APPLICATION OF TRADITIONAL MACHINE LEARNING ALGORITHMS AND DEEP LEARNING TECHNIQUES IN STOCK FORECASTING MODEL

Cong Minh Do **Hoang Son Nguyen** **Hong Anh Duong**

ABSTRACT

The art of predicting stock market behavior has always been a great challenge for both academics and investors, yet the necessity for such act belies in the inevitable need for risk management for financial market operators and gaining abnormal return for stock brokers. Despite the equity market chaotic and complex nature which stems from a large number of factors, this challenge has been tackled by multiple authors across disciplines, ranging from economics, statistics, computer science and even physics. The introduction of Machine Learning, especially Deep Learning to the financial market landscape has resulted in numerous experiments being conducted to predict future trends of stock prices with varying degree of success. In this paper, we present an approach to predict prices of 3 stocks from different sectors and markets as well as trend forecasting using traditional Machine Learning algorithms such as Linear Regression, Polynomial Regression, K Nearest Neighbor and explore the applicability of Deep Learning techniques in time series forecasting using two forms of special type of recurrent neural network: Long Short-Term Memory (LSTM) and Conv1D LSTM.

INTRODUCTION

Stock price forecasting system serves as a foundational basis for sound investment strategies and secure risk management model, yet the feasibility of the aforementioned task has been under suspicion of the famous Efficient Market Hypothesis (EMH). According to Malkiel and Fama (1970), under the EMH assumptions, all information relevant to investment decision shall be available to all market participants and immediately reflected in the price of stock. From a theoretical perspective, EMH claims it is impossible to achieve above-market returns adjusted to the level of risk assumed consistently and over the long term. Nonetheless, the popularity of computationally expensive system with heavy reliance on machine learning algorithm is on the rise, evidenced by the percentage of algorithmic trading on US equity markets reaching approximately 85% on a typical trading day (M.Giantz and R.Kissell,2013). A possible explanation for such phenomenon might be the predictive power of machine learning algorithm, suggesting the underlying assumptions behind EMH may fail to hold in practice. Motivated by this context, we will conduct an experiment using a variety of approaches based on machine learning and deep learning algorithms to tackle this goal, then present a comparison in terms of accuracy and goodness of fit score in stock movement predictions as relevant metrics for each individual approach.

The algorithm of choice are as follows: Regression family (Linear and Polynomial Regression), K Nearest Neighbor and Recurrent Neural Network, in particular Long Short-Term Memory and a modified version with convolutional layers stacked on top. The first three models will act as baseline models against which the efficiency of LSTM networks will be assessed. Our specific concern for this architecture results from the novelty in design of LSTM. The replacement of artificial neuron cell by memory cell in the hidden layer of recurrent network gives LSTM the unprecedented capability of distinguishing between early and recent information by assigning different weights for each while the forget gate facilitates eradicating memory deemed as unnecessary for decision-making process. Given LSTM network success in handling sequential data and the extensive application in the field of natural language processing, we are curious as to its performance to forecast stock trend movements.

The paper we present modelled and predicted the closing stock prices and movement prediction for 3 stocks : Alibaba, VinGroup and Pepsico. Most stock in our dataset comprise of 10 year historical daily prices and transactional information. Each time step is characterized by a set of technical indicators: opening price, closing price, high, low and volume. Upon this dataset the model will be trained, validated and attempt to predict future price of 3 stocks and whether there will be an increase in closing price the next day. A more comprehensive description will be provided in Section III

The method employed in this article is technical analysis, which is a method based solely on historic transactional data from the market introduced by Charles Dow. The assumptions for technical analysis are presented as follows
(1) prices are defined exclusively by the supply-demand relation;
(2) prices change following tendencies;
(3) changes on supply and demand cause tendencies to reverse;
(4) changes on supply and demand can be identified on charts;
And (5) patterns on charts tend to repeat
(David et.al,2017)
In other words, under these assumptions, socio-political, social sentiments or other macro-economical factors are not taken into consideration.

The aim of this study is to provide an overview of a subset of the most popular machine learning and deep learning techniques performance with regards to time-series analysis, in particular predicting stock price movements. Result obtained from this article is expected to fill in the gap of current existing literature by conducting the assessment of different models accuracy in predicting stock prices movement with a special regard of Vietnam equity market. Further use of our suggestions may include the development of more profitable automated trading strategies for investors, more accurate prediction for risk managers and deeper understanding of most commonly used time series modelling. The remainder of the article will be structured as follows: section II will briefly exemplify the relevant literature both in the context of conventional financial market and the prevalence of newly-introduced computer science algorithm with financial market practitioners. Section III will give a detailed description of the approach proposed, including a comprehensive depiction of experimental framework, including feature engineering and data preprocessing and hyperparameters tuning process, section IV will present the results of our experiments while section V will conclude the article.

BACKGROUND AND RELATED WORK

The predictability of stock market has always been the subject of passionate argument within the world of academia. EMH hypothesis proposed by Malkiel and Fama(1970), arising from the empirical observation of the great similarity between changes in price and random walk process, states that even when exploitable return patterns come up, the profit generated by short term transactions will be diminished due to transaction costs and commision. In other words, a change in stock price is independent of all previous historic transactional information, and the future price of an asset only reflect the information available to the market in the future. In addition, a series of experiments conducted by Biondo et.al (2013) has disproved the predictive capacity of conventional technical trading methods such as Moving Average Convergence Divergence and Relative Strength Index.

On the other hand, there have been numerous attempts to show that stock market prices are, to some extent, predictable. Even the author of EMH, Malkiel (2003) suggested the assumption under which EMH holds, all participants are rational might be inconsistent with reality, and the development of a profitable predictive algorithm might be possible. The first category of relevant work is econometric models, which consists of a range of classical forecasting models like autoregressive method (AR), moving average model (MA), autoregressive integrated moving average (ARIMA). The main idea encompassing these methods involves the evolving variable of interest being regressed on its own prior values and an independent noise terms. The drawbacks of these models are its strong reliance on a

relatively strict set assumptions: the error term must be identically and independently distributed i.e each error term must have the same probability distribution and mutually independent etc… The second category of forecasting algorithm is machine learning models, which have been demonstrated to generate more accurate predictions with regards to financial market than conventional econometric approaches (Hsu et al., 2016). He also proved that the forecasting effect of the financial market was affected by the maturity of the market, the input variable, the base forecasting time and the forecasting method.

The integration of computer science methodology and into finance landscape in recent years combined with advances in computational power have established a new body of academic research to study the applicability of more computationally intensive machine learning algorithm. Most commonly used methods of shallow machine learning in stock forecasting model nowadays are Regression, K-Nearest Neighbor, Support Vector Machine/Regression or models that combine them with other algorithms. One of the most notable work in time series classification is Ballings et al.,2015, which involves financial data collection of over 5000 publicly listed European companies and benchmarking ensemble methods (Random Forest, AdaBoost and Kernel Factory) against single classifier models (Neural Networks, Logistic Regression, Support Vector Machines and K-Nearest Neighbor). Using area under the receiver operating characteristic curve (AUC - ROC) as the performance metrics, the article suggested Support Vector Machines demonstrate highest predictive power when being applied to technical analysis indicators. In the same vein, Mitesh et al.,2015 proposed using Factor analysis as dimensionality reduction techniques to reduce from 50 different factors to only 4 most important factors before applying multiple linear regression to predict Indian Stock Exchange index (NIFTY). The R squared - goodness of fit score for the model above reached 90%.

With regards to deep learning domain, various artificial intelligence architectures that imitate biological process i.e artificial neural network or long short-term memory network have been deployed to forecast the future direction of equity market. Advances in the field of natural language processing has motivated the incorporation of sentiment analysis over social media, online presses into stock forecasting. The most common method involves taking news text data as input to foresee price movement. Nonetheless, the degree of success was generally unsatisfactory to obstacles in handling available data sources (Huynh et al.,2017) For instance, language sequences can be comprised of very large vocabulary and difficult to grasp the long term context and dependencies between sequences, thus might render the categorization of an article to predefined categories (negative, neutral and positive) useless. Additionally, even though attempts have been made to study the chaotic and complicated interactions between social sentiments and stock movement, theoretical justification remains elusive and results generally subject to overfitting and poor performance. Within the scope of this article, we will focus solely on presenting relevant neural network architectures that depends on technical analysis indicator.

The first article worth mentioning was Stock market's price movement prediction with LSTM neural networks by David et al.,2017 presented in International Joint Conference Neural Networks. By deploying a classification model based on LSTM network and gathering historic price patterns of 5 stocks, the aim of their study was to predict whether the price of each individual stock will increase in the next 15 minutes. On top of past price data, the model was also trained on additional 175 technical indicators generated by TA-Lib library to represent a more diverse set of features, consisting of relative strength index, intensity of movement tendency, visual graphical patterns etc… His findings suggested LSTM-based classifier outperform Multi-Layer Perceptron and Random Forest, getting up to an average of 55.9% of directional accuracy. Similarly, in addition to conventional time series of candles features (open, close, high, low and volume), Xu et al.,2019 also combined Stock Technical Index (moving average, psychological line,…) and stock macro index (price earning ratio, price to book ratio, price cash flow ratio) as input for LSTM network for trend prediction of multiple stocks on Chinese equity

market and achieved up to over 65% accuracy for some particular stocks. Jain et al.,2018 suggests using Conv1D LSTM instead of CNN or LSTM separately since empirical evidence obtained by his experiment demonstrate much lower error rate (MAPE at 1.98%) generated by Conv1D LSTM, but directional accuracy was not taken into account in his experiment. It is worth mentioning that despite multiple goodness of fit score being used in their papers, only MAPE is scale independent, therefore the only comparable benchmark against which other models can be assessed. To the best of our knowledge, current state of the art model in stock movement classification reached an accuracy of 76% by using Attention-based Multi-input LSTM (Sangyeon et al.,2019)

Problem Formulation

The majority of existing literature only focus on one out of two main metrics in assessing stock price predictive capability of models: Accuracy and MAPE. The training process also differs in the sense while the first one will generally focus more on time series classification i.e: using a class label for each timestep, using a categorical loss function to optimize accuracy while the latter focus on minimizing the magnitude of error in predictions i.e: reducing MAPE. In this context, we aim to bridge this gap in understanding by using both metrics to create a model of best fitting, then evaluate their accuracy in terms of signal direction.

In particular, we will attempt to address the following research question in the article :

1 - How does all the stock forecasting models- regression, KNN, LSTM, Conv1D LSTM compare in terms of accuracy and MAPE ?

With respect to each algorithm mentioned above, we will also seek answers to the followings:

2 - What is the impact of adding convolutional layers on top of LSTM network in terms of accuracy and MAPE ?

3- What is the impact of adding polynomial features to regression algorithm in terms of accuracy and MAPE ?

4 - What is the impact of adding extra features to KNN algorithm in terms of accuracy and MAPE ?

PROPOSED APPROACH

A. Raw data collection

At this stage, historical stock data is collected from 35 companies by crawling on finance.yahooo.com, nasdaq.com and vndirect.com. With respect to each individual stock, historical information consist of opening price, closing price, high, low and volume on a daily basis. These original raw features will act as a basis on which the feature engineering process will be further implemented and discussed for each algorithm of choice. Our purpose of selecting of 10 years time window to ensure the inclusion of both bullish and bearish trend over the period. The detailed description of each technical indicator will be provided in Table 1.

Table 1. Stock market technical indicators

Opening Price	Opening price for a specific trading day
High	Highest price in a specific day
Low	Lowest price in a specific day
Closing Price	Closing price for a specific trading day
Volume	Trading volume in a specific day
Date	Trading day in format of year/month/date

B. Evaluation metrics

All the performance metrics only concerns the closing price of stocks at the end of the day. From an investor perspective, signal from closing price is perhaps the most important indicator since the ability to foresee closing price will give one adequate time to decide on trading strategies and adjusting their position. The output from all of our algorithm will be series numerical values corresponding to prediction of closing prices in the future, from which comparison will be made with the test set to evaluate whether the directional prediction is correct and how far the

predicted series drift away from ground truth observations.

Since the aim of this article is not limited to evaluate the accuracy in predicting stock price movement i.e whether the closing price will rise the next day, we also employ Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE) to calculate the deviation of prediction with regards to ground truth value i.e how far the predicted lines are from ground truth observations. These metrics have also been employed in Nayak et al.,2015, Manahov et al.,2014 papers as they are excellent general purpose error metric for numerical predictions . Formula for calculation of RMSE and MAPE are as follows

$$\text{MAPE} = \frac{1}{T}\sum_{i=1}^{T}\left|\frac{d_i - \widehat{d}_i}{d_i}\right|$$

$$\text{RMSE} = \sqrt{\frac{1}{T}\sum_{i=1}^{T}\left(d_i - \widehat{d}_i\right)^2}$$

Where
T is total test sample;
d_i is real sample value;
$\hat{d}$ is the value predicted by our model.

Due to the inherent difference of MAPE and RMSE: the former is scale independent while the latter does not possess such property: only MAPE will be used for final comparison stage and RMSE will only be used for hyper parameters tuning process.

To make certain that the notation will be consistent over all the experiments, the general policy for direction classification will be as follows

$y = 1$ if $Close_{t+1} > Close_t$;
$y = 0$ if otherwise.

Accuracy is then simply calculated as the percentage of correct predictions over the total number of observations. Higher accuracy is likely to ensure a higher level of profit, thus better investment strategies. It is worth noting that other papers may refer to this accuracy rate as Hit ratio (Qiu and Song 2016) and despite some small differences in notation, the main intuition is similar. Therefore, in this paper, we will refer to this aforementioned metric as accuracy for readers convenience.

Out of financial data collected from 35 companies, only 3 were chosen to assess performance on: Alibaba, VinGroup and Pepsico. The selection of stock was to ensure an appropriate distribution of firms of different sizes and different financial markets from both developed and developing regions. A general consensus among academics with respect to most distinguishable characteristic from financial market in developed nations and developing nations is that equity market in developed regions of the world represent a stronger form of market efficiency, thus lowering the possibility of exploiting historic price data in the pursuit of abnormal financial gains. (John el at, 1994). While Alibaba is a multinational conglomerate company specializing in e-commerce, VinGroup focus more on real estate development and Pepsico is a multinational food, beverage and snack corporation, we expect our model to demonstrate predictive capability in stock movement in a variety of industries which inherently differ in business scale, trading frequency and market maturity etc. Test size of 0.2 applies for Alibaba stock, VinGroup stock and Pepsico stock.

C. Experimental Framework

1. Regression family (Linear Regression and Polynomial Regression)

a. Feature Engineering

We aim to predict the change in closing price at time step t using changes in closing price from N previous time steps. In other words, the difference values will be regressed on its N previous values. To determine the optimum N for each algorithm, a for loop is set to collect the values of RMSE and MAPE corresponding to each N, then optimal N will be the value where RMSE and MAPE reach its minimum as a function of N. In this paper, we will examine each value of N from N_{MIN} is 2 N_{MAX} is 60. The model hyperparameters will be tuned using the validation set size of 0.2, and the performance will be reported on test set size of 0.2

b. Data preprocessing

One of the important properties of regression model is that all the observations must be independent from one another.(Brooks, 2014) While this assumption usually holds true for the majority of cross sectional data analysis, this is not the case with time series data for the reason

$$P_{t+1.} = P_t + \varepsilon$$

Where

P_{t+1} is the price at time step t+1

P_t is the price at time step t

ε is the difference

For that reason we decide to take the difference of the time series of first order, we ensure that the observations are independent of one another, thus regression analysis is plausible.

c. Hyperparameters tuning

We utilized 2 modules from sklearn library to perform this experiment: sklearn.linear_model and sklearn.pipeline with PolynomialFeatures.

As for Linear Regression, sklearn model will converge at the point where residuals sum of squared is minimized, therefore no hyperparameters tuning is required at this stage

As for Polynomial Regression, GridSearchCV are used to ensure best hyper parameters are used to maximize coefficient of determination - R squared score in the process of fitting the prediction line to ground truth observations. The range of degree for polynomial kernel will be from 2 to 5.

2. K-Nearest Neighbor

a. Feature engineering

As our main variable of interest is closing price, from the original set of features (Opening Price, High, Low, Volume,Date), only Opening Price and Date will be kept. It is apparent that highest/lowest price and trading volume calculation can only be performed once the trading session has concluded, therefore incorporation of those features to report accuracy during testing process is equivalent to allowing the algorithm to cheat by foreseeing the future. This is an obvious misalignment with investors' expectations.

On top of 2 original features, a total of additional 12 features are generated using fastai modules. An in-depth description of extra 12 features is provided in Table 4

Table 2. Detailed description additional 12 features generated by fastai module

Year	Value of the year a specific trading date belongs to, ranging from 2009 to 2019
Month	Value of the month a specific trading date belongs to, ranging from 1-12
Week	Value of the week a specific trading date belongs to, ranging from 1 - 52
Day	Value of the day in month a specific trading date belongs to, ranging from 1 to 31
Day of week	Value of the day in week a specific trading date belongs to, ranging from 0 to 6
Day of year	Value of the day in year a specific trading date belongs to, ranging from 1 to 356
Is month end	Whether a specific trading date is the last day of the corresponding month, 1 or 0
Is month start	Whether a specific trading date is the first day of the corresponding month, 1 or 0
Is quarter end	Whether a specific

	trading date is the last day of the corresponding quarter, 1 or 0
Is quarter start	Whether a specific trading date is the first day of the corresponding quarter, 1 or 0
Is year end	Whether a specific trading date is the last day of the corresponding year, 1 or 0
Is year start	Whether a specific trading date is the first day of the corresponding year, 1 or 0

As for VinGroup in particular, we decide to experiment with 2 additional manually created features: Tetamlich and Viathantai. For Tetamlich, the value will be 1 if a specific date is within a week prior to Lunar new year holiday or a week after the holiday has concluded, and 0 if otherwise. For Viathantai, the value will be one if the date corresponds to Via Than Tai holiday and 0 if otherwise. It is customary for Vietnamese people to make an investment on above-mentioned dates as such acts are believed to bring good luck and fortune for investors, therefore 2 new features are added to ensure the impact on demand side of equity market of traditional holidays are accounted for in the model.

b. Data preprocessing

Regarding KNN algorithm, we use MinMaxScaler() function imported from sklearn.preprocessing modules to translate all features to a predetermined range from 0 to 1. For each value in a feature, MinMaxScaler subtract the minimum value in the feature then divide by the difference between original maximum and minimum value. One advantage of this transformation over other methods available in sklearn library was the preservation of original distribution while maintaining the significance of outliers.

c. Hyperparameters tuning

To predict the value of stock price in time step t+1, KNN algorithm determines the similarity between old data points and new data points by calculating Euclidean distance and assign a value using k neighbors. The next question to arise is then how many neighbors should be used for calculation. We attempt to determine the optimal k value by using a for loop to collect the values of RMSE and MAPE corresponding to each k. The optimal value of k is where RMSE and MAPE reach its minimum as a function of k. Examination will be conducted for each value of k from k_{MIN} is 2 k_{MAX} is 60.

3. Long Short Term Memory and 1D Convolutional Long Short-Term Memory

Long Short Term Memory (LSTM) networks is a subset of recurrent neural network, which distinguishes itself from standard feed-forward neural network by a feed-back loop. This special architecture allows for information from earlier time step retain in hidden state and continue to pass through following layers, which has successfully proved its predictive power in handling sequential data. Nonetheless, recurrent neural network greatly suffers from vanishing gradients problem as the sequence becomes longer and the introduction of LSTM was set to tackle this issue. The problem is addressed by introducing an alternative to artificial neuron in hidden layers: memory cell unit which consists of three gates known as input gate, forget gate and output gate. These gates will aid the procedure of weights adjusting and discarding information from the cell state when needed. Motivated by the suggestion of superiority of deep neural networks in dealing with non-linear model, evaluation of LSTM performance will be conducted in the following experiment.

1D Convolutional Long Short Term Memory (Conv1D-LSTM) network was proposed by F.Chollet (2018) as a method to combine the advantages of 2 separate deep learning models. Given the extensive

and successful application of Convolutional Neural Network (CNN) in Computer Vision tasks, 1D CNN are also found to be extremely effective for extracting features from sequential data by Yoon Kim (2014). In sentiment analysis and topic categorization experiments conducted by Yoon, 1D CNN achieved significantly better results compared with previous articles as a multitude of kernel types are expected to represent local patterns and capture data locality better, which is of great importance in handling signal analysis. In this paper, once LSTM performance has been assessed, we will stack two 1D CNN layers on top of the original network to extract the features instead of directly feeding them into the LSTM layers for further process.

a.Feature Engineering

The input for our models consist of 5 original features: Open, High, Low, Close and Volume in previous N look back trading days to predict closing price of 1 trading day into the future. Results acquired from a series of experiments on Alibaba stock price prediction with different look back days value: 5,10,15,20,30 and 60 suggest that 20 day window yield the best accuracy, therefore we will keep the number of look back days at 20 in the next subsection. A more comprehensive description of picking the best look back window size and optimizer will be provided in section IV.

b. Data Preprocessing

Similar to K Nearest Neighbor algorithm, we use MinMaxScaler() function from sklearn library to scale down all features within range of 0 and 1 before feeding data into the network. The same process is applied to testing data.

c. Network Architecture

Our models are inspired by Jain et al.,2018 with several modifications being made. LSTM network is composed of an input layer, followed by 2 LSTM layers and 2 fully connected layers while Conv1D-LSTM stacks 2 additional conv1D on top of the original architecture. Details for the models are provided as follows

LSTM Architecture

Input : 20x5
LSTM 128,activation= tanh
Dropout,keep rate = 0.5
LSTM 32, activation = tanh
Dropout,keep rate = 0.5
Fully Connected 16, activation = ReLU
Fully Connected 1,activation = Linear

Conv1D LSTM Architecture

Input : 20x5
Conv1D: 1x1x80,strides 1 + ReLU
MaxPool1D, pool size = 2
Conv1D: 1x1x48, strides 1 + ReLU
MaxPool1D,pool size = 2
Dropout,keep rate = 0.5
LSTM 128,activation= tanh
Dropout,keep rate = 0.5
LSTM 32, activation = tanh
Dropout,keep rate = 0.5
Fully Connected 16,activation = ReLU
Fully Connected 1,activation = Linear

Learning strategies are identical for both models. Instead of using default weights and bias initialization, we use Xavier and He initialization to ensure the maintenance of variance in weight gradients across layers (Xavier et al.,2010). We switch the activation function from ReLU in Jain et al., 2018 to Tangent Hyperbolic function (Tanh) to prevent exploding gradients phenomenon observed in our training phase. Mean Squared Error is used as loss function in addition with L2 regularization techniques to prevent potential model overfitting

phenomenon. Our optimizer of choice is RMSprop and we also employ reduce learning rate strategy with a factor of 0.1 for each 3 epochs in which the loss of validation set (size 0.3) reached a plateau. Batch size is maintained at 16 and the models are trained on 30 epochs across experiments. The number of trainable parameters for LSTM network is 89761 while Conv1D LSTM is 116145

RESULT AND EVALUATION

As our experiment will concern the accuracy of label on stock price movement (1 for increase and 0 for otherwise), it is essential to make certain the class label are fairly distributed for each stock. The distribution for class label are provided in Table 5.

Table 3. Class label distribution corresponding to each stock

	Class 0	Class 1
Pepsico	46.71%	53.29%
Alibaba	52.89%	47.11%
VinGroup	43.97%	56.03%

1. Regression family (Linear Regression and Polynomial Regression)

The first step in constructing regression model is choosing the N number of prior values that will be regressed on to make the prediction. Optimal number of N is then chosen as the minima of the function of RMSE and MAPE with respect to N. Assessment will be conducted for the range $N_{MIN} = 2$ and $N_{MAX} = 60$. Values for optimal N are provided in the following table

Table 4. Optimal N for regression algorithms

	Linear Regression	Polynomial Regression
Pepsico	50	50
Alibaba	25	50
VinGroup	59	50

Once the optimal N has been determined, result for our regression models are provided in the following table

Table 5. Regression family results

	Linear Regression		Polynomial Regression	
	Acc	MAPE	Acc	MAPE
Pepsico	51.89	203.7	47.31	212.94
Alibaba	56.01	200.09	49.79	223.74
VinGroup	48.39	112.43	54.81	123.5

Illustration of prediction series generated by regression algorithms are provided in the source code

2. K-Nearest Neighbor

Similar to regression algorithms, firstly the optimal number K of nearest neighbors are chosen as minima of RMSE and MAPE as a function of K. The optimal number of K is provided in the following table

Table 6. Optimal K for K-Nearest Neighbor algorithm

	Optimal K
Pepsico	59
Alibaba	10
VinGroup	49

Once the optimal K has been determined, we will experiment with 2 additional features: Tetamlich and Viathantai in the case of VinGroup, on top of 13 initial features including ones generated by fastai module

and original features. Result for K-Nearest Neighbor algorithm on are provided in the following table

Table 7. K-Nearest Neighbor results with 13 initial features

	Accuracy	MAPE
Pepsico	53.47	47.61
Alibaba	65.56	48.05
VinGroup	51.49	90.71

Impact of additional features

After generating 2 additional features for VinGroup, we calculate the accuracy and MAPE for VinGroup again and found the accuracy surged from 51.49% to 57.02% while MAPE metrics went down from 90.71% to 84.81%, suggesting two features manually generated based on knowledge of Vietnamese traditions and customs increase the predictive power of KNN algorithm

3. LSTM Family (LSTM and Conv1D-LSTM)

Before we establish the final version of LSTM for predictions, the choice of number of look back days and optimizer needs to be carefully taken into consideration. For this reason, evaluation will be conducted solely on Alibaba stock prices to determine the optimal number of look back days and optimizer. Our LSTM model will run with 5 different look back days values: 5,10,20,30,60 and 3 different optimizer: Adam, RMSprop and Stochastic Gradient Descent (SGD) with learning rate at 0.001, momentum at 0.9. Then the optimal look back days and optimizer will be determined by the values corresponding to lowest MAPE observed

Table 8. Variations in multiple look back window (RMSprop optimizer for all)

Look back days	MAPE
5	9.36
10	10.47
20	8.24
30	11.52
60	13.49

Once the size of look back window has been determined at 20, we experiment with different optimizer

Table 9. Variations in multiple optimizers

Optimizer	MAPE
Adam	10.93
RMSprop	8.24
SGD	9.67

Results obtained from these former experiments clearly suggest the optimal size of look back window is 20 and the optimal optimizer is RMSprop. Therefore the choice of look back window size and optimizer will be kept for the remainder of this paper.

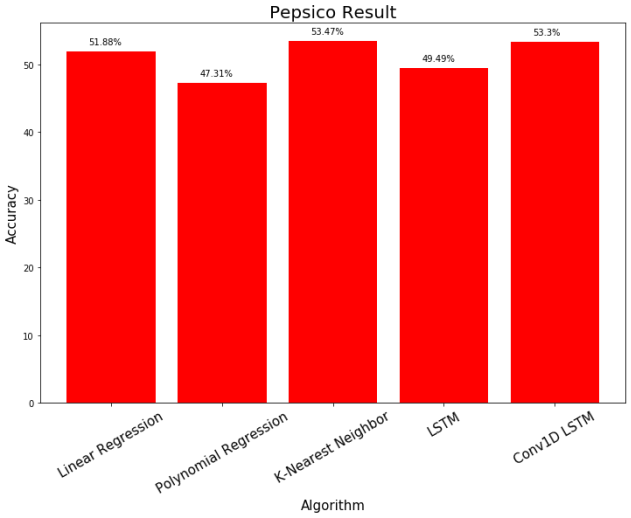
After important hyperparameters such as look back days and optimizer has been fine tuned, the following table will report the result of LSTM network and Conv1D LSTM network on predicting stock prices:

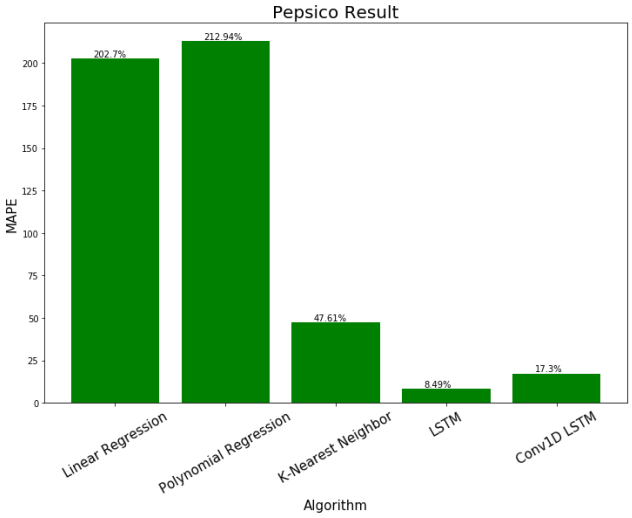
Table 10. LSTM and Conv1D LSTM results

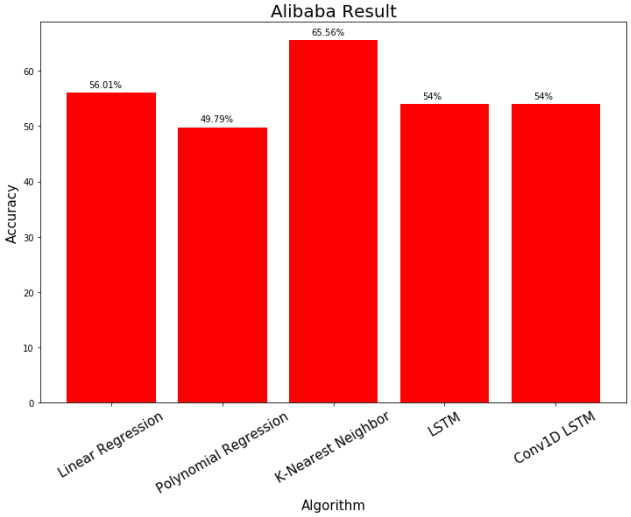
	LSTM		Conv1D LSTM	
	Acc	MAPE	Acc	MAPE
Pepsico	49.49	8.49	53.30	17.30
Alibaba	54	8.24	54	17.95
VinGroup	52.63	18.39	55.06	25.18

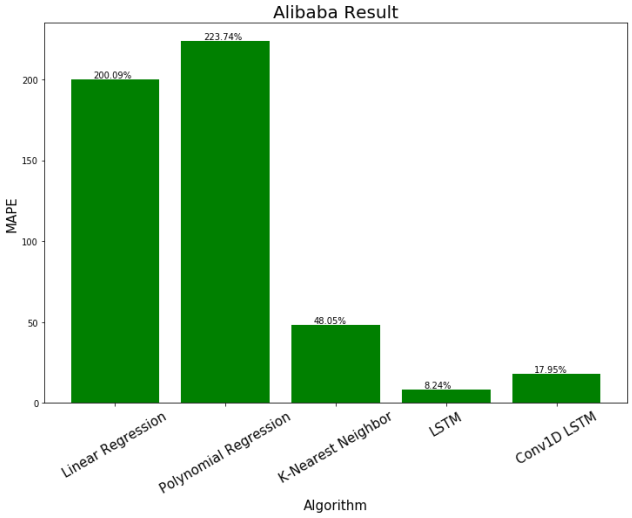
4. A comparison over all algorithms

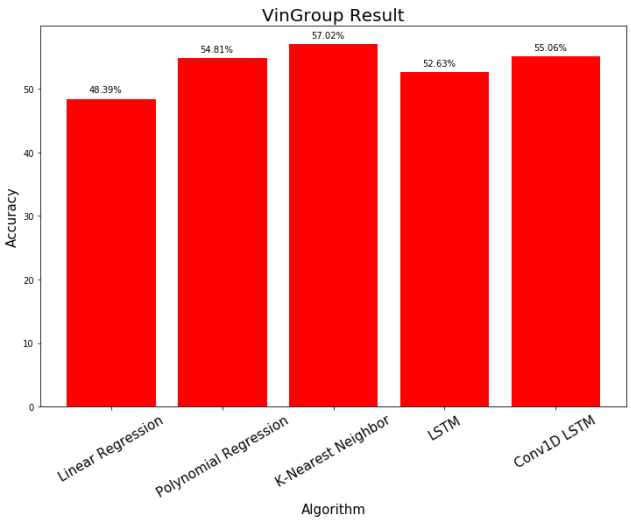
The following figures will illustrate the performance of all the algorithms being used in this experiment.

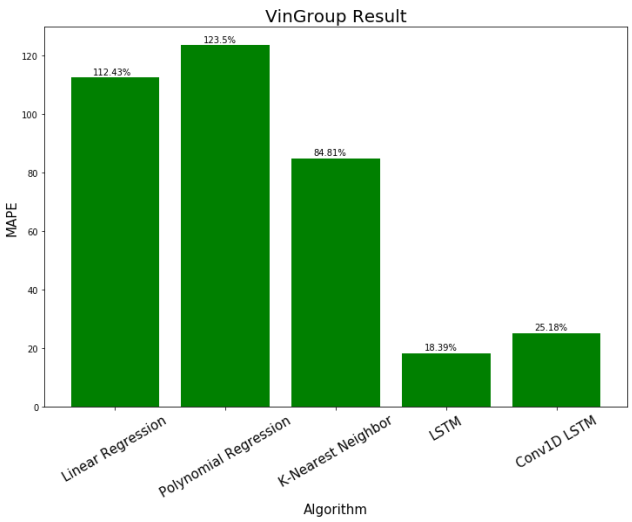

Pepsico Result
Accuracy
Algorithm
51.88%
47.31%
53.47%
49.49%
53.3%
Linear Regression
Polynomial Regression
K-Nearest Neighbor
LSTM
Conv1D LSTM


Pepsico Result
MAPE
Algorithm
202.7%
212.94%
47.61%
8.49%
17.3%
Linear Regression
Polynomial Regression
K-Nearest Neighbor
LSTM
Conv1D LSTM


Alibaba Result
Accuracy
Algorithm
56.01%
49.79%
65.56%
54%
54%
Linear Regression
Polynomial Regression
K-Nearest Neighbor
LSTM
Conv1D LSTM


Alibaba Result
MAPE
Algorithm
200.09%
223.74%
48.05%
8.24%
17.95%
Linear Regression
Polynomial Regression
K-Nearest Neighbor
LSTM
Conv1D LSTM


VinGroup Result
Accuracy
Algorithm
48.39%
54.81%
57.02%
52.63%
55.06%
Linear Regression
Polynomial Regression
K-Nearest Neighbor
LSTM
Conv1D LSTM


VinGroup Result
MAPE
Algorithm
112.43%
123.5%
84.81%
18.39%
25.18%
Linear Regression
Polynomial Regression
K-Nearest Neighbor
LSTM
Conv1D LSTM

Empirical evidence obtained from this experiment clearly suggest family of LSTM network produced models that fit better to ground truth observations, evidenced by MAPE metrics is consistently lower across experiments. From our illustration, it is apparent that LSTM models generate best fitting prediction and KNN models produce most accurate predictions. The accuracy peaked at 65.56% using K-Nearest Neighbor in case of Alibaba stocks and also higher than that generated by other algorithms

Impact of polynomial features
Regarding regression family, the higher accuracy is achieved by using Linear Regression in the case of Alibaba with N=25, while the MAPE metrics of polynomial regression is higher for 3 out of 3 cases provided, indicating the addition of polynomial features did not improve the accuracy of the algorithm overall and models generated by polynomial regression has lower goodness of fit score in comparison with its linear counterparts.

Impact of Conv1D layers
The addition of two Conv1D layers stacked on top of original LSTM network did contribute to higher accuracy overall, yet MAPE metrics is marginally higher in 3 out of 3 companies tested, suggesting a worse fitting model.

CONCLUSION

In order to aid the understanding of machine learning and deep learning models applicability in stock forecasting model, this study has aimed to investigate and present an experiment regarding predictive capability of a subset of most well known and widely used algorithms. By using three conventional models as baseline: Linear Regression, Polynomial Regression and K-Nearest Neighbor, we assess most advanced Deep learning network performance in predicting price and directional movement of 3 stocks: Pepsico, Alibaba and VinGroup in terms of accuracy and goodness of fit score (MAPE). Results obtained from this experiment suggests that while LSTM model can produce model with highest goodness of fit, K-Nearest Neighbor proved to be most effective for stock trend predictions. While the modified version of LSTM network with two 1D convolutional layers stacked on top performed better in terms of accuracy, errors produced by Conv1D LSTM deviate further away from ground truth observation in comparison with the original LSTM. We also found the addition of polynomial features did not improve the regression model both in terms of MAPE and accuracy. With respect to the difference in using extra features in K-Nearest Neighbor, directional accuracy of VinGroup rose by nearly 6% while a decrease in MAPE was observed, indicating higher predictive power overall.

Future Direction
To further facilitate the application of artificial intelligence in stock market forecasting models, we aim to conduct more experiments with various types of neural network architecture as well as study different approaches for features engineering.

REFERENCES

Malkiel BG, Fama EF, Efficient capital markets: a review of theory and empirical work. J. Finance. 1970, 25(2): 383-417

Glantz, M & Kissell, R. (2013). Multi-Asset Risk Modeling: Techniques for a Global Economy in an Electronic and Algorithmic Trading Era. Multi-Asset Risk Modeling: Techniques for a Global Economy in an Electronic and Algorithmic Trading Era. 1-516.

Nelson, David & Pereira, Adriano & de Oliveira, Renato. (2017). Stock market's price movement prediction with LSTM neural networks. 1419-1426. 10.1109/IJCNN.2017.7966019.

A. E. Biondo, A. Pluchino, A. Rapisarda, and D. Helbing, “Are Random Trading Strategies More Successful than Technical Ones?” PLoS ONE, vol. 8, p. e68344, Jul. 2013.

Malkiel BG. The efficient market hypothesis and its critics. J Econ Perspect. 2003l 17(1): 59-82

Hsu M-W, Lessmann S, Sung M-C, Ma T, Johnson JE. Bridging the divide in financial market forecasting: machine learners vs financial economists. Expert Syst Appl. 2016;61(1):215-234

Ballings M, Van den Poel D, Hespeels N, Gryp R. Evaluating multiple classifiers for stock price direction prediction. Expert Syst Appl. 2015;42(20):7046-7056

Shah, Mitesh A. and Chetna D. Bhavsar. "Predicting Stock Market using Regression Technique." (2015).

Huynh, Huy & Dang, L. Minh & Duong, Duc. (2017). A New Model for Stock Price Movements Prediction Using Deep Neural Network. 57-62. 10.1145/3155133.3155202.

Jiawei, X. and Murata, T. (2019). Stock Market Trend Prediction with Sentiment Analysis based on LSTM Neural Network. *International MultiConference of Engineers and Computer Scientists 2019 IMECS 2019*, March 13-15(2078-0958).

Jain, S., Gupta, R. and A.Moghe, A. (2018). Stock Price Prediction on Daily Stock Data using Deep Neural Networks. *International Journal of Neural Networks and Advanced Applications*, Volume 5, 2018(2313-0563).

Kim, Sangyeon & Kang, Myungjoo. (2019). Financial series prediction using Attention LSTM.

Dickinson, J. P. and Muragu, K. (1994), MARKET EFFICIENCY IN DEVELOPING COUNTRIES: A CASE STUDY OF THE NAIROBI STOCK EXCHANGE. Journal of Business Finance & Accounting, 21: 133-15-0.

Nayak RK, Mishra D, Rath AK. A Naive SVM-KNN based stock market trend reversal analysis for Indian benchmark indices. Appl. Soft Comput. 2015;35(1):670-680

Manahov V, Hudson R, Gebka B. Does high frequency trading affect technical analysis and market efficiency? And if so, how? Journal of International financial markets. Inst Money. 2014;28(1):135-157

Brooks, C. (2019). A Brief Overview of the Classical Linear Regression Model. In *Introductory Econometrics for Finance* (pp. 94-145). Cambridge: Cambridge University Press. doi:10.1017/9781108524872.005

F. Chollet "Deep Learning with Python" , Manning Publications Co. , Chap 6 , 2018, pp. 225-232

Yoon Kim. Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882, 2014.

Glorot, Xavier and Yoshua Bengio. "Understanding the difficulty of training deep feedforward neural networks." *AISTATS* (2010).

Qiu M., Song Y., Akagi F. (2016). Application of Artificial Neural Network for the Prediction of Stock Market Returns: The Case of the Japanese Stock Market. Chaos, Solitons and Fractals 85:1-7.