contrast, stockDL focuses on two deep learning architectures, a black-box, and tries to understand the pattern in the historical data and derive meaningful insights from it.

Moving Average Strategy: It follows a set of rules to decide whether to Trade In or Stay Out of the market for the month considered. This strategy focuses on smoothening out the price trends by filtering out the noise from short-term predictions. It acts as a support in case of an uptrend of the time frame taken. This method helps in understanding where Moving Averages will offer support and where it offers resistance. Support indicates a price level where we can expect a downtrend, whereas resistance suggests an increase in the price level, that is, an uptrend. Traditional investors have developed various tools to use the moving average to indicate upcoming trends in the prices.

$$\frac{\sum_{i=m}^{n} Price}{m - n}$$

$$Where\ m \to is\ the\ starting\ date\ (Six\ years\ before\ n)$$
$$And\ n \to is\ the\ final\ date\ (Date\ of\ makeing\ the\ prediction, Current\ date)$$

Buy and Hold Strategy: The investor using this strategy buys stocks and holds them for a long time irrespective of the market fluctuations. This approach is generally long-term and relies on the confidence of the investor. It is a passive investment strategy, and the investor might not sell the possessions at the optimal time.

Deep Learning (LSTM Strategy): LSTM is the backbone of the two architectures defined in this study. LSTM learns to keep the relevant information and forget non-relevant data. RNNs can retain the information at time t about the input seen many timestamps before t; this fails in practical implementation due to the problem of vanishing gradients (the gradient gradually becomes zero because of multiplication of long series of numbers less than zero). LSTM saves the information for later and prevents the older signals from being lost during the processing. The LSTM cells allow the past information to be reinjected later, overcoming the vanishing gradient problem.

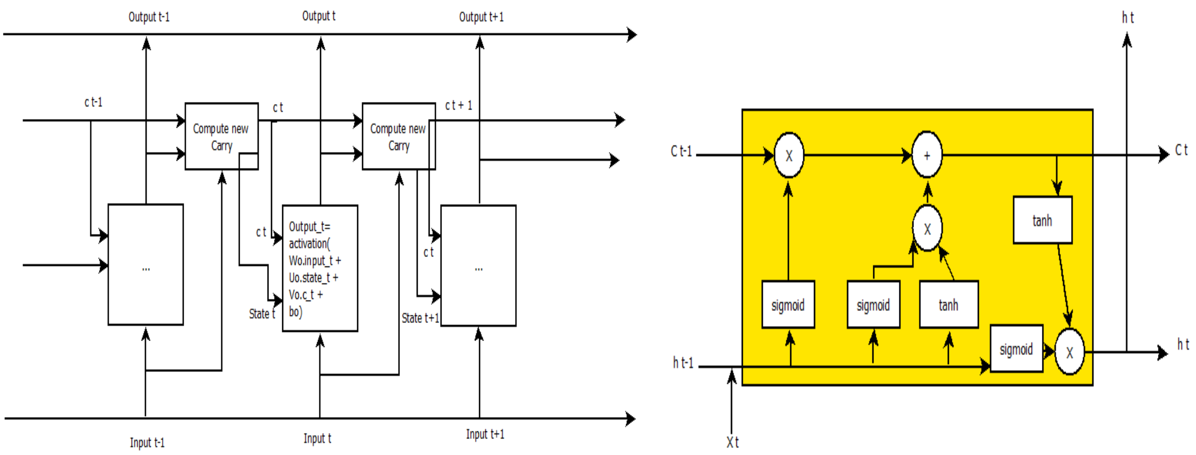

Figure 1: (a) Anatomy of an LSTM with three input cells[12], (b) A single LSTM Cell

3.2 Data

The data module of stockDL collects the data required for the study. The data is retrieved from the Yahoo Finance API based on a unique symbol provided to each stock (this unique symbol is called the stock ticker), the starting date, and the end date.

The data tables considered in the study include information such as opening rate (of the day), the highest rate (of the day), the lowest rate (of the day), the closing rate (of the day), and the volume traded (on that day). As we consider dividends and stock split in calculating the annual yield, the columns comprising it are dropped.

Table 1: First five rows of the historical stock data for further pre-processing of HDFC Bank Limited

Date	Open	High	Low	Close	Volume
1996-01-01	2.458312	2.458312	2.373122	2.417745	350000
1996-01-02	2.417746	2.454255	2.393406	2.413689	412000
1996-01-03	2.413689	2.429916	2.393406	2.421802	284000
1996-01-04	2.421802	2.417746	2.385293	2.405576	282000
1996-01-05	2.405576	2.417746	2.393406	2.401519	189000

Table 2: Stock Symbols (Ticker) of the stocks used in the study

Stock Name	Stock Symbol (Ticker)	Stock Exchange	Currency
Alphabet Inc. (Google)	GOOG	NASDAQ Global Select Market	USD
Apple Inc	AAPL	NASDAQ Global Select Market	USD
HDFC Bank Limited	HDFCBANK.NS	National Stock Exchange	INR
Reliance Industries Limited	RELIANCE.NS	National Stock Exchange	INR

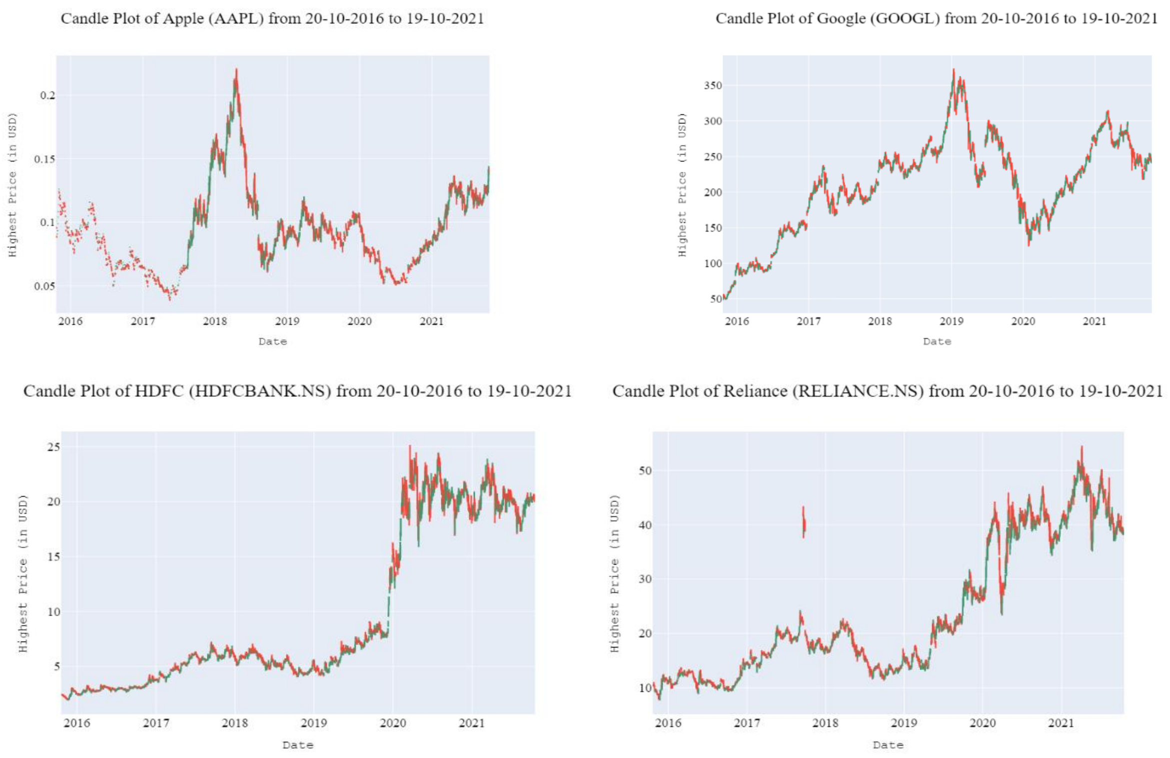

Figure 2: Candle plots for the stocks considered (a) Apple Inc. (AAPLO (b) Google (GOOGL) (c) HDFC Bank Limited (HDFCBANK.NS) (d) Reliance Industries Limited (RELIANCE.NS)

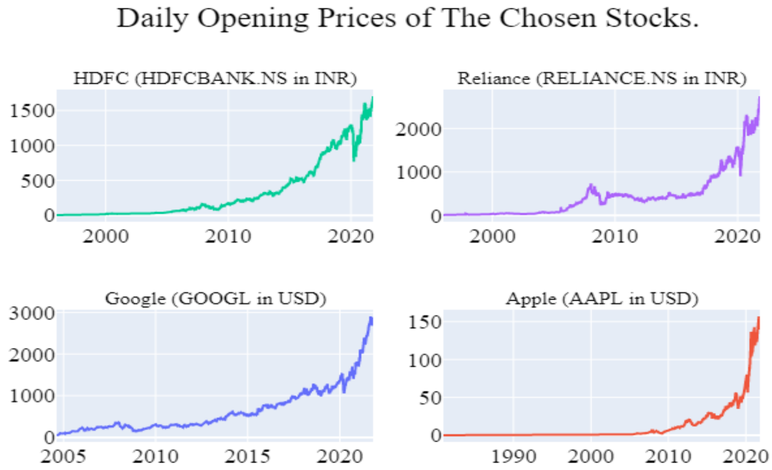

Figure 3: Opening Price of each day of the stocks considered

Apart from storing the data related to the stock, we have stored the first date of each month in the six years considered. These first dates will be used in the data preprocessing module, as the strategy is to decide if we should stay in the market and trade for that particular month or move out with the existing amount.

3.2.1 Data Preprocessing

After dropping the extra columns, we converted the data obtained from the API from daily stock data to monthly stock data. The projection model will use this monthly data to decide if it is profitable to trade in the month by considering the historical data. We have calculated the moving average for 12 months (yearly) and 24 years after generating the monthly data. This module also creates a six-year window of the past six years from the day of making predictions, which will be the final data for the LSTM network after scaling.

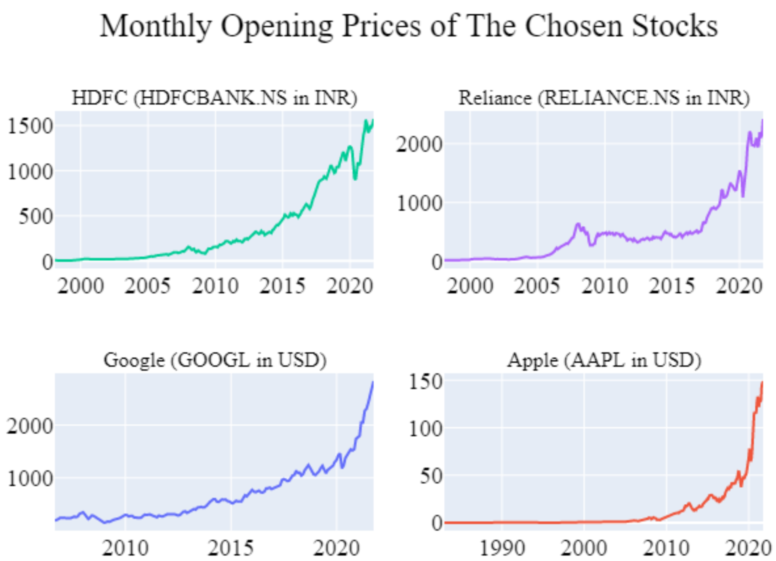

Figure 4: Opening Prices of the considered stocks for the first trading day of each month

We split the monthly data into training and testing data to train and evaluate the model. The split data was normalized separately to prevent any data leak. We normalized the data to values between -1 and 1 using min-max scaling, To ensure that no feature is falsely prioritized based on its value.

Normalization replaces the value in each column with the following formula:

$$m = \frac{x - xmin}{xmax - xmin}$$

$Where, m \ = \ new\ cell\ value,\ x \ = \ initial\ cell\ value,$

$xmax \ = \ maximum\ column\ value,$

$xmin \ = \ minimum\ column\ value$

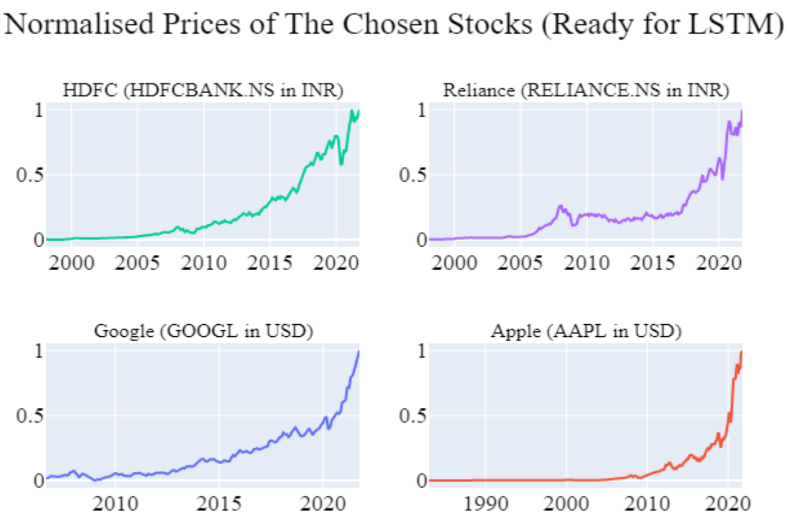

Figure 4: Normalised Monthly Opening Prices

3. stockDL Algorithm and Model Architecture: