جــامــعــة الــشــرق الأوســط
MIDDLE EAST UNIVERSITY
Amman - Jordan

Jordanian Companies’ Stock Price Prediction using Hybrid RNN with Long Term Short Memory and Tabu List Memory

تنبؤ أسعار أسهم الشركات الأردنية بأستخدام الشبكات العصبية المتكررة الهجينة مع ذاكرة طويلة قصيرة المدى وذاكرة قائمة Tabu

By:

Abdullah Akram Al-Jumaili

Supervisor:

Dr. Hassan Al-Tarawneh

Submitted in Partial Fulfilment of the Requirements for the

Master Degree in Computer Science

Computer Science Department

Faculty of Information Technology

Middle East University

August, 2020

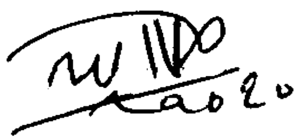
Authorization

I, **Abdullah Akram Al-Jumaili**, authorize Middle East University to provide Libraries, Organizations and Individuals with copies of my thesis on request.

Name: Abdullah Akram Al-Jumaili

Date: 18 / 10 / 2020

Signature:



Examination Committee Decision

This is to certify that the thesis entitled Jordanian Companies' Stock Price Prediction using Hybrid RNN with Long Term Short Memory and Tabu List Memory" was successfully defended and approved on 18/10/2020.

Examination Committee Members	Signature
(Supervisor) **Dr. Hassan Al-Tarawneh** **Middle East University**	
(Chairman of Examination Committee and Internal Committee Member) **Dr. Bassam Al-Shargabi** **Middle East University**	
(External Committee Member) **Dr. Ayesh Al-Hroub** **Al-Isra University**	

Acknowledgment

(الحمد لله على نعمه السابغات المتواليات وعلى رسوله الامين افضل الصلوات)

To my parents, who believed in me, motivated & helped me to achieve the impossible, to my sisters and finally I'll never forget the people who supported & encouraged me to achieve this research and make it real.

The Researcher

Abdullah Akram Al-Jumaili

Dedication

بسم الله الرحمن الرحيم

"وقل رب زدني علما"

This thesis is dedicated:

To my country Iraq and my second country Jordan.

To those whom my thanks do not fulfil them my Father and my Mother.

To the most precious people l know my siblings: Nagham, Muhammad and Zainab

Jordanian Companies' Stock Price Prediction using Hybrid RNN with Long Term Short Memory and Tabu List Memory

By: Abdullah Akram Al-Jumaili

Supervisor: Dr. Hassan Al-Tarawneh

Abstract

Stock price prediction is one of the most challenging tasks for cooperation and how they can make a decision according to this prediction. The difficulty of stock price prediction raising due to many factors such as market changes and globalization, which pose a challenge for analyzing stock market movements and price behaviors is extremely challenging. The effective expectation of a stock's future cost could provide business gain and positive impact on the company. This research propose method Hybrid RNN with Long Term Short Memory and Tabu List and test the proposed method on the benchmark, then apply the method for real Jordanian companies' historical business records and check predicted stock prices produced by using the proposed method with real stock prices. Daily stock data has been collected from the Amman Stock Exchange (ASE) in order to train and test the proposed model. This work aims to provide an accurate method of stock pricing prediction and improve prediction accuracy by removing predicted prices that are far from the real stock price and redundant predictions, to offer a consistent method for the Jordanian companies to survive and flourish in market by analyzing their performance and predict stock price in future to provide them with a tool to avoid bad business decisions and improve their services. The accuracy of the proposed model is high and promising as the accuracy of RNN with LSTM and Tabu list were better than using only RNN with LSTM as shown in the results, which means that the proposed model can be utilized as a reliable tool for stock prices prediction.

Keyword: Recurrent Neural Network, Long Short Term Memory, Tabu List, Stock Price Prediction

تنبؤ أسعار أسهم الشركات الأردنية بأستخدام الشبكات العصبية المتكررة الهجينة مع ذاكرة طويلة قصيرة المدى وذاكرة قائمة Tabu

إعداد: عبد الله أكرم الجميلي

إشراف: الدكتور حسان الطراونة

الملخص

يعد توقع سعر السهم أحد أكثر المهام تحديًا للشركات وكيفية اتخاذ القرار وفقًا لهذا التوقع. تكمن الصعوبة التنبؤ بأسعار الأسهم لكثير من العوامل مثل تغيرات السوق والعولمة الاقتصادية، والتي تشكل تحديًا لتحليل تحركات سوق الأسهم وسلوكيات الأسعار، أمرًا صعبًا للغاية. التوقع الفعال لسعر السهم في المستقبل يمكن أن يوفر مكاسب مادية وتأثيرًا إيجابيًا على الشركة. يقترح هذا البحث طريقة الشبكات العصبية المتكررة الهجينة مع ذاكرة طويلة قصيرة المدى وذاكرة قائمة Tabu واختبار الطريقة المقترحة على المعيار، ثم تطبيق الطريقة على السجلات أسعار الاسهم للشركات الأردنية الحقيقية والتحقق من أسعار الأسهم المتوقعة الناتجة باستخدام الطريقة المقترحة مع أسعار الأسهم الحقيقية. تم جمع بيانات أسعار الاسهم اليومية من بورصة عمان لتدريب واختبار النموذج المقترح. يهدف هذا العمل إلى توفير طريقة دقيقة للتنبؤ بسعر الأسهم وتحسين دقة التنبؤ من خلال إزالة الأسعار المتوقعة البعيدة عن سعر السهم الحقيقي والتنبؤات المتكررة، لتقديم طريقة مضمونة للشركات الأردنية للبقاء والازدهار في السوق من خلال تحليل الأداء والتنبؤ بسعر السهم في المستقبل لتزويدهم بأداة لتجنب قرارات العمل السيئة وتحسين خدماتهم. دقة النموذج المقترح عالية وواعدة حيث أن دقة الشبكات العصبية المتكررة الهجينة مع ذاكرة طويلة قصيرة المدى وذاكرة قائمة Tabu كانت أفضل من استخدام الشبكات العصبية المتكررة الهجينة مع ذاكرة طويلة قصيرة المدى فقط كما هو موضح في النتائج، مما يعني أنه يمكن استخدام النموذج المقترح كأداة موثوقة لتنبؤ أسعار الأسهم.

الكلمات المفتاحيه : الشبكات العصبية المتكررة، ذاكرة طويلة قصيرة المدى، ذاكرة قائمة Tabu، تنبؤ أسعار ألاسهم.

Table of contents

List of Figures

List of Tables

Table of Abbreviations

Abbreviations	Full Form
RNN	Recurrent Neural Network
LSTM	Long Short Term Memory
TL	Tabu List
TS	Tabu Search
CNN	Convolutional Neural Network
ASE	Amman Stock Exchange
DWT	Discrete Wavelet Transform
BTT	Backpropagation Through Time
GA	Genetic Algorithm
MSE	Mean Squared Error
MAE	Mean Absolute Error
RMSE	Root Mean Squared Error
DRNN	Discrete Wavelet Transform with Recurrent Neural Network
ARIMA	Auto Regressive Integrated Moving Average

Chapter 1

Introduction

Modeling and forecasting of the money related market has been an appealing subject to researchers and analysts from different scholastic fields. Stock value forecast is one of the most testing errands for participation and how they can settle on a choice as indicated by this expectation. In the current situation of finance related global market, particularly in the stocks exchange, estimating the cost of stocks utilizing different methods and intelligent neural networks are the most appealing issue to be examined. The issue of stock value prediction raising because of numerous elements, for example, showcase changes and globalization, which act like a test for breaking down securities exchange developments and value practices is incredibly testing result of business sectors dynamic, nonlinear, nonstationary, nonparametric, boisterous, and tumultuous nature (Al-Radaideh, et al., 2013).

Moreover, predicting stock price movement is suffering from various factors (for example: political events, general economic conditions, and traders' expectation) within the stock exchange. Therefore, choosing appropriate input features for prediction models should be considered. Companies in Jordan are facing a lot of issues that affect their growth and market survivability which consider as a major obstacle for investors and entrepreneurs, this problem affects Jordan's economy and becomes an obstacle for economic growth.

The presented work of this thesis highlights the great importance of using modified RNN (Recurrent Neural Network) with Tabu List memory which represents short-term memory structure of Tabu Search that consist of a list of previous solutions that must be avoided or list of moves that are forbidden and LSTM (Long Short Term Memory), to enhance the efficiency of prediction and to avoid outlier predictions.

1.1 Problem Statement

The task of predicting stock market prices changes in the future is a challenging, researchers' attention attracted greatly for many years in predicting stock prices, it involves an assumption of fundamental information that is publicly available in the past that has some projecting relationships to the future stock prices.

Most prediction methods struggle to provide accurate prediction for long period of time, there are many researchers trying to find a way to tackle this problem, providing different types of methods to approach that goal.

Researches proposed using machine learning algorithms for the purpose of stock price prediction because they possess the ability to learn from historical information and utilize it for future stock prices prediction, we chose neural networks due to the effectiveness demonstrated in previously published researches.

The stock exchange market depicts savings and investments that are advantageous to increase the effectiveness of Jordanian economy and help it flourish, future stock returns have some predictive relationships with the publicly available information of present and historical stock market indices, every company is interested in predicting the future stock prices, this topic possesses a major challenge to design and develop an efficient predictive model that assists the Jordanian companies to take appropriate decisions.

Stocks prices turmoil can lead to financial disasters for companies, this scenario poses a risk for entrepreneurs who want to start their business and startup companies want to grow and flourish in the market, this issue motivated me to provide reliable prediction tool that excels in forecasting stock prices changes in the future better than previously proposed methods.

1.2 Aim and Objectives

The ability to predict future stock market prices is extremely helpful to a corporation as it allows for a more controlled services environment with good preparation for market challenges and rapid changes in the business environment so corporations can make right decisions in a protective manner.

This work aims to provide a long-term stock market prediction based on RNN with LSTM and Tabu list approach to be utilized in historical data for the company. The main objectives of the proposed work is to improve stock price prediction accuracy using modified RNN with LSTM and Tabu list by proposing a new prediction model that provide the ability to eliminate missing or predictions that far away from real price by comparing it to real stock prices during the testing phase.

1.3 Research Questions

The research question for this study are:

1. How well does an RNN with LSTM and Tabu list predict daily stocks' prices of the Amman Stock Exchange dataset?
2. Can the RNN with LSTM and Tabu list provide a reliable stock price prediction and improved prediction accuracy better than the previous researches?

Chapter 2

Background and Related Works

This chapter presents a summary of RNN, LSTM and Tabu list as well as detailed description of RNN, LSTM and Tabu List and their advantages and disadvantages on stock prediction process.

2.1 Recurrent Neural Network

Recurrent neural networks (RNN) are a family of flexible dynamic models that connect artificial neurons over time. RNN is a multi-layered neural network that store information in nodes, enabling it to learn data series and produce an out data series. RNNs are applied in deep learning and in improvement of models that imitate the action of neurons in the human brain (Yao, et al., 2015).

They are effective to be used in environments in which context is essential for result prediction, it’s unique from the artificial neural networks because the usage feedback loops to process a sequence of data that informs final output, which can also be a sequence of data. These feedback loops allow information to be persevered. RNN is mainly used for time series analysis because it has feedback connections inside the network that allow past information to persist, and time series and nonlinear prediction capabilities. The temporal representation capabilities of RNN have advantages in tasks that process sequential data, such as financial predictions, natural language processing, and speech recognition. RNN has a high dimensional, continuous, representation of the latent state. A notable advantage of the richer representation of RNNs is their ability to use information from an input in a prediction at a much later point in time (Koutnik, et al., 2014).

RNN is the earliest of its style algorithms that can remember earlier inputs in memory, when a huge set of sequential data is given to it. RNN predict the next move based on the past moves that happened before (Yao, et al., 2015) (Koutnik, et al., 2014).

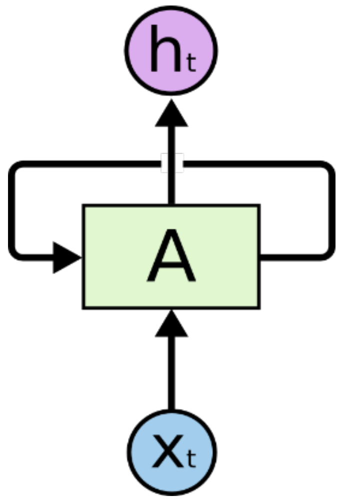


Figure 1. RNN loop [1]

In figure 1 part of neural network ***A***, it has input x_t and output with value h_t. The (input x_t → Action ***A*** → output h_t) loop allows information to be passed from the past step of the RRN to the next step by forwarding results from the last step to the next step, so the previous outputs will not be lost or disregarded in the process (DiPietro, et al., 2020).

These loops make RNN enigmatic, it can be thought of as multiple copies of the same network, each passing a message to a successor. If we separate the loop (figure 2), we will have a chain similar nature uncovers that RNNs are related to sequences and lists. They are the natural architecture of the neural network to use for such data.

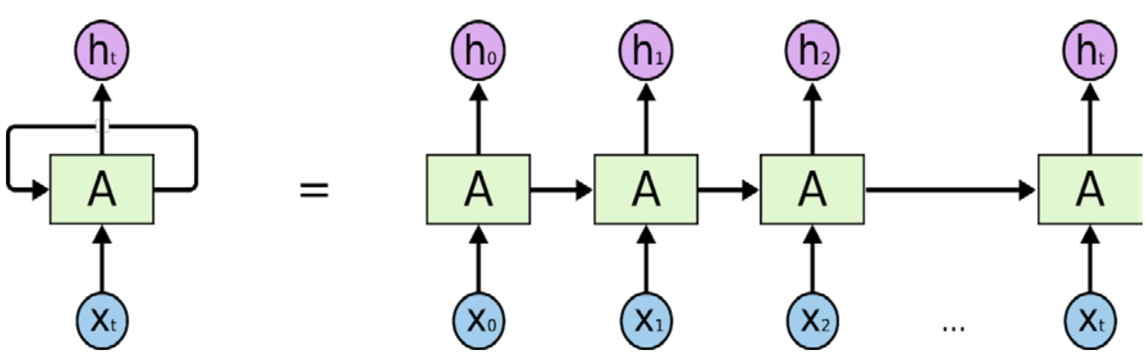


Figure 2. Unrolled RNN [1]

[1] Images from Christopher Olah website by https://colah.github.io/posts/2015-08-Understanding-LSTMs

So, what makes RNNs better than traditional Neural Networks? The answer is: there is a major limitation of traditional Neural Networks suffer from constraints (Sangaiah, 2019):

1) Fixed-sized inputs & outputs

2) No memory, no feedback.

Researchers used RNN in stock pricing prediction, for example Xie, et al utilized RNN to anticipate stocks exchange changes, the information taken from numerous organizations in China Shanghai Shenzhen 300 Index (Xie, et al., 2017).

Wei, et al, utilized RNN to anticipate Taiwan financial exchange in by applying four techniques: (connection grid, stepwise relapse, choice tree, and Elman NN) in the forecast procedure (Wei, et al., 2012).

Jarrah, et al., implemented two methods RNN and Discrete Wavelet Transform (DWT) used for Saudi stock price trends prediction, DWT addressed the issue of predicting the direction of motion of stock market index along with stock prices, RNN predicted future value of the stock market index (Jarrah, et al., 2019).

2.1.1 Types of RNNs

1. One to one: Vanilla mode of processing without RNN, from fixed-sized input to fixed-sized output (e.g. image classification).
2. One to many: Sequence output (e.g. image captioning which takes an image and outputs a sentence of words).
3. Many to one: Sequence input (e.g. emotion analysis where a provided paragraph is analyzed for stating a positive or negative emotion).

4. Many to many: Sequence input and sequence output (e.g. machine translation: RNN interprets a paragraph in English and then shows the paragraph in Spanish).
5. Many to many (with synchronization): Synchronized sequence input and output (e.g. video classification to categorize each frame from the video).

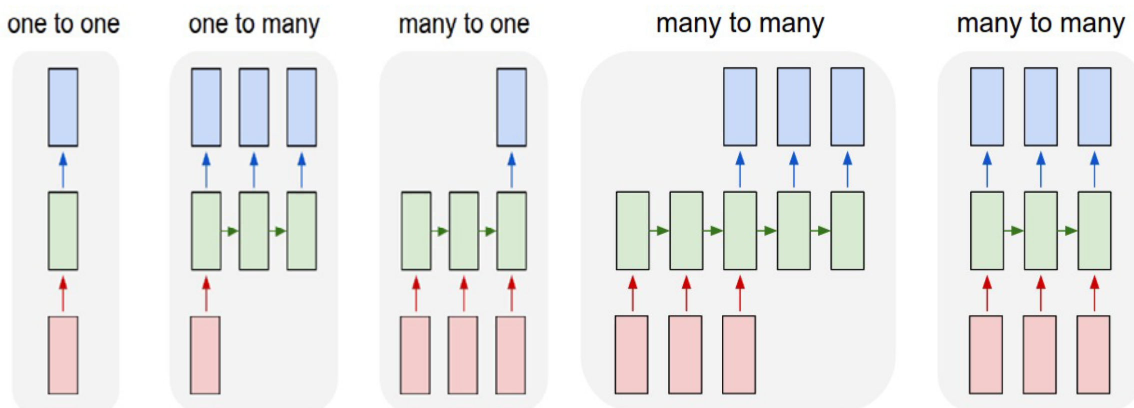


Figure 3. Types of RNNS, Input vectors are colored in red, output vectors are colored in blue and vectors colored in green vectors hold the RNN's state [2]

2.1.2. RNN Drawbacks

RNN suffers from two drawbacks:

1. RNN relies on recent information to perform the present task but it has no memory to store past information for long period, for example: if RNN used to predict what dish from the menu it will be served in a restaurant based on the previously served dish in the day before. When the information about the last served dish in the past day is missing, RNN can fix the problem of missing information about the last dish served by guessing which dish might

[2] Images from DeepAI website by https://deepai.org/machine-learning-glossary-and-terms/recurrent-neural-network

be served in the past day to predict the next served dish in the next day, this process is to overcome the lack of information that might affect the next step (Yao, et al, 2015).

But what happens if RNN did not have information about the past two weeks served dishes, RNN will not be able to predict the next dish to be served due to lack of information about the previous time.

2. RNNs use backpropagation algorithm for the purpose of training and it is applied for every timestamp which is called Backpropagation Through Time (BTT) as illustrated in figure 4 (Koutnik, et al., 2014) (Sangaiah, 2019).

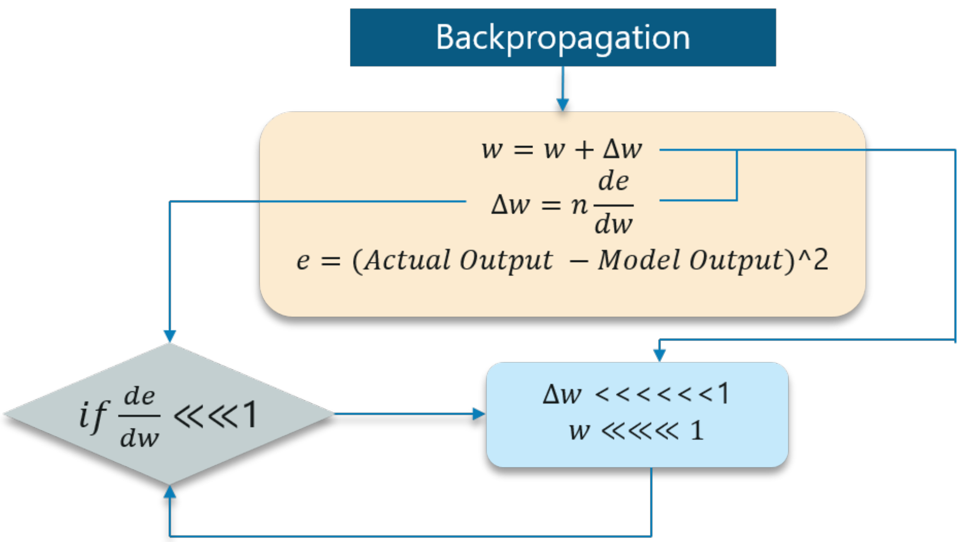


Figure 4. Backpropagation in RNN [3]

the BTT goal is to determine the error by calculating the difference between the output and optimal output and raise to a power of 2. When the error assessed changes in the error regarding the change in the weight is considered. But with each learning rate, this must be multiplied with the same (Yao, et al., 2015) (Koutnik, et al., 2014).

[3] Images from Anirudh Rao website by https://www.edureka.co/blog/recurrent-neural-networks/

The issues can arise from using backpropagation are:

a. Vanishing Gradient: A lot of iterations will cause the new weights to be extremely negligible and this leads to the weights not being updated, thus it will vanish (Hochreiter, 1998).

b. Exploding Gradient: The working of the exploding gradient is similar, but the weights here change drastically instead of negligible change (Pascanu, et al, 2012).

If an error happened in one of the RNN inputs this will result to a false prediction and this will affect the accuracy of next predictions and it will RNN mostly need recent data to perform the requested functions, but sometimes the requirement for the data that they were collected or obtained in the past (long-term dependencies), that is when LSTM (Long Short Term Memory) networks come in hand, LSTM eliminate issues rise with long-term dependencies like vanishing or exploding gradients.

2.2 Long Short Term Memory

In 1997, a modification of RNN with Long Short-Term Memory units (LSTMs), was suggested by German researchers Sepp Hochreiter and Juergen Schmidhuber as a resolution to vanishing gradient problem (Hochreiter, et al, 1998).

LSTM is an artificial recurrent neural network (RNN) architecture capable of learning long-term dependencies and retrieving information for long periods of time is practically their default behavior, not something they struggle to learn LSTM networks are a more complex variant of RNNs that often prove more powerful, they proved their importance for the more naturally retain information for many time steps, which is supposed to make them train easier than RNN.

Moreover, hidden units are updated using interactions with multiplication and they can perform more complicated transformations for the same number of latent units (Hochreiter, et al, 1998). The major difference between LSTM and RNN, is that RNN contains one single function as illustrated in figure 5, the single function consists of *tanh* function that exists in the layer, *tanh* function considered as a squashing function that used for a range of values between -1 to 1 and change the values according to input values.

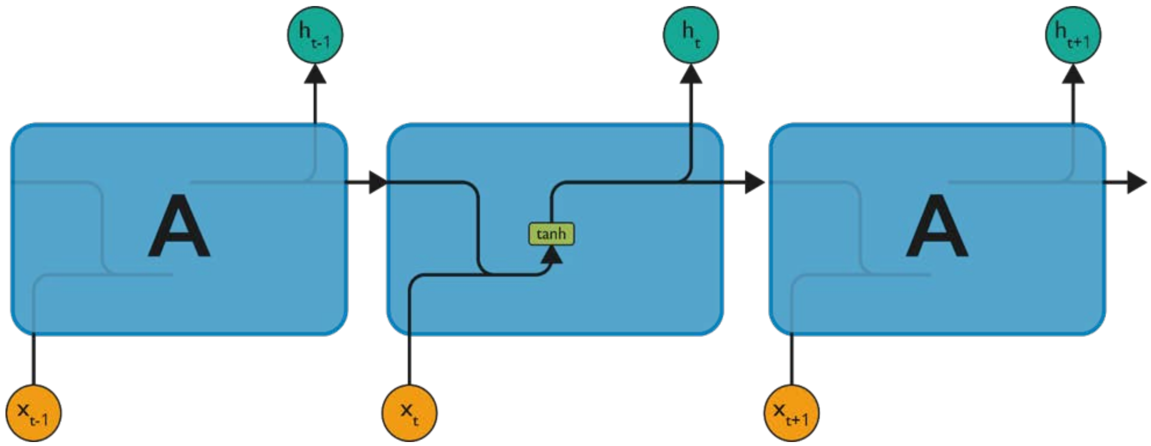


Figure 5. tanh function [4]

Unlike RNN, LSTM (figure 6) has more functions in LSTM layers, each layer has structure and processes the input data to forward it to the next layer in a horizontal way across the data channel

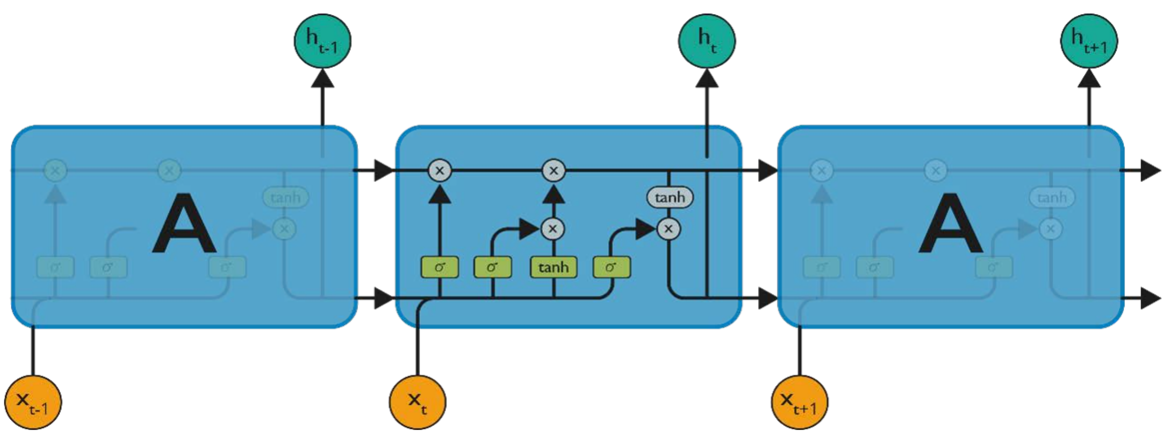


Figure 6. LSTM layers [4]

[4] Images from DeepAI website by https://deepai.org/machine-learning-glossary-and-terms/recurrent-neural-network

LSTM steps as follows (Hochreiter, et al, 1998):

1. **<u>Identifying</u>:** this layer function is to identify the information that will not be required for further processing and discard it out of the cell, this process is done be Sigmoid layer called forget gate layer as shown in figure 7.

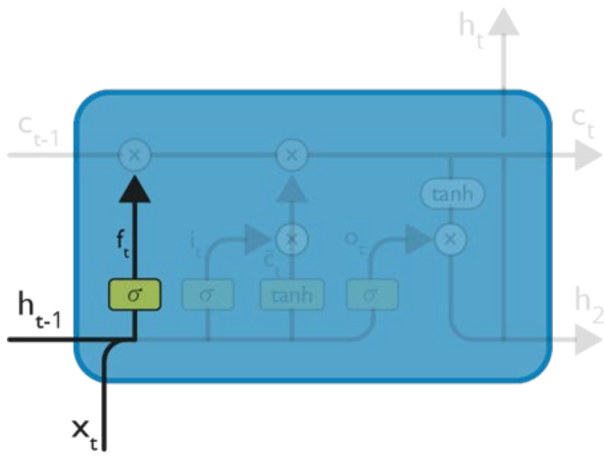


Figure 7. Forget gate [4]

Sigmoid layer produces an output as a number between 0 (discard previous cell state) and 1 (keep previous cell state) based on calculation done by using the following formula:

$$f_t = \sigma \left(w_f \; [h_{t-1}, x_t] + b_f \right)$$ (Varsamopoulos, et al, 2018)

w_f = *Weight*

h_{t-1}= *Output from previous timestamp*

x_t = *New input*

b_f = *Bias*

2. **<u>Deciding</u>:** this step will decide whether to store new information in the cell state or not, the decision process consists of two steps as follows (figure 8):

   a. Sigmoid layer called forget gate layer, it will decide which values will be updated.

[4] Images from DeepAI website by https://deepai.org/machine-learning-glossary-and-terms/recurrent-neural-network

b. *tanh* layer makes a vector of new candidate values $C\tilde{}_t$ that can be added to the cell state.

New input and previous timestamp are entered into sigmoid function that results in the value i_t then it is multiplied by c_t and then added to cell state.

$$i_t = \sigma (w_i \; [h_{t-1} , x_t] + b_i) \qquad \text{(Varsamopoulos, et al, 2018)}$$

$$C\tilde{}_t = tanh (w_c \; [h_{t-1} , x_t] + b_c) \qquad \text{(Varsamopoulos, et al, 2018)}$$

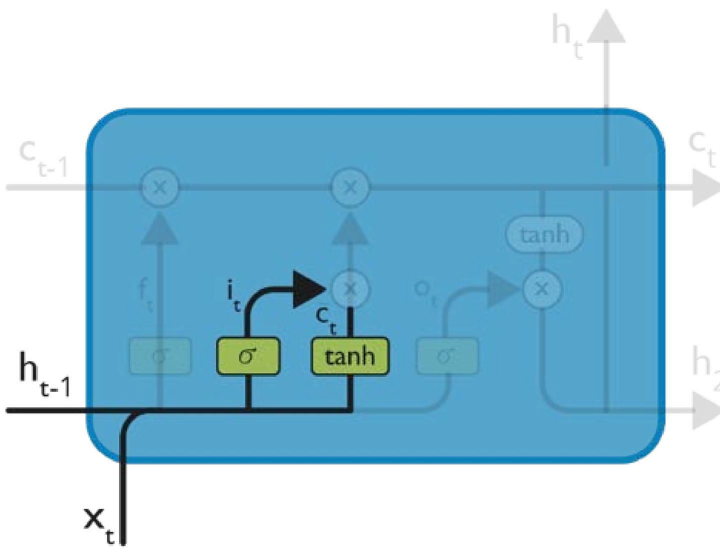


Figure 8. Input gate [4]

3. **<u>Updating</u>:** in this step, the old cell state c_{t-1} will be updated into new cell state c_t, in the beginning, $c_{t\text{-}1}$ multiplied by f_t to discard the values had been decided in the past step, then add the result of multiplying i_t by $C\tilde{}_t$, the result will be the new candidate values scaled by the amount that the decision to update each state value as shown in figure 9.

$$c_t = f_t \times c_{t-1} + i_t \times C\tilde{}_t \qquad \text{(Varsamopoulos, et al, 2018)}$$

[4] Images from DeepAI website by https://deepai.org/machine-learning-glossary-and-terms/recurrent-neural-network

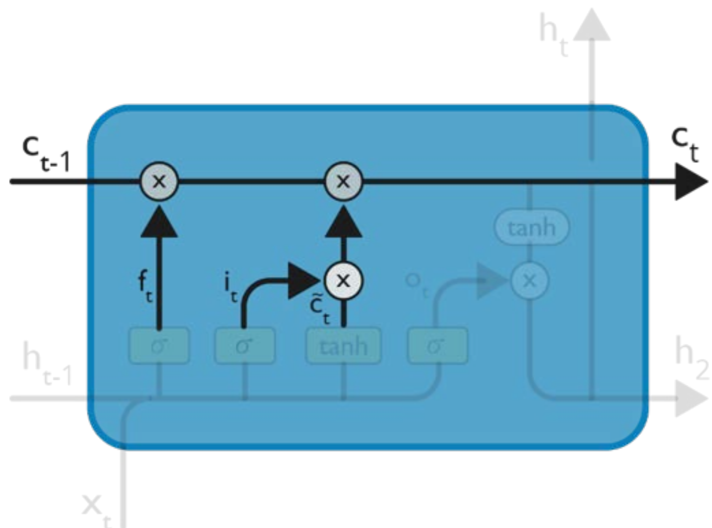


Figure 9. Update cell state process [4]

4. **<u>Selecting Output</u>:** Sigmoid layer decides which parts of the cell state that will be as the output. The cell state will be inserted in *tanh* (value is between -1 and 1), then it will be multiplied by the output of sigmoid gate because chosen parts will be included in the output (figure 10).

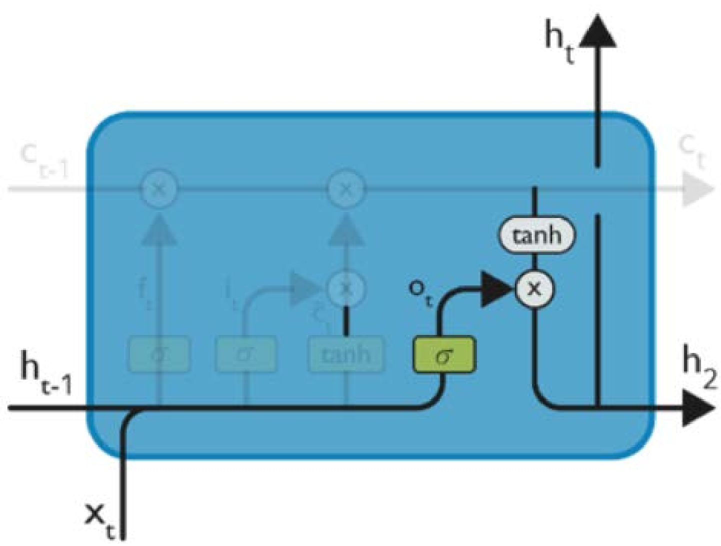


Figure 10. Output gate [4]

$$o_t = \sigma (w_o \, [h_{t-1} , x_t] + b_o) \qquad \text{(Varsamopoulos, et al, 2018)}$$

[4] Images from DeepAI website by https://deepai.org/machine-learning-glossary-and-terms/recurrent-neural-network

$$h_t = o_t \times tanh\ (\ c_t\ )$$ (Varsamopoulos, et al, 2018)

Then the output is complete.

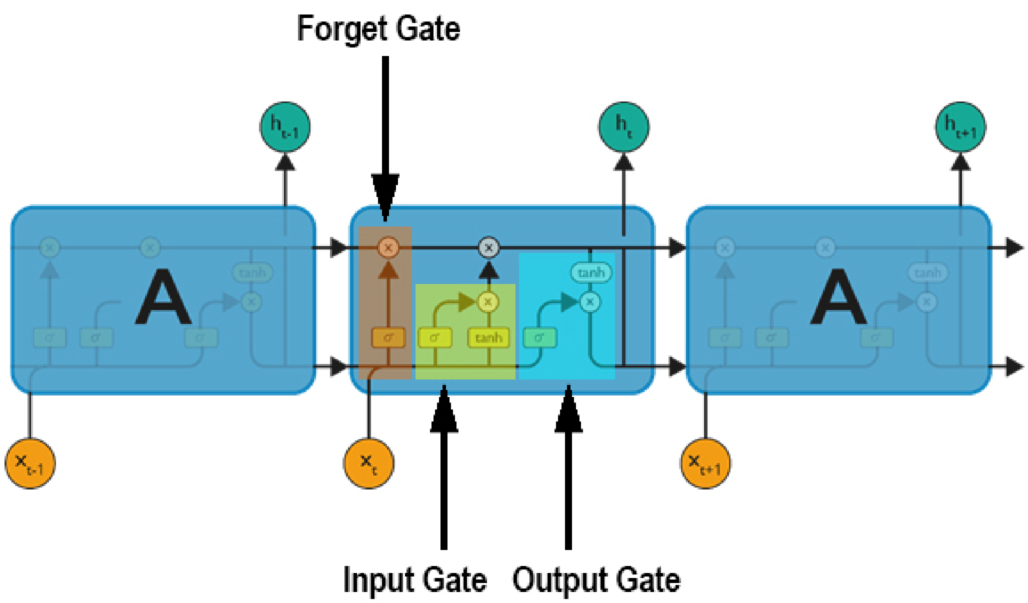


Figure 11. LSTM gates [4]

Summing up the four steps:

<u>1st step</u>: select which values to be discarded.

<u>2nd step</u>: chose what new inputs to be added to the network.

<u>3rd step</u>: combine the previously obtained inputs to generate the new cell states.

<u>4th step</u>: generate the output as per requirement.

[4] Images from DeepAI website by https://deepai.org/machine-learning-glossary-and-terms/recurrent-neural-network

Researchers used LSTM in stock pricing prediction, for example researchers in (Shah, et al, 2019) presented an LSTM model with organizations' stocks change figuring calculation to investigate just as expectation of future change of an organization in the Indian offer market. Paper in (Qiu, et al, 2020) utilizes consideration based LSTM and RNN with using wavelet change to process stock information. In paper (Zineb, et al, 2019) utilized LSTM to anticipate following day stock shutting value, utilizing two databases for day by day and yearly factors. In (Chung, et al, 2018) the proposed strategy utilizes uses LSTM & RNN with Genetic Algorithm (GA), GA was employed to search optimal or near-optimal value for the size of the time window and several LSTM units in an LSTM network.

2.3 Tabu List

Tabu list (TL) is apart from Tabu Search (TS) which considered as a metaheuristic search and it explores the solution space beyond local optimum by using TL, it’s size is an important tool to control the search in a short duration, given the purpose of an elective set of attributes for defining tabu status. There are many methods to define TL: fixed value, arbitrarily selected, or dynamically adjusted, TL is the set of specified number of solutions, the number of solutions determine the size of TL (Salhi, et al, 2002) (Ikkai, et al, 2005).

Tabu search interact with problems that it solves with finite solution space and it is a neighbourhood search method, it searches around its neighbours and there is flexible memory structure. To simplify TL objective, it keeps track of current solutions and putting them in its list, this means it prevents from choosing an option that has been chosen before in the past, which prevents going into cycles based on local maxima issue (Salhi, et al, 2002) (Shang, et al, 2019).

TS utilize TL to avoid making a move with same value or feature that has been selected before unless it leads to the ultimate solution or global maxima for the problem, this means that TS will keep searching for solution as long as the searching is being carried and it will return the best found solution in that period thanks to TL, in other words TL is implemented to avoid missing the movement of the best solution (Alhroob, et al, 2018) (Glover, 1990).

If TL is too long this means it would be costly to store choices and prevents solution searching process improvement, and if TL is too short this means the searching process cannot escape deep local minima. In conclusion, TL used to avoid cycling by forbidding revisiting moves (revisiting the same solution) and allowable memory to save several promising non-accepted neighbor solutions during the search process and use again when the search get trapped in local maxima (Alhroob, et al, 2018) (Józefowska**,** et al, 2002).

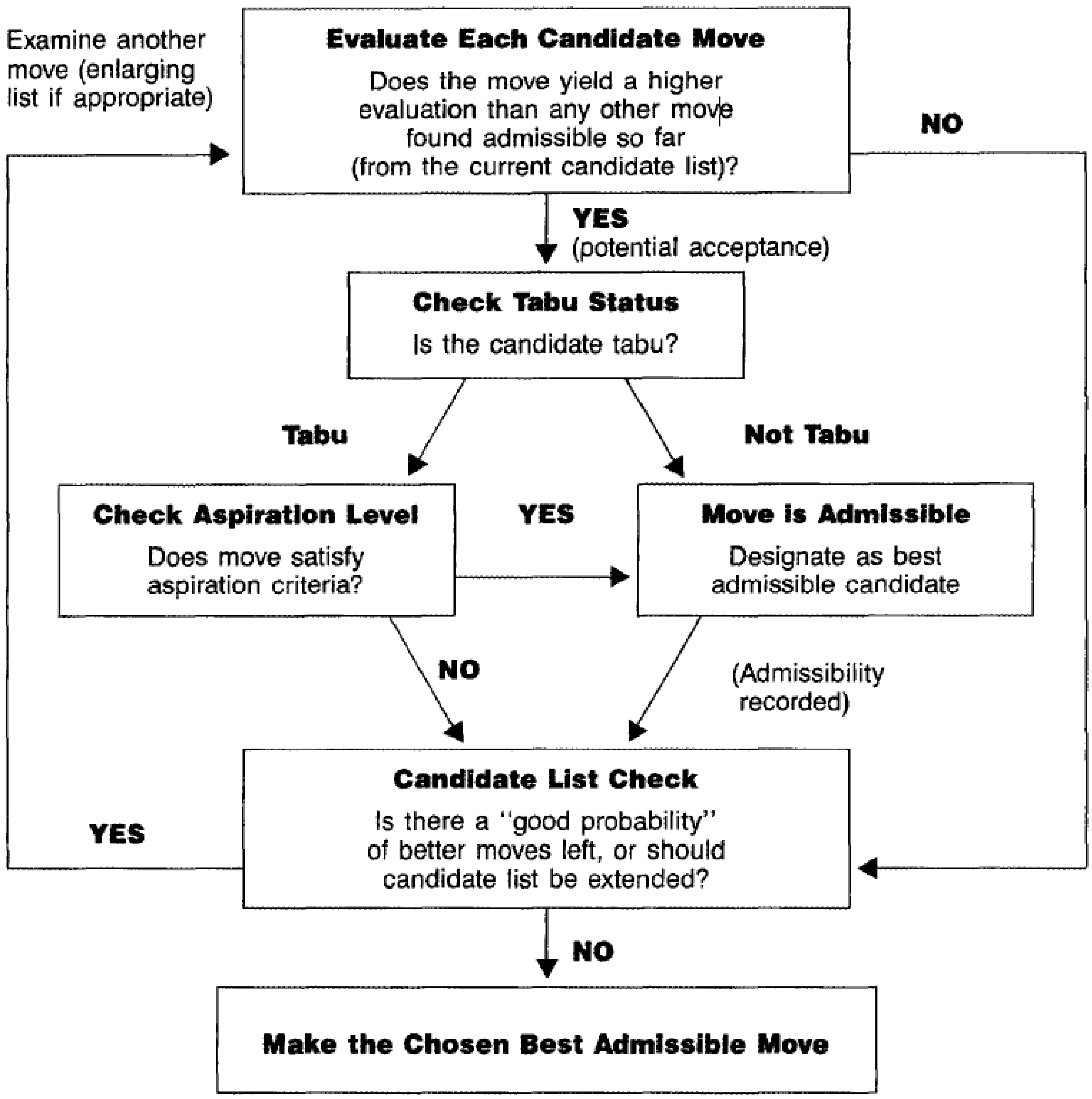


Figure 22. Tabu List flowchart [5]

2.4 Discussion

Many papers proposed methods for stock market prediction analysis by using RNN, Jahan, et al. (2018) developed a model by using RNN to forecast future stock prices based on the closing price of Advanced Micro Device (AMD) for 168 days, the error in predicted prices was less than five percent, but the model lacks the ability to predict hard changes in stock prices and prediction for different types of companies stock prices (Jahan, et al, 2018).

[5] Images from (Glover, 1990)

Xie, et al. (2017) proposed using RNN to predict stock market trends, the data taken from many companies in China Shanghai Shenzhen 300 Index. The results showed RNN can provide a reliable way to forecast future stock prices with decent accuracy, but the proposed method is too simple and needs more works to be robust regarding the accuracy of the prediction (Xie, et al., 2017).

Wei, et al. (2012) used hybrid RNN to predict Taiwan stock market in using four methods: (correlation matrix, stepwise regression, decision tree, and Elman NN) in the prediction process, the proposed method suffers from more conservative and noisy results (Wei, et al., 2012).

Khoa, et al. (2006) proposed using a model designed using both FFN (Feed Forward Neural Network) and simple RNN trained by time and profit-based backpropagation algorithm, the goal is to forecast S&P 500 (Standard and Poor 500) stock index which reflects the stocks prices of 500 largest companies in the U.S. one month ahead. Simple RNN with its 'time capture' abilities had better anticipated results than feed-forward neural network in all experiments. The model failed to predict the stock prices accurately and did not provide any considerable improvements toward the prediction process (Khoa**,** et al, 2006).

Hsieh, et al. (2011) developed a modified RNN with one of Swarm Intelligence Algorithms called ABC (Artificial Bee Colony Algorithm), the developed method is ABC-RNN that used to perform stock prices prediction, the test data taken from international stock markets including: Dow Jones Industrial Average Index (DJIA), London FTSE-100 Index (FTSE), Tokyo Nikkei-225 Index (Nikkei), and Taiwan Stock Exchange Capitalization Weighted Stock Index (TAIEX). The data processing using the WT (Wavelet Transform), then apply RNN to construct and chose input features, then apply ABC to optimize the RNN weights. The model considered being complex and lacking pattern processing ability (Hsieh**,** et al, 2011).

Jarrah, et al. (2019) proposed the use two systems to predict Saudi stock price trends, the first system addressed the issue of predicting the direction of motion of stock market index along with stock prices, the second system predicted future value of the stock market index. The research proposed Discrete Wavelet Transform (DWT) with RNN to create a simple method for prediction, this method suffers from unreliable results as a result of stock price spikes due to Islamic religious periods like Ramadan or Hajj (Jarrah, et al., 2019).

Zineb, et al. (2019) used LSTM to predict next-day stock closing price, using two databases for daily and annual variables, it's considered promising but in terms of refining noisy and bad predictions, it doesn’t contribute that much or suggests a way of improving it (Zineb, et al, 2019).

Alturki, et al. (2020) utilized Multivariate RNN with a LSTM layer to predict daily stock price for oil companies only in Saudi Arabia, the model forecast the next day stock price so the stocks broker will determine to buy or keep or sell the stock based on the result provided in research, the prediction occur based on historical oil prices of Brent crude oil with selected stock features. The method proposed in the research provided good results 55% accuracy and 23% investment gain, but the issue is that the method was not tested in the other business fields to demonstrate the accuracy, another issue is that the model predicts the stock price for a single day only and not for a specific period (Alturki, et al, 2020).

Moghar, et al. (2020) used RNN with a LSTM to predict open prices for Google and Nike stocks, the results were in terms of accuracy loss was low which means it is considered as promising solution but the proposed method suffers from the need for big stock prices dataset for 39 years (from 1980 to 2019) which is not available for most of the companies, another issue is the research lacks data length and number of training epochs that must be added to maximize the prediction accuracy (Moghar, et al, 2020).

Qiu, et al. (2020) used attention-based LSTM & RNN with utilizing wavelet transform to process stock data, the proposed model has a better fitting degree and improved accuracy of prediction results and provides more outcomes than other compared methods, but the impact of historical data on price trends is too singular and may not be able to fully and accurately forecast price on a given day (Qiu, et al, 2020).

Roondiwala,, et al. (2017) developed model by using RNN with LSTM to predict the stock prices for a group that consists of the 50 most active and liquid stocks in the Indian markets called NIFTY50, the developed LSTM model is composed of a sequential input layer followed by 2 LSTM layers and dense layer with ReLU activation and then finally a dense output layer with a linear activation function. The drawback of using the proposed model which is data must be normalized and filtered before being processed (Roondiwala, et al, 2017).

Chung, et al. (2018) proposed method uses LSTM & RNN with Genetic Algorithm (GA), GA was employed to search optimal or near-optimal value for the size of the time window and several LSTM units in an LSTM network, to consider temporal properties of the stock market, and utilized customized architectural factors of a model (Chung, et al, 2018).

Minami, et al. (2018) designed a sequential learning model for estimate a single stock price with company action information and “Macro-Economic” indices using RNN-LTSM method. The training data are taken from a single Japanese company, the model considers other variables besides stock price for prediction process, like company decisions, economic trends, and financial events into the mode mechanism to increase the prediction accuracy. The drawback of using this method is tailored to specific companies and resulted accuracies may vary from company to another, which means the designed model is not for general use (Minami, et al, 2018).

Shah, et al. (2019) introduced a modified LSTM model with companies' net growth calculation algorithm to analyze as well as prediction of future growth of a company in the Indian share market. The proposed model lacked the diversity of companies' business scopes and the number of companies' data was not that much, which leads to the question of prediction accuracy level with more companies' data involved (Shah, et al, 2019).

Selvin, et al. (2017) used three deep learning architectures RNN, LSTM, and CNN (Convolutional Neural Network) to predict stock prices, the data collected from three companies registered in NSE (National Stock Exchange of India) that operates in two sectors healthcare and IT for two periods of time. The model lacked the adaptability to stock market changes, and it's designed for specific markets and companies (Selvin, et al, 2017).

Generally, the previous researches suffered from the need for huge amount of training data and relatively massive computational power in order to achieve precise results. These shortcomings in previous works along with the need for highly accurate stock market prediction have been the motivation for this work to propose more suitable solutions that provide better prediction which will help corporations to make the right decision in the future for their business.

Our proposed model starts with acquiring a financial information database for a Jordanian companies, then RNN stars monitoring stock price ups and downs in the daily period represented by selected features, after a specified period for example 100 days, the learning process will be complete and prediction starts, method predict the stock price, for example, day 101 and continue for the preferred period and keeps updating depending on the new stock price changes.

Then take average last 5 predictions and calculate the average (**X**) of them and compare it with the real stock price of that day to check the difference between produced prediction and stock price, if the difference is high it means there is an issue with the proposed method and call this type of data as missing or not good.

To save predictions that are far away from real stock price in another list so it won't affect other predictions, and perform another prediction iteration and then a new prediction produced **(Y)**, then compare Y with X, if the gap is small will take Y and delete X value, if not will discard Y value and keep X, this process aims for improving the accuracy of prediction.

We proposed to tune the prediction during unexpected events change stock price dramatically so it will not affect method precision, add a tabu list memory to save the last 5 predictions in the queue, then optimize prediction values to improve prediction accuracy.

The goal is to provide a reliable prediction for company performance and business in the future to help companies in Jordan to survive in the market.

Chapter 3

Methodology

3.1 Problem Description

As mentioned earlier, predicting the stock market for Jordanian companies is an essential challenge, especially with Jordanian market conditions change, this leads many companies to search for a more effective way to predict stock market than traditional methods, these help companies to use such technologies to survive & grow in the market and avoid mistakes that can lead to stock price drop.

3.2 Financial Data Collection

The financial data was collected from Amman Stock Exchange (ASE), which is a non-profit independent institution authorized to function as a regulated market for trading securities in Jordan, through their official website www.exchange.jo , for the period between 2015 to 2019.

The data contains daily values for the selected features (Date, Open, High, Low, Close, Volume) for each Jordanian company registered in the ASE, collected data sample shown in figure 13.

trade_date	name	code	symbol	volume	high	low	open	close_price
02-01-19	البنك الإسلامي الأردني	111001	JOIB	9526.29	2.88	2.87	2.88	2.87
02-01-19	البنك الاردني الكويتي	111002	JOKB	6740.04	2.88	2.86	2.86	2.87
02-01-19	البنك التجاري الأردني	111003	JCBK	193.6	0.88	0.88	0.88	0.88
02-01-19	بنك الاسكان للتجارة والتمويل	111004	THBK	40468.95	8.37	8.37	8.37	8.37
02-01-19	بنك الاستثمار العربي الاردني	111005	AJIB	25671	1.29	1.29	1.29	1.29
02-01-19	بنك صفوة الإسلامي	111006	SIBK	1378	1.15	1.14	1.14	1.15
02-01-19	بنك الإتحاد	111007	UBSI	1120	1.6	1.6	1.6	1.6
02-01-19	البنك الاستثماري	111014	INVB	2640	1.32	1.32	1.32	1.32
02-01-19	بنك المال الأردني	111017	EXFB	14732.6	0.94	0.92	0.92	0.94
02-01-19	بنك القاهرة عمان	111021	CABK	18449.6	1.34	1.33	1.33	1.34
02-01-19	بنك الاردن	111022	BOJX	12297.62	2.47	2.42	2.47	2.42
02-01-19	البنك الاهلي الاردني	111033	AHLI	9934.95	1.09	1.05	1.05	1.09
02-01-19	البنك العربي	113023	ARBK	48980.16	6.19	6.13	6.19	6.15
02-01-19	الشرق الأوسط للتأمين	121002	MEIN	1178.8	1.45	1.35	1.35	1.45
02-01-19	التأمين العربية - الأردن	121005	AICJ	21.6	0.54	0.54	0.54	0.54

Figure 13. Sample of the collected stock prices

3.3 Data Preparation

The data is acquired from the historical business performance database of companies in Jordan, then the data will be categorized and analyzed to select the features that the RNN with LSTM and Tabu List utilize to provide a solution to the problem.

The proposed features (as shown in table 1) are selected according to the problem domain of this research, these features will provide the input for the proposed method to tune and enhance the prediction results accuracy and offer a better solution than other used methods.

Table 1 Selected Features

Selected feature	Description
Date	The date for the stock transaction
Open	The stock price at the beginning of the day
High	The maximum value the stock price has been achieved at a specific date
Low	The minimum value the stock price has been achieved at a specific date
Close	The stock price at the end of the day
Volume	Number of transactions have been done at that date

3.4 Programming Tools

The proposed method of RNN with LSTM and Tabu List was built using Keras which is Python deep learning API with the help of TensorFlow2 to optimize the code and the built in functions for reading stocks' prices datasets and use graph drawing function to get the difference between the real stock price with stock price predicted by RNN with LSTM and by RNN with LSTM and Tabu List.

3.5 Proposed Methodology

The proposed method follows the following steps:

1) Acquire company financial performance datasets from Amman Stock Exchange
2) Analyse the collected dataset which has been obtained from the company's financial performance
3) Stock price features are selected, and data transferred to testing data sets
4) Analyse the selected features
5) RNN start processing the extracted features and begin the prediction process, then send the stocks features to LSTM
6) LSTM start the learning process based on the processed data received from RNN
7) LSTM initiate the training phase
8) LSTM start the test phase
9) Tabu List checks the predicted prices with the real stock prices
10) Tabu List removes unmatched predicted prices to enhance the prediction accuracy by eliminating the far from the real stock price and redundant predictions, inform the LSTM with the unmatched results, so in case that the predicted stock prices were far away from the real stock prices, LSTM have to redo the training and testing phase
11) Models are trained and tested several times using different model parameters to find the most suitable parameter value

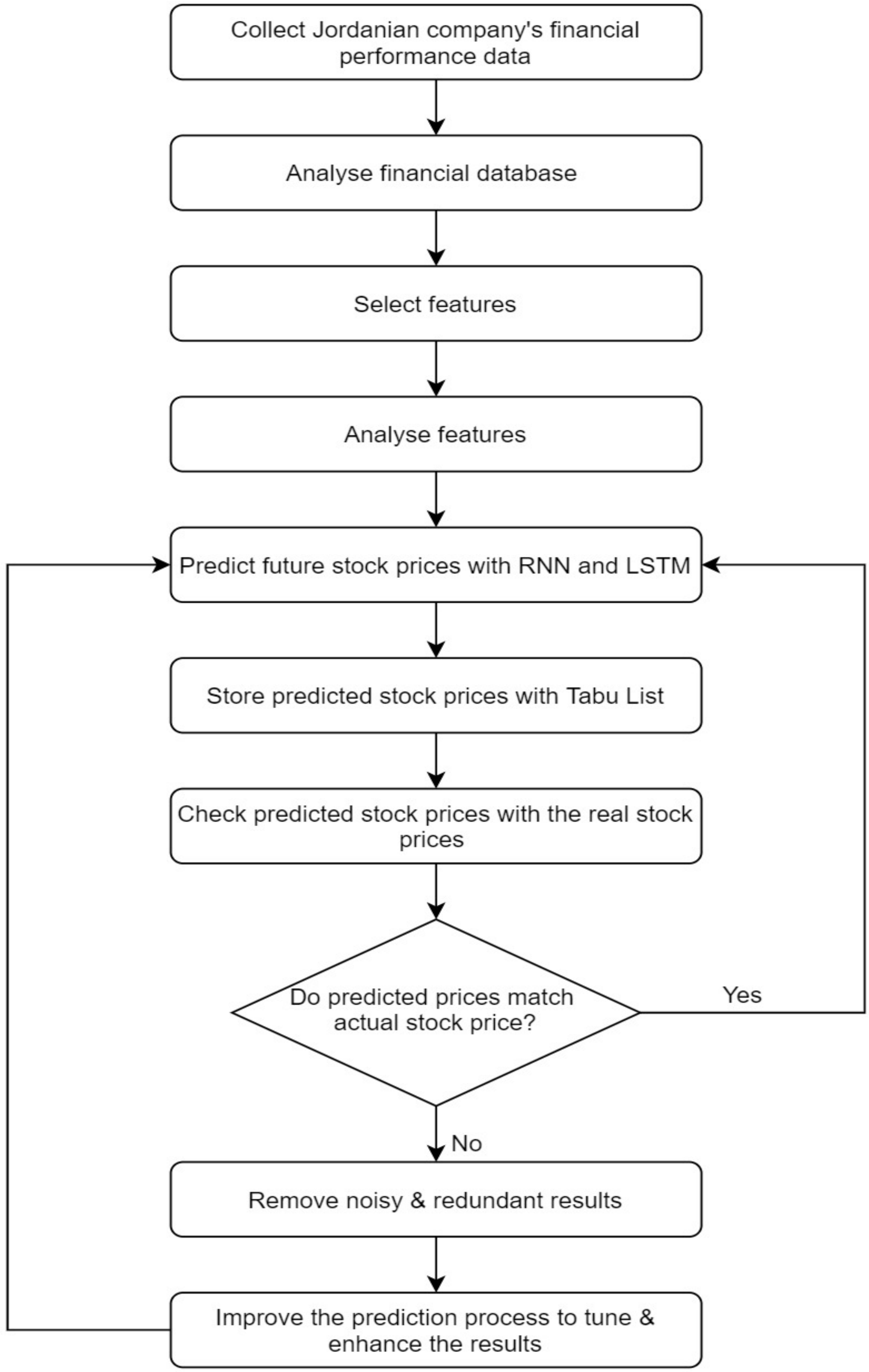


Figure 14. The Proposed Hybrid RNN with Long Term Short Memory with Tabu List Memory Model

3.6 Detailed Workflow

The method we have proposed to use RNN with LSTM and Tabu list to predict the stock prices for companies in Jordan.

The steps as follows:

- Loading the financial records that we have acquired and chose a certain number of days then the program will load the financial data according for the choses period, the loaded data will be used to extract the selected features (figure. 15).

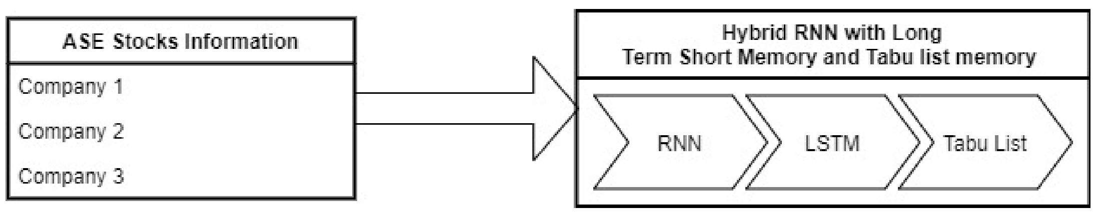


Figure 15. Stock data loading process into the model

- Specify the range of data that will be used to train RNN process, so for example we chose the financial data from 01-Jan-2015 to 31-Dec-2018 for training.
- specify the range of data that will be used to test the accuracy of the primary results of the method, so for example we select the financial data from 01-Jan-2019 to 31-Dec-2019 for testing.
- Extract the selected the features that monitor the stock prices daily and assign them to the variables used in the training and testing modules (illustrated in figure. 16).

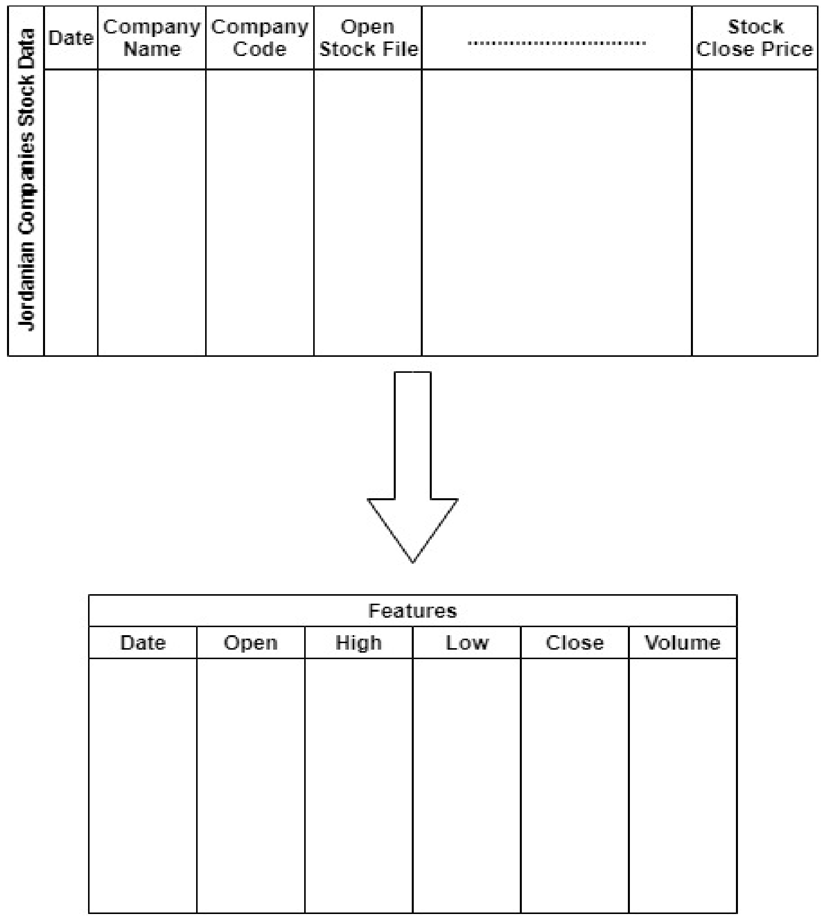


Figure 16. Features extraction

- Normalize the values of training variables to numbers between 0 and 1.
- Start the training process for the preferred amount of time (for example: 60 days, 140 days, 240 days. 360 days).

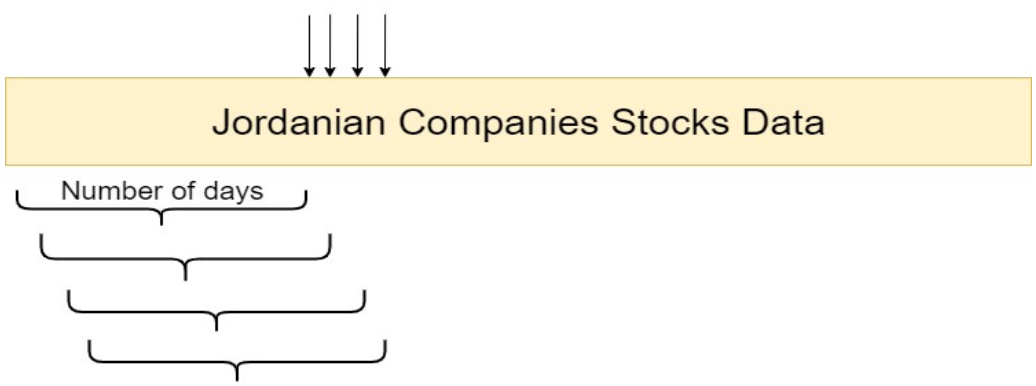


Figure 17. Training data selection

- Fit the training results into matrices to be used as an input for the LSTM process.

Figure 18. Training results fitting

- Activate the LSTM layer by training the results acquired in the past step for a specified number of epochs.
- Perform testing for the results obtained from the training step.
- Apply Adam optimizer algorithm
- Start the stock price prediction process for the desired period.
- Apply Tabu list method to improve the accuracy of the results of the prediction process and inform the LSTM if the predictions are getting far from the real stock prices to restart the training phases for the purpose of increasing the accuracy.
- Visualize the results and show them to visualize the difference between them and the actual stock prices.

3.7 Detailed Tabu List Workflow

Traditional Tabu List contains the weights of next possible solutions, we modified it to store the stock prices predictions for specified date, this process is to ensure that predictions that are far away from real stock prices and redundant predictions so they won't affect the prediction accuracy.

TL procedures are mentioned in the following:

1. Store LSTM initial stock price predictions (*P*) for certain number of times for stock price (S_i) in certain date *i*, the predictions store in the Tabu list.

   For example: Jordan Petroleum Refinery Co (JoPetrol) on 27-Mar-2019 stock price was 3.07, LSTM stock predictions are (3.35, 3.22, 2.84, 3.05, 3.22) for the same date.

P_1	P_2	P_3	P_4	P_5
3.35	3.22	2.84	3.05	3.22

2. Remove redundant predictions so it will not affect the overall prediction accuracy, redundant removal process occurs by checking each prediction value with other values and eliminate the redundant value, so this will stop the accuracy getting affected by redundant predictions.

P_1	P_2	P_3	P_4	P_5
3.35	3.22	2.84	3.05	3.22

   Then the resulted predictions are:

P_1	P_2	P_3	P_4	P_5
3.35	3.22	2.84	3.05	–

3. Check stored predictions values individually $\boldsymbol{P_x}$ with real stock price $\boldsymbol{S_i}$ (3.07) to identify if the difference the accuracy $\boldsymbol{A_x}$ between prediction and real stock prices is big or small using the following equation: $\boldsymbol{A_x} = \left| \frac{(P_x - S_i)}{100} \right|$

   In the example: TL check the value of P_1 (3.35) with the value of S_i (3.07) to determine the if the prediction P is precise or not, using the equation mentioned above:

$$\boldsymbol{A_1} = \left| \frac{(P_1 - S_i)}{100} \right| = \left| \frac{(3.35 - 3.07)}{100} \right| = \left| \frac{0.28}{100} \right| = 0.0028$$

   Then check the next prediction P_{x+1} with S_i to find accuracy $A_{x+2,}$ so the process as follows:

$$\boldsymbol{A_2} = \left| \frac{(P_2 - S_i)}{100} \right| = \left| \frac{(3.22 - 3.07)}{100} \right| = \left| \frac{0.15}{100} \right| = 0.0015$$

   And the rest of the accuracy values as follows:

$$\boldsymbol{A_3} = \left| \frac{(P_3 - S_i)}{100} \right| = \left| \frac{(2.84 - 3.07)}{100} \right| = \left| \frac{-0.23}{100} \right| = 0.0023$$

$$\boldsymbol{A_4} = \left| \frac{(P_4 - S_i)}{100} \right| = \left| \frac{(3.05 - 3.07)}{100} \right| = \left| \frac{-0.02}{100} \right| = 0.0002$$

   The prediction accuracy process continues until all the predicted stock prices checked individually,

   The result accuracy A_x as follows:

A_1	A_2	A_3	A_4	A_5
0.0028	0.0015	0.0023	0.0002	–

4. TL checks A_x for the prediction with the best accuracy value (smallest prediction) and take the remaining predictions that have accuracy equal or less than 25% to be checked with another list T that contain predictions for previous days, this process would be used to check and refine future predictions to tune the accuracy of LSTM, in other words this would measure if the overall accuracy are getting better or worst so LSTM can take the feedback.

Choosing the smallest value for accuracy (which is under less or equal to 25%):

A_1	A_2	A_3	A_4	A_5
0.0028	0.0015	0.0023	0.0002	–

So, the predicted price for JOPT stock on 22-Mar-2020 is 3.05 (P_4 value).

Choosing accuracy values that are less or equal to 25%:

A_1	A_2	A_3	A_4	A_5
~~0.0028~~	0.0015	0.0023	–	–

The A_2 value 0.0015 and A_3 value 0.0023 will be added to list T, it will be used to monitor the overall accuracy periodically and if the accuracy is getting higher and not smaller, Tabu List will inform the LSTM to redo the training again to improve the prediction process, this step is ensure there is a feedback to LSTM to make sure that LSTM strive to reach the best possible accuracy resulted from training and testing phases.

3.8 RNN with LSM and Tabu List Pseudocode

```
Load dataset (companies' stocks data)
        Read stocks data features (Date, Open, High, Low, Close, Volume)
Preprocessing and Feature Extraction
        Format Data for preprocessing and feature extraction
        Separate dataset into training set
```

```
        Put the remaining dataset to testing set
        Scale training data between 0 and 1
        Scale testing data between 0 and 1
        Create training dataset array
        Create testing dataset array
        Reshape Training dataset to LSTM layer
        Reshape Testing dataset to LSTM layer
    Apply LSTM model
        Repeat Until Reaching the desired number of epochs
            Start
                Add LSTM layer
                Define return sequence number
                Define input
            End
        Compile LSTM model
        Apply Adam optimizer algorithm
        Save the output to train dataset array
    Apply Tabu List on the preliminary results
    Filter the results
```

3.9 Evaluation Phase

This phase concentrates to improve the initial solution producing by the construction phase, by minimizing the accuracy. To achieve this target using RNN with new mechanisms (e.g. Tabu list and allowable memory). we compared and evaluated the implemented model performance with the original stock data benchmark and new real-world data.

The results of the evaluation process started by running the proposed model using ASE stocks data as an input and checking the difference between the real stocks' prices with the predicted prices to check the difference and tune the model to produce prices that are close to real stock prices.

The evaluation done in two phases:

I. We are going to compare the accuracy of the stocks prices predicted using RNN with LSTM and the accuracy RNN with LSTM and Tabu list, to check the impact of applying Tabu list as an addition to RNN and LSTM model.

II. Use the evaluation metrics used in Jarrah, et al., (2019) published research, they used Mean Squared Error (MSE), Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) criteria to check the overall accuracy of their stock pricing model (Jarrah, et al., 2019). Then we will compare resulted MSE, MAE, and RMSE from using RNN with LSTM and TL with their results generated from using DWT and RNN.

Chapter 4

Experimental Results and Discussion

This chapter presents detailed experimental results of the proposed approach for hybrid model that consists of using RNN with LSTM and Tabu list for stock prices prediction. The data range was between 1st Jan 2015 to 31th Dec 2019 to ensure that the model can provide better results and high accuracy and compare the results by using the conventional RNN and LSTM model and RNN with LSTM and Tabu list model proposed in this thesis. The stocks prices data was separated into training and testing data sets based on 80% for training and 20% for testing, the rates chosen to give more information to RNN & LSTM training phase so the resulted accuracy would be better because most of the dataset was used in the training phase as shown in (Moghar, et al, 2020). This ratio was implemented on all the mentioned results in this chapter.

We divided the companies' stocks prices data sets into two groups based on the available duration, the separation in order to check if our model can predict, big part of the acquired stock prices database from ASE website are for companies that issued their stocks before or since 2015 and other part are for companies that issued their stocks after 2017 year so they have shorter dataset than the first group. Dataset division is done to prove that our proposed model produces results with high accuracy disregard the companies' stocks dataset size.

4.1 Results for companies that issued their stocks before or since 2015

First company from first group that was used for stock prediction model we have developed is Cairo Amman Bank (CABK), figure 19 shows the real predicted prices for a period of time in year 2019 with comparison to predicted prices resulted from applying RNN with LSTM, and applying RNN with LSTM and TL.

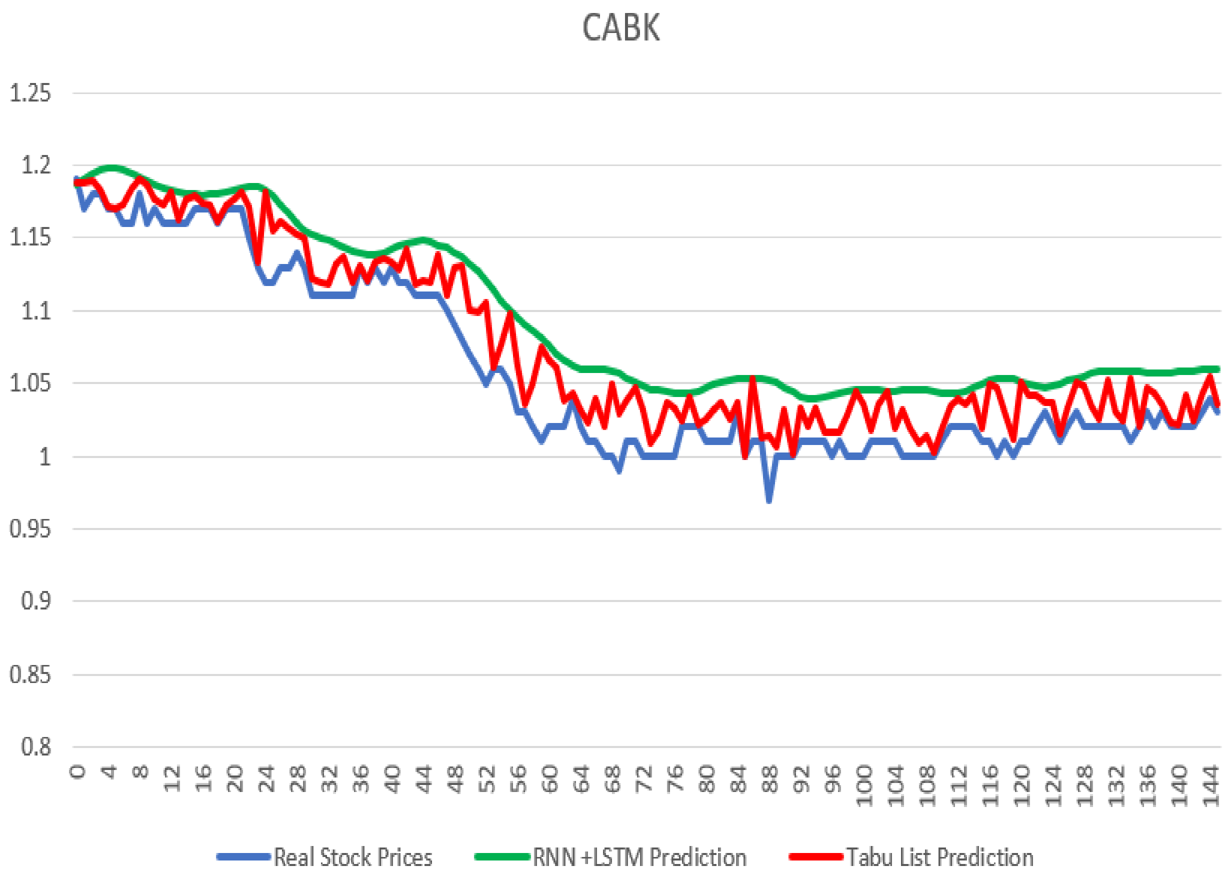


Figure 19. Cairo Amman Bank (CABK) predicted stock prices

Another company was used for stock prediction model we have developed is Jordan Petroleum Refinery Co (JOPT), figure 20 shows the real predicted prices for a period of time in year 2019 with comparison to predicted prices resulted from applying RNN with LSTM, and applying RNN with LSTM and TL.

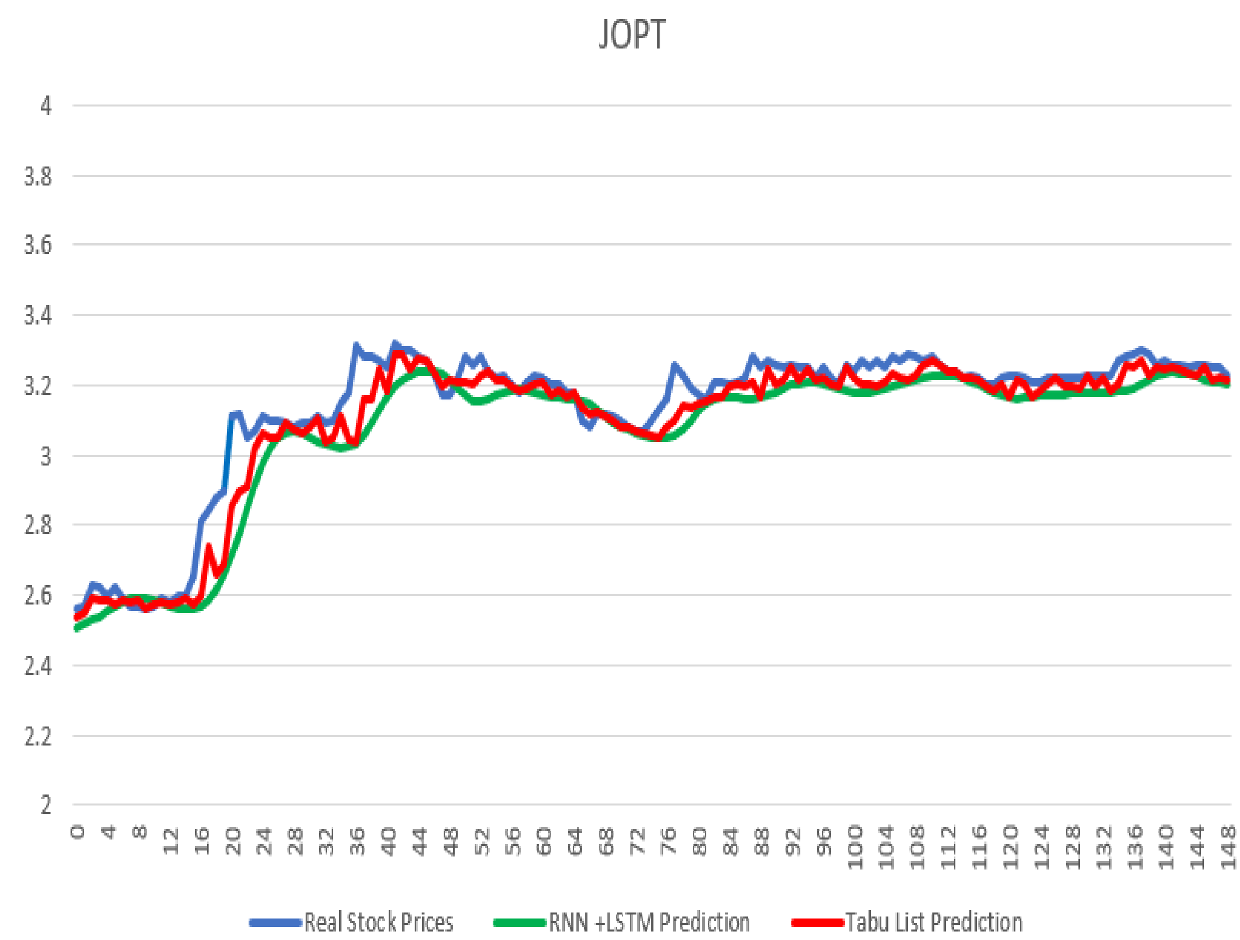


Figure 20. Jordan Petroleum Refinery Co (JOPT) predicted stock prices

Another company was used for stock prediction model we have developed is Jordanian Real Estate Company for Development (JRCD), figure 21 shows the real predicted prices for a period of time in year 2019 with comparison to predicted prices resulted from applying RNN with LSTM, and applying RNN with LSTM and TL.

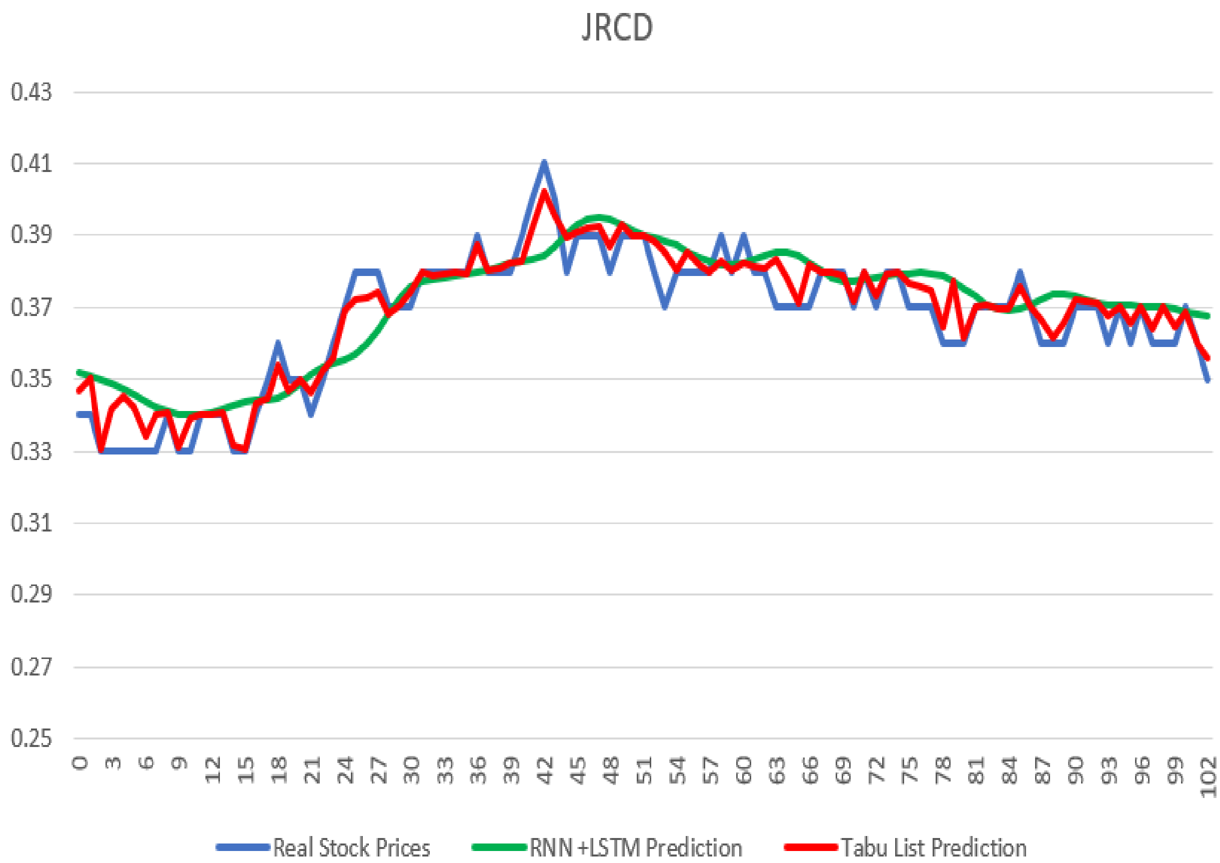


Figure 21. Jordan Petroleum Refinery Co (JOPT) predicted stock prices

Another company was used for stock prediction model we have developed is Capital Bank of Jordan (EXFB), figure 22 shows the real predicted prices for a period of time in year 2019 with comparison to predicted prices resulted from applying RNN with LSTM, and applying RNN with LSTM and TL.

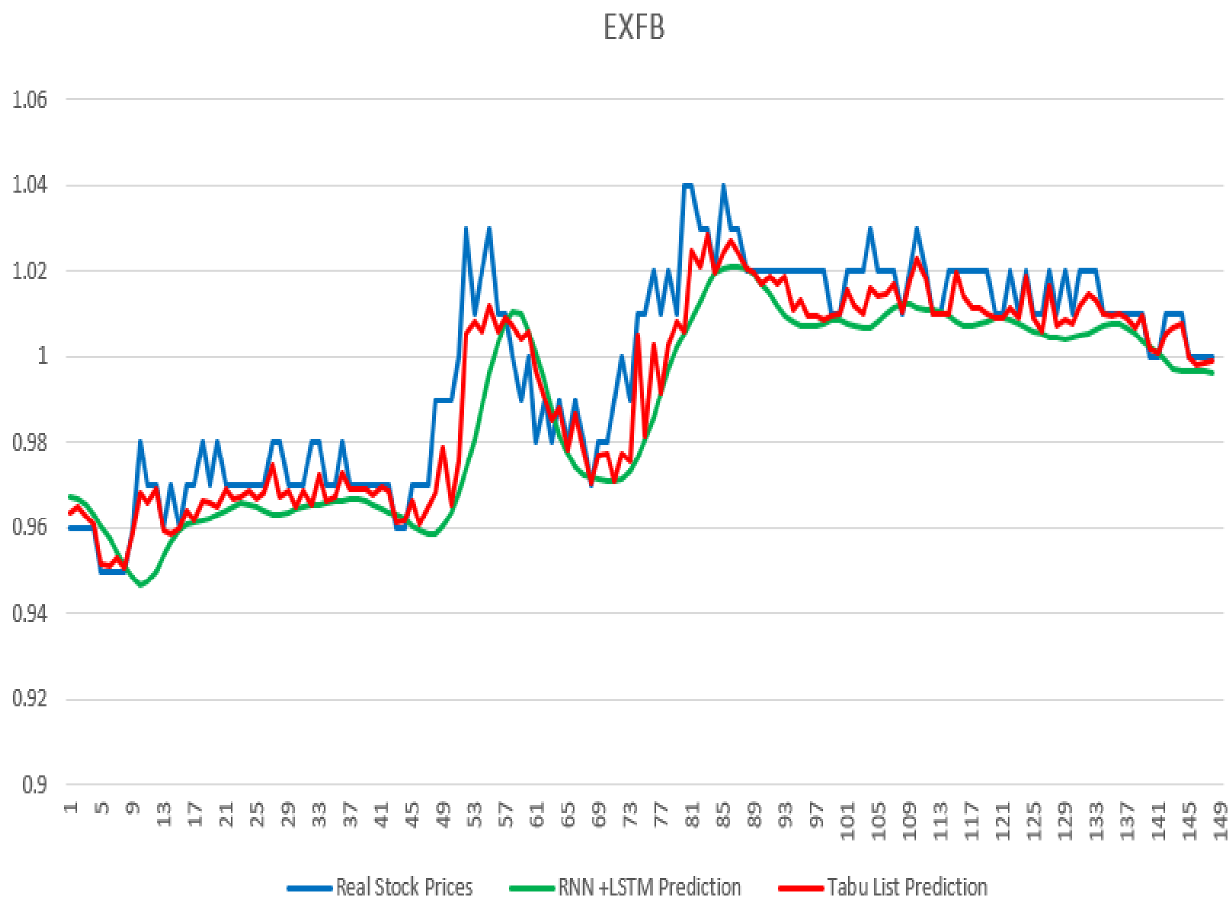


Figure 22. Capital Bank of Jordan (EXFB) predicted stock prices

Another company was used for stock prediction model we have developed is Jordan Telecommunications (JTEL), figure 23 shows the real predicted prices for a period of time in year 2019 with comparison to predicted prices resulted from applying RNN with LSTM, and applying RNN with LSTM and TL.

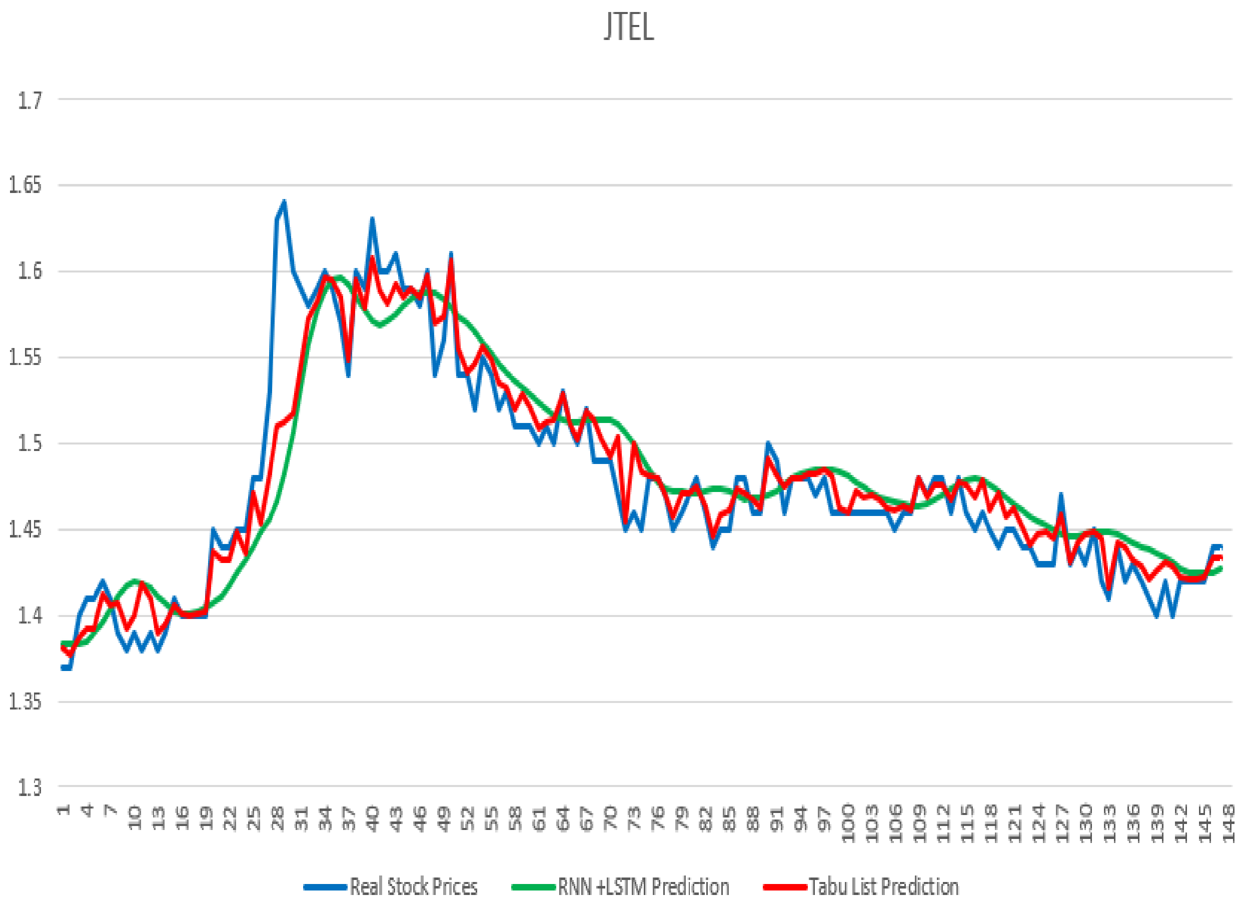


Figure 23. Jordan Telecommunications (JTEL) predicted stock prices

Another company was used for stock prediction model we have developed is Jordan Phosphate Mines Co PLC (JOPH), figure 24 shows the real predicted prices for a period of time in year 2019 with comparison to predicted prices resulted from applying RNN with LSTM, and applying RNN with LSTM and TL.

Figure 24. Jordan Phosphate Mines Co PLC (JOPH) predicted stock prices

Another company was used for stock prediction model we have developed is Rum Group for Transportation and Tourism Investment (RUMM), figure 25 shows the real predicted prices for a period of time in year 2019 with comparison to predicted prices resulted from applying RNN with LSTM, and applying RNN with LSTM and TL.

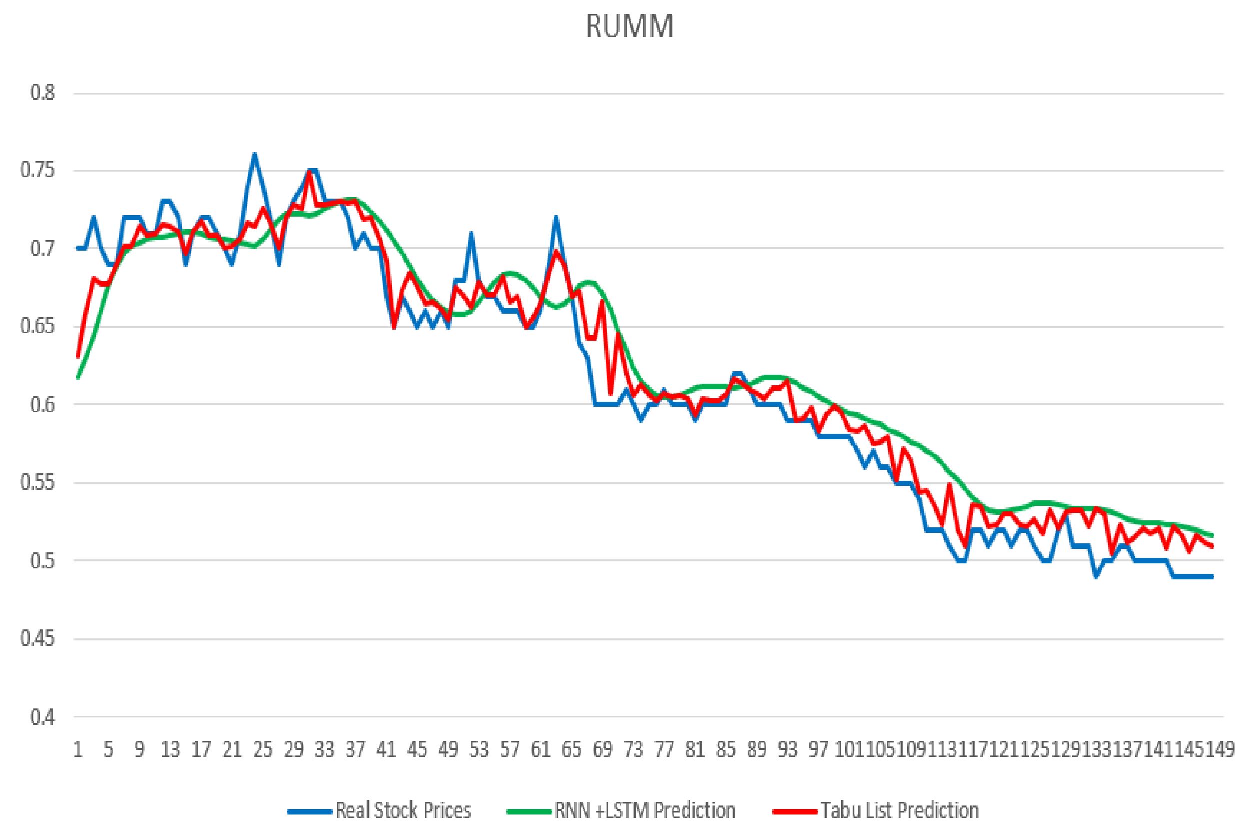


Figure 25. Rum Group for Transportation and Tourism Investment (RUMM) predicted stock prices

Another company was used for stock prediction model we have developed is Jordan Electric Power Co (JOEP), figure 26 shows the real predicted prices for a period of time in year 2019 with comparison to predicted prices resulted from applying RNN with LSTM, and applying RNN with LSTM and TL.

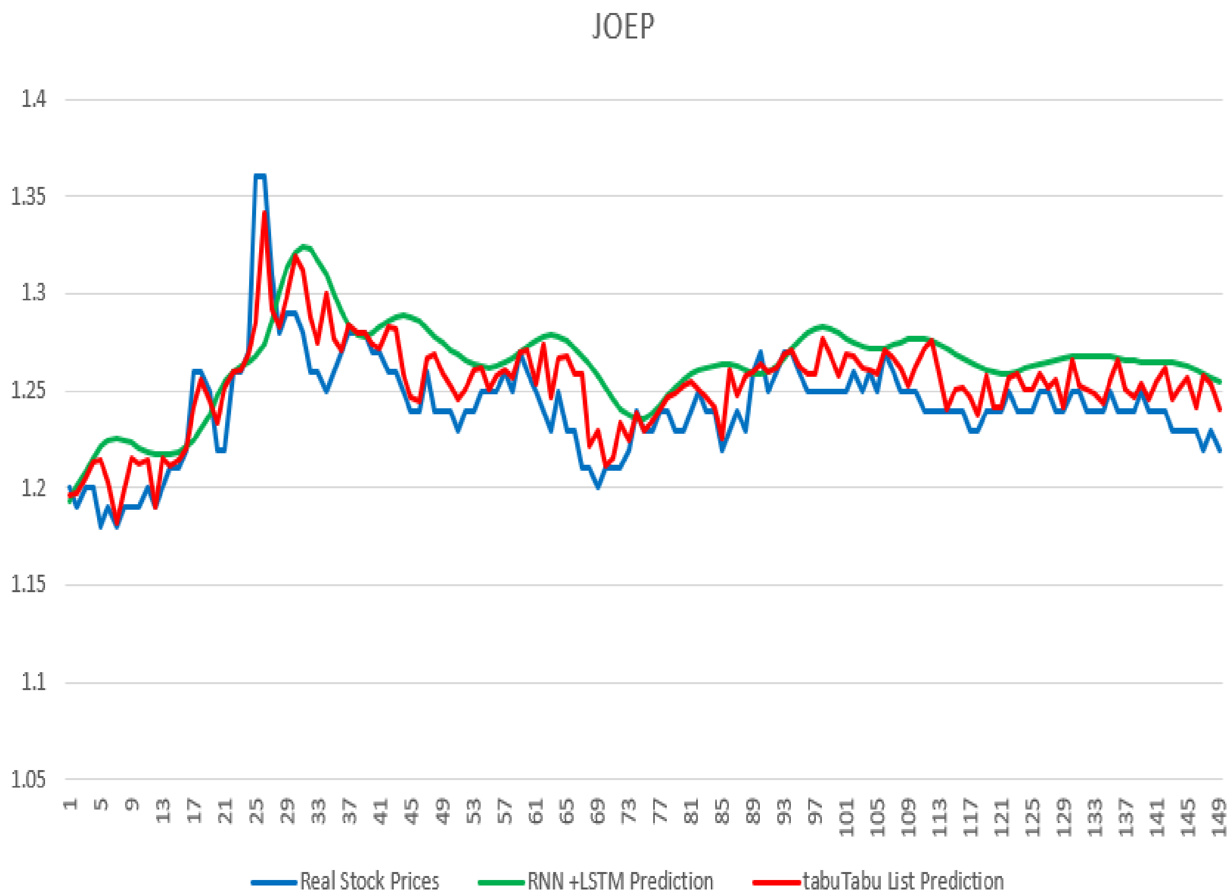


Figure 26. Jordan Electric Power Co (JOEP) predicted stock prices

Table 2 Results for companies that issued their stocks before or since 2015

	Company Name	Real Stock Price	RNN with LSTM Prediction	RNN with LSTM and Tabu List Prediction
1	Cairo Amman Bank (CABK)	1.058082192	1.094912013	1.076618623
2	Jordan Petroleum Refinery Co (JOPT)	3.134966443	3.070329097	3.098166262
3	Jordanian Real Estate Company for Development (JRCD)	0.367087379	0.370539131	0.369058117
4	Capital Bank of Jordan (EXFB)	0.997432432	0.988067924	0.992512
5	Jordan Telecommunications (JTEL)	1.476054422	1.479648586	1.477888605
6	Jordan Phosphate Mines Co PLC (JOPH)	3.287114094	3.234010833	3.26376747
7	Rum Group for Transportation and Tourism Investment (RUMM)	0.616418919	0.628192842	0.622577723
8	Jordan Electric Power Co (JOEP)	1.242348993	1.263509919	1.252996248

4.2 Results for companies that issued their stocks since or after 2017

Second group of companies that are issued their stocks in the past three years which means we have smaller dataset to train and test compared to the first groups of companies, the stock prices from 2017 till the end of 2019 and the predicted prices for a period of time in year 2019.

First company from the second group that was used for stock prediction model we have developed is Investment House for Financial Services (INVH), figure 27 shows the real predicted prices for a period of time in year 2019 with comparison to predicted prices resulted from applying RNN with LSTM, and applying RNN with LSTM and TL.

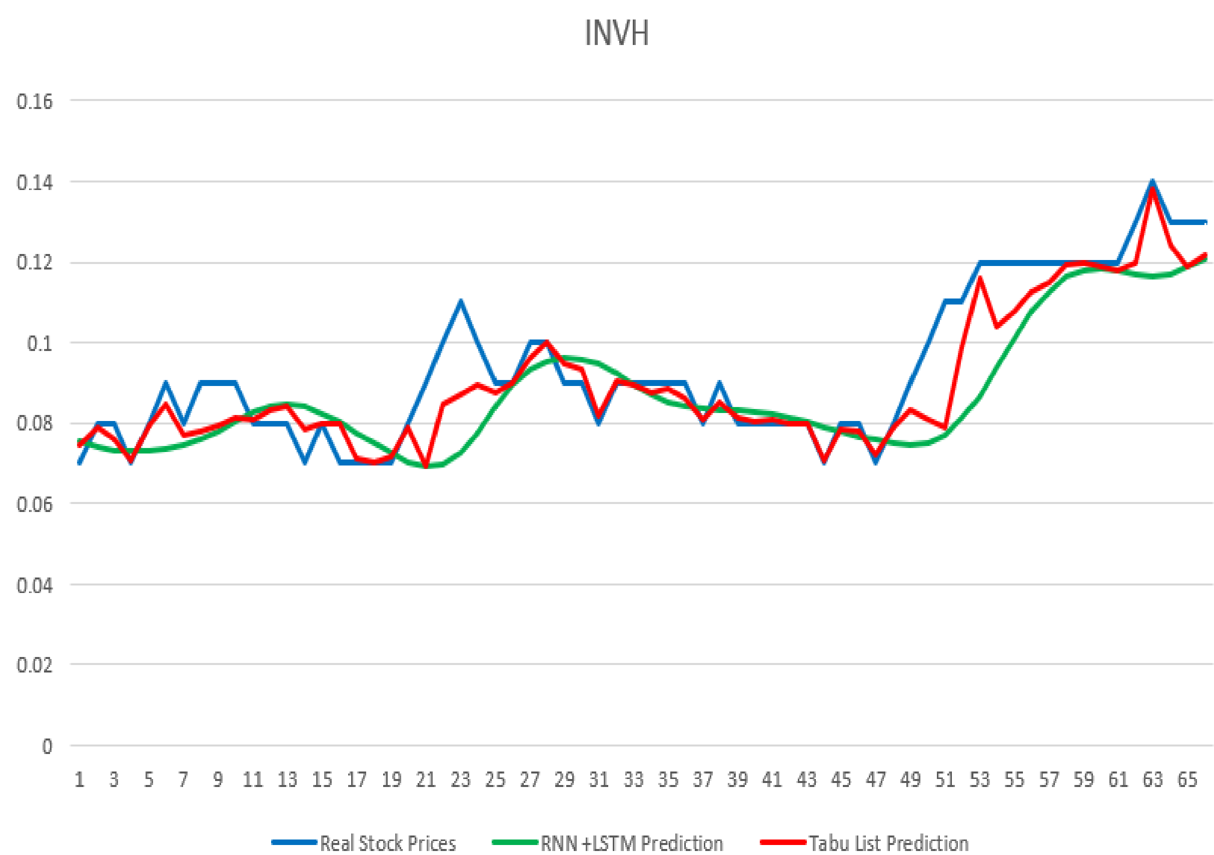


Figure 27. Investment House for Financial Services (INVH) predicted stock prices

Another company that was used for stock prediction model we have developed is Investment Arab East for Real Estate Investments Co (REAL), figure 28 shows the real predicted prices for a period of time in year 2019 with comparison to predicted prices resulted from applying RNN with LSTM, and applying RNN with LSTM and TL.

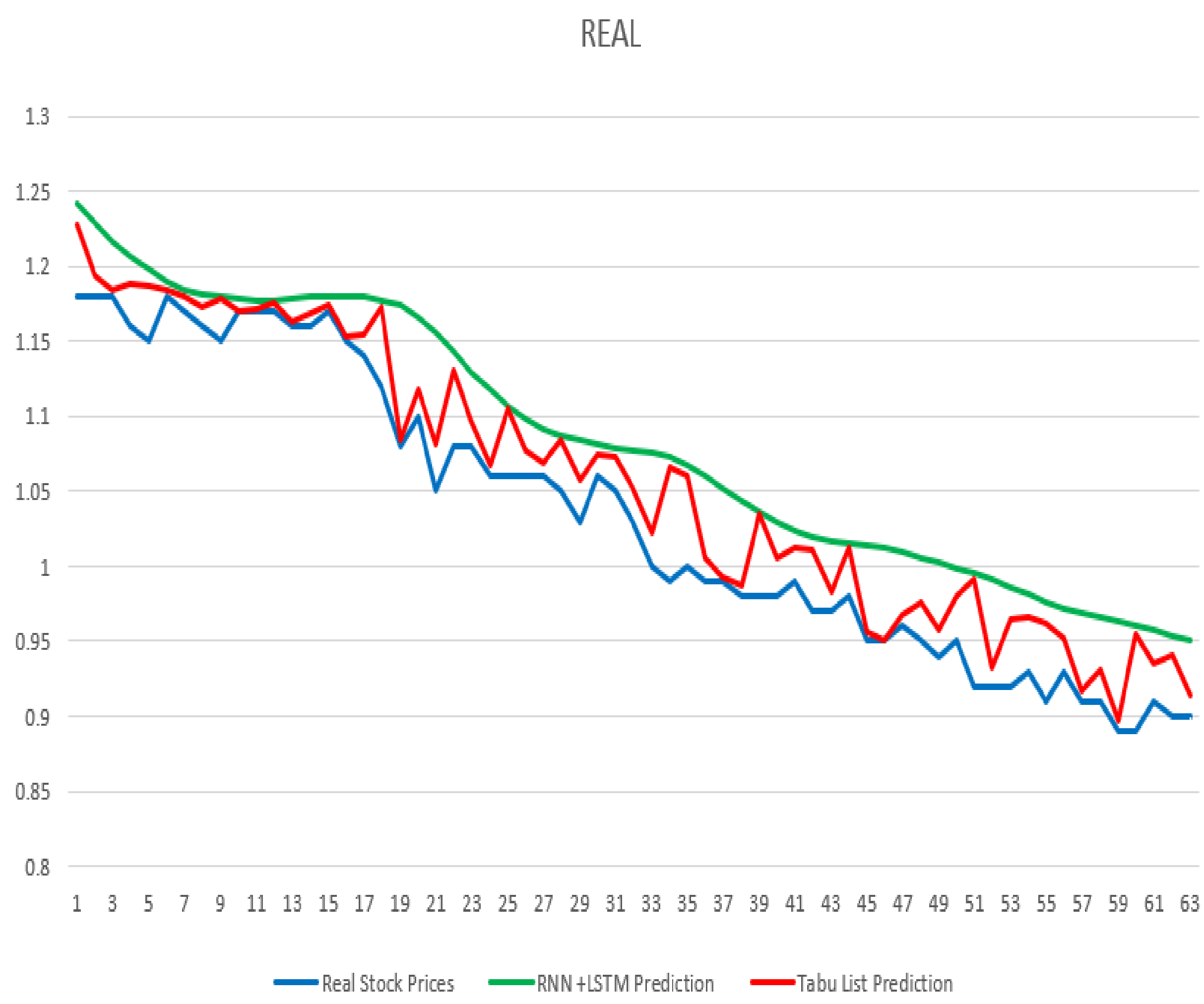


Figure 28. Investment Arab East for Real Estate Investments Co (REAL) predicted stock prices

Another company that was used for stock prediction model we have developed is AlIsraa for Islamic Finance and Investment (ISRA), figure 29 shows the real predicted prices for a period of time in year 2019 with comparison to predicted prices resulted from applying RNN with LSTM, and applying RNN with LSTM and TL.

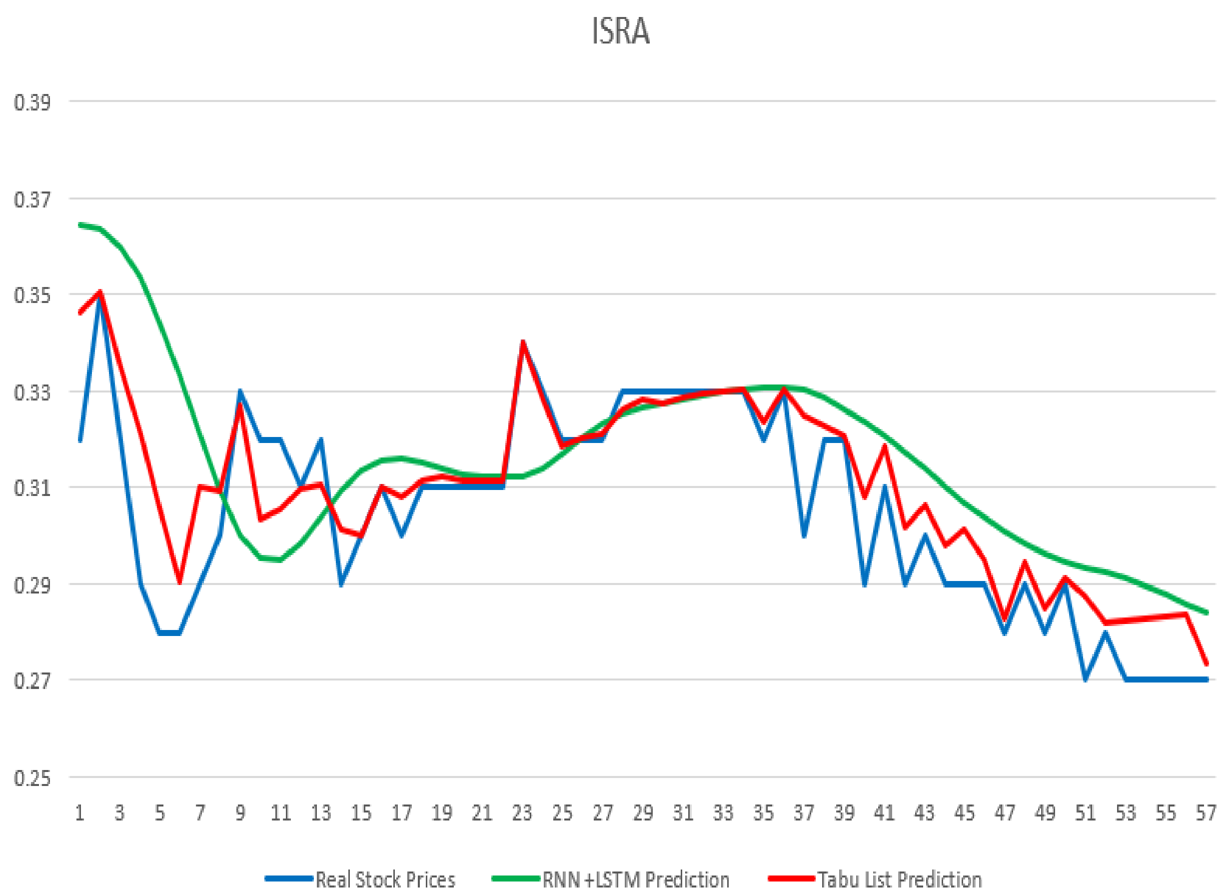


Figure 29. Investment Arab East for Real Estate Investments Co (REAL) predicted stock prices

Another company that was used for stock prediction model we have developed is Arab International Hotels (AIHO), figure 30 shows the real predicted prices for a period of time in year 2019 with comparison to predicted prices resulted from applying RNN with LSTM, and applying RNN with LSTM and TL.

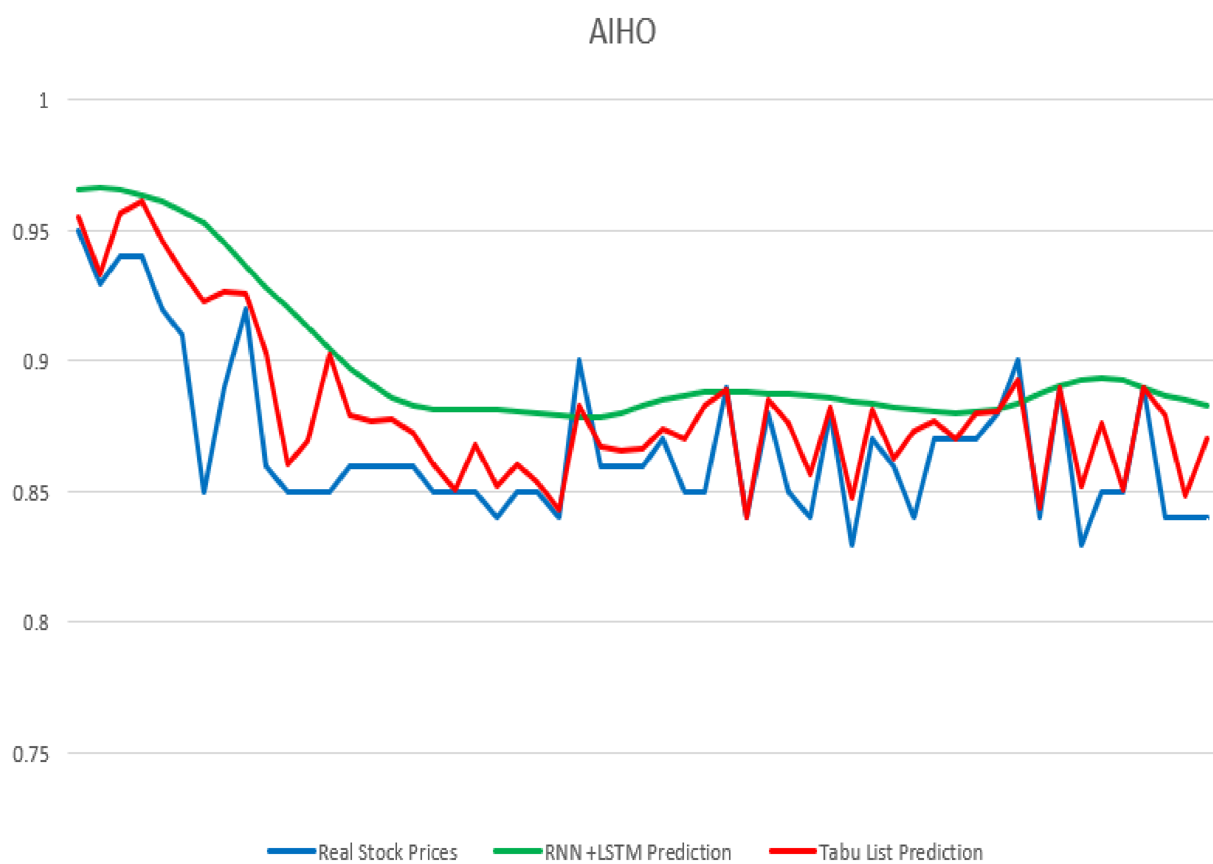


Figure 30. Investment Arab East for Real Estate Investments Co (REAL) predicted stock prices

Another company that was used for stock prediction model we have developed is Arab Ihdathiat Co-Ordinates (IHCO), figure 31 shows the real predicted prices for a period of time in year 2019 with comparison to predicted prices resulted from applying RNN with LSTM, and applying RNN with LSTM and TL.

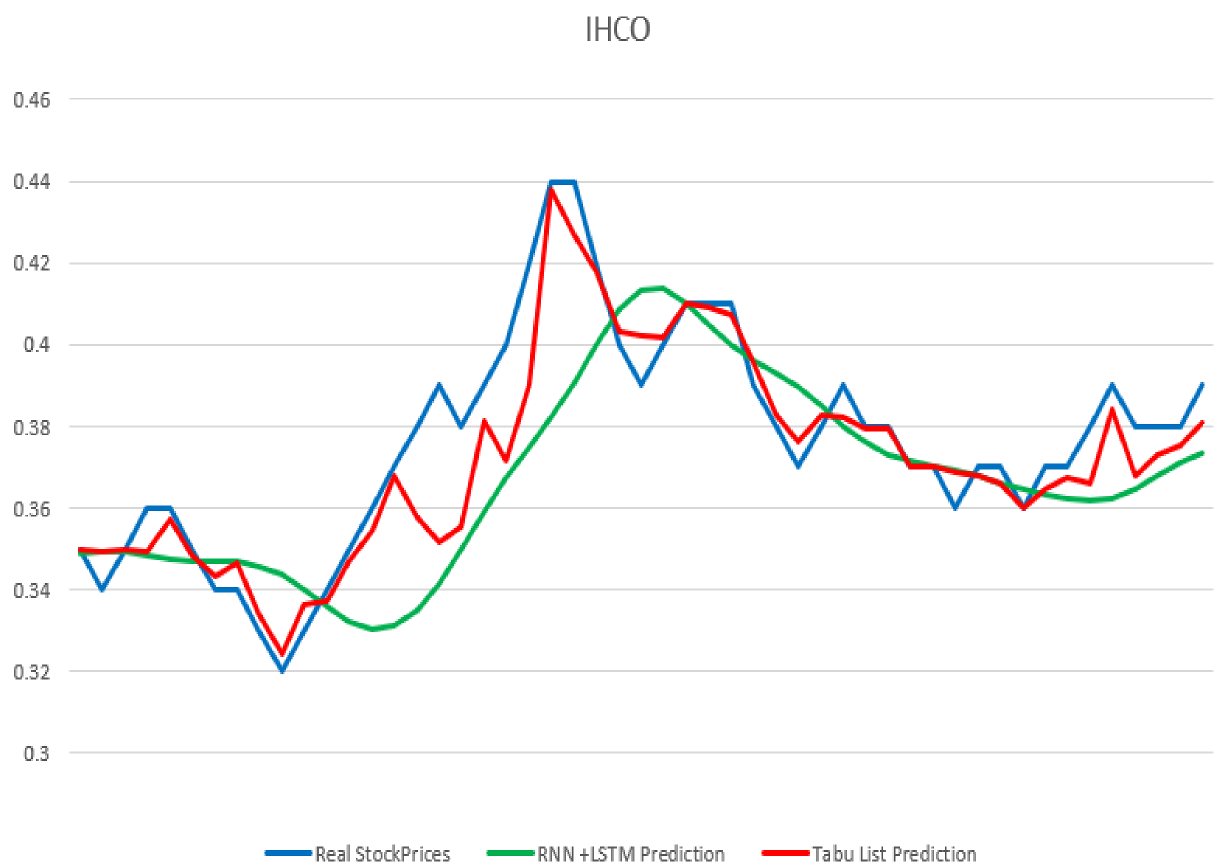


Figure 31. is Arab Ihdathiat Co-Ordinates (IHCO) predicted stock prices

Another company that was used for stock prediction model we have developed is The Jordan Pharmaceutical Manufacturing (JPHM), figure 32 shows the real predicted prices for a period of time in year 2019 with comparison to predicted prices resulted from applying RNN with LSTM, and applying RNN with LSTM and TL.

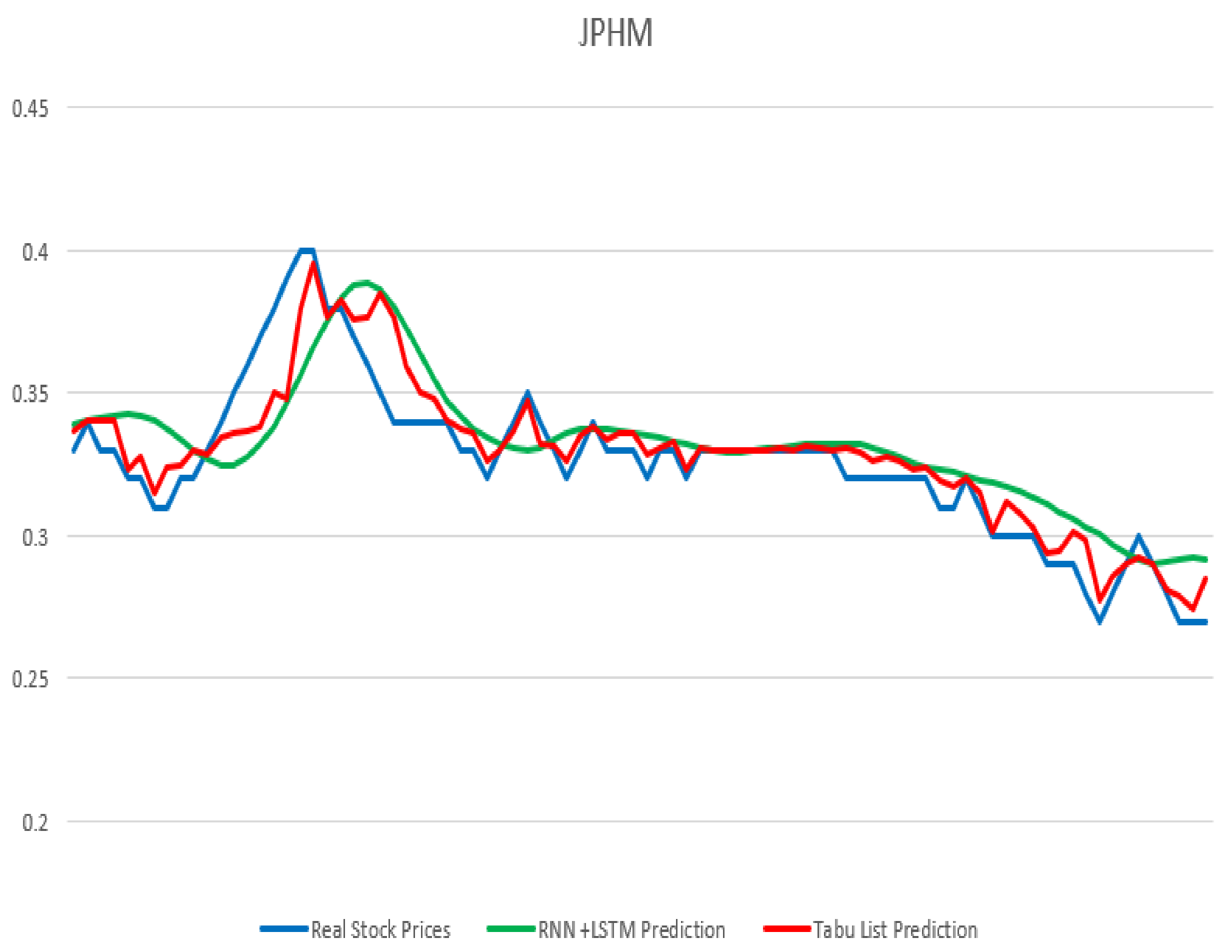


Figure 32. The Jordan Pharmaceutical Manufacturing (JPHM) predicted stock prices

Another company that was used for stock prediction model we have developed is Alshamekha for Realestate and Financial Investments Co.Ltd (VFED), figure 33 shows the real predicted prices for a period of time in year 2019 with comparison to predicted prices resulted from applying RNN with LSTM, and applying RNN with LSTM and TL.

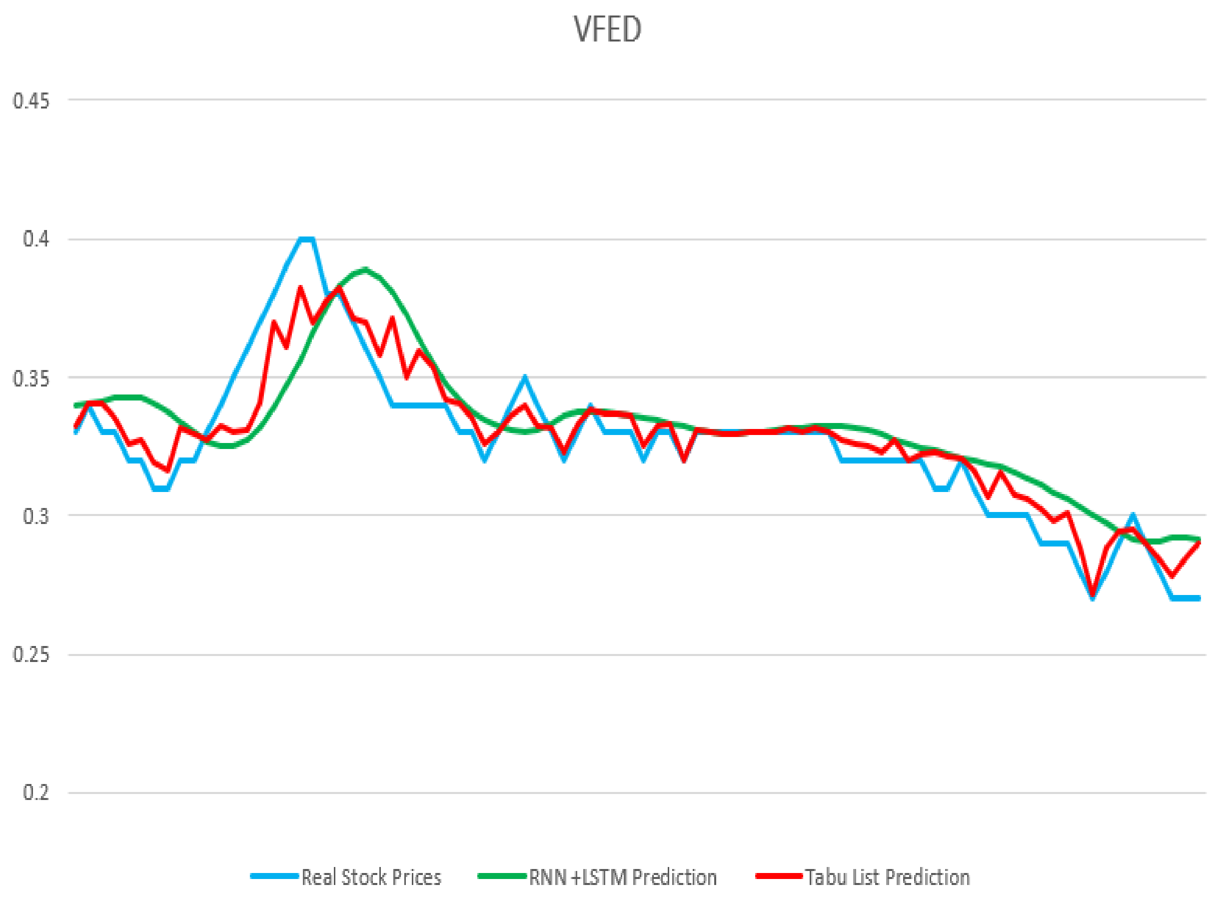


Figure 33. Alshamekha for Realestate and Financial Investments Co.Ltd (VFED) predicted stock prices

Another company that was used for stock prediction model we have developed is Deera Investment & Real Estate Development Co (DERA), figure 34 shows the real predicted prices for a period of time in year 2019 with comparison to predicted prices resulted from applying RNN with LSTM, and applying RNN with LSTM and TL.

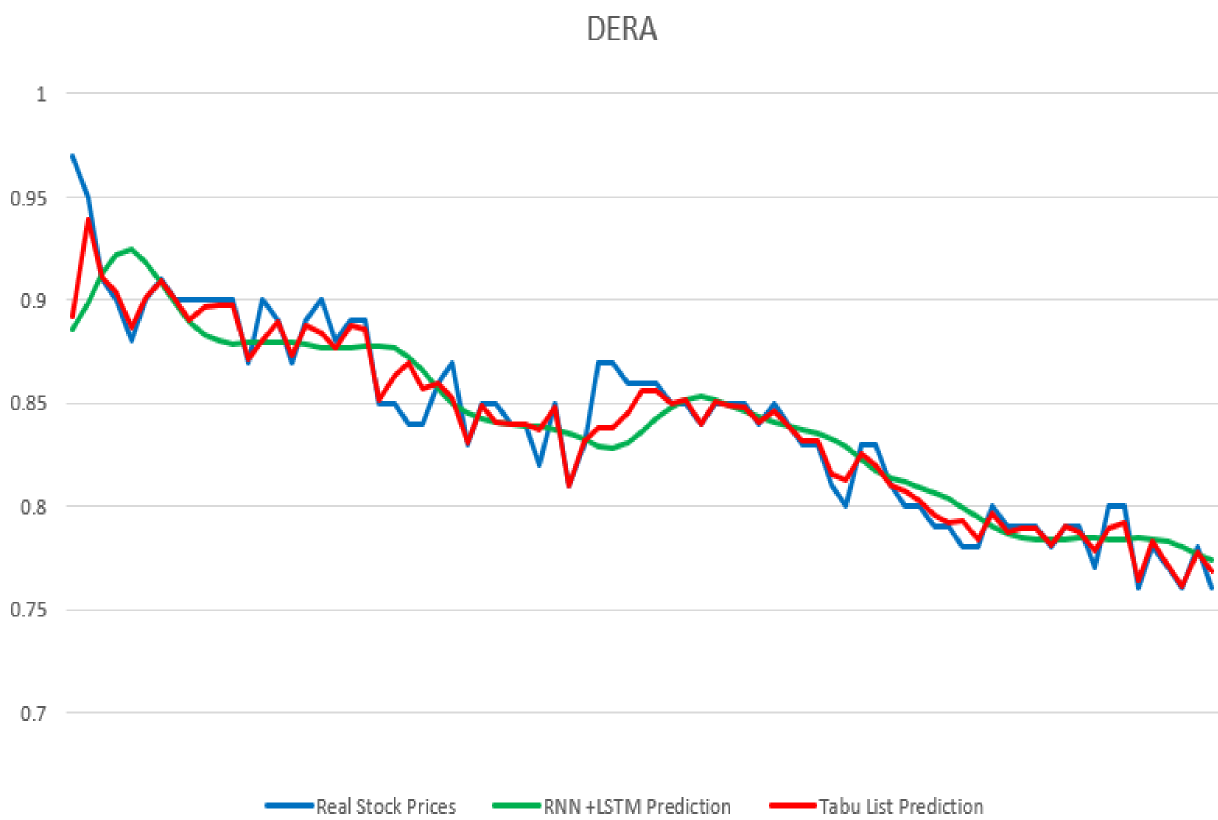


Figure 34. Deera Investment & Real Estate Development Co (DERA) predicted stock prices

Table 3 Results for companies that issued their stocks since or after 2017

	Company Name	Real Stock Price	RNN with LSTM Prediction	RNN with LSTM and Tabu List Prediction
1	Investment House for Financial Services (INVH)	0.093333333	0.087125169	0.089655833
2	Investment Arab East for Real Estate Investments Co (REAL)	1.033809524	1.082587809	1.057316
3	AlIsraa for Islamic Finance and Investment (ISRA)	0.305438596	0.315876699	0.310757228
4	Arab International Hotels (AIHO)	0.867636364	0.899071504	0.881842255
5	Arab Ihdathiat Co-Ordinates (IHCO)	0.376078431	0.376078431	0.376078431
6	Jordan Pharmaceutical Manufacturing (JPHM)	0.325930233	0.332161029	0.32912786
7	Alshamekha for Realestate and Financial Investments Co.Ltd (VFED)	0.325930233	0.332161029	0.328449895
8	Deera Investment & Real Estate Development Co (DERA)	0.841265823	0.840355828	0.839507646

4.3 Assessing RNN with LSTM and Tabu List Accuracy

During the research we realized that errors are the main difference between the prediction of real and predicted stock prices, we calculated the average error from every company's dataset we have used to evaluate the accuracy of our model with other models proposed in previous researches.

Jarrah, et al., (2019) used Mean Squared Error (MSE), Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) criteria to check the overall accuracy of their stock pricing model (Jarrah, et al., 2019).

We calculated the MSE, MAE and RMSE to calculate the percentage of errors occurred in results and compare them with other researchers work.

4.3.1 Mean Squared Error (MSE)

MSE is calculated by taking the average of the square of difference between the original and predicted values of the data (Selvin, et al, 2017).

The general MSE equation is as following:

$$\mathrm{MSE} = \frac{\mathbf{1}}{\boldsymbol{n}} \sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{n}} (\boldsymbol{x_i} + \boldsymbol{y_i})^{\mathbf{2}} \qquad \text{(Selvin, et al, 2017)}$$

Where ***n*** is the total number of stock prices predictions that model generates, ***i*** is the value between 1 to ***n***, ***x*** is the real stock price and ***y*** is the predicted stock price (Selvin, et al, 2017).

In machine learning cases MSE used to measure model's performance, MSE results for our model are mentioned in table 4.

Table 4 MSE results for all companies used in the testing phase

Company Code	Number of test days	MSE
CABK	146	0.00055
JRCD	103	0.00003
EXFB	148	0.00008
JTEL	147	0.00044
JOPH	149	0.00469
RUMM	148	0.00034
JOEP	149	0.00030
JOPT	149	0.00402
AMAL	144	0.00037
IPCH	144	0.00048
JNTH	145	0.00004
JOST	146	0.00011
ULDC	147	0.00146
UINV	147	0.00057
MECE	148	0.00007
BOJX	148	0.00030
AHLI	148	0.00016
SPIC	148	0.00277
IDMC	149	0.00002
JOIB	149	0.00198
ARBK	149	0.00065
INVH	66	0.00007
REAL	63	0.00091
ISRA	57	0.00011
AIHO	55	0.00044
IHCO	51	0.00012
JPHM	86	0.00010
VFED	86	0.00011
DERA	79	0.00016

4.3.2 Mean Absolute Error (MAE)

MAE takes the average of error in accuracy from every stock price in a dataset and gives the output as a difference between real and predicted stock prices (Wang, et al, 2009).

The general MAE equation is as following:

$$\text{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n}$$ (Wang, et al, 2009)

Where n is the total number of stock prices predictions that model generates, i is the value between 1 to n, x is the real stock price and y is the predicted stock price (Wang, et al, 2009).

In machine learning cases MAE used to measure model's performance, MAE results for our model are mentioned in table 5.

Table 5 MAE results for all companies used in the testing phase

Company Code	Number of test days	MAE
CABK	146	0.01856
JRCD	103	0.00390
EXFB	148	0.00593
JTEL	147	0.01220
JOPH	149	0.04015
RUMM	148	0.01322
JOEP	149	0.01270
JOPT	149	0.03932
AMAL	144	0.01275
IPCH	144	0.01317
JNTH	145	0.00449
JOST	146	0.00781
ULDC	147	0.03066
UINV	147	0.01466
MECE	148	0.00522
BOJX	148	0.01155
AHLI	148	0.00967
SPIC	148	0.03712
IDMC	149	0.00289
JOIB	149	0.03419
ARBK	149	0.01646
INVH	66	0.00523
REAL	63	0.02351
ISRA	57	0.00729
AIHO	55	0.01513
IHCO	51	0.00698
JPHM	86	0.00686
VFED	86	0.00738
DERA	79	0.00647

4.3.3 Root Mean Squared Error (RMSE)

RMSE considered as the standard deviation of the errors that occur during prediction, which resembles MSE but theroot of the value is considered while determining the accuracy of the model (Wang, et al, 2009).

The general MAE equation is as following:

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - x_i)^2}{n}} \qquad [35]$$

Where ***n*** is the total number of stock prices predictions that model generates, ***i*** is the value between 1 to ***n***, ***x*** is the real stock price and ***y*** is the predicted stock price (Wang, et al, 2009).

In machine learning cases MAE used to measure model's performance, MAE results for our model are mentioned in table 6.

Table 6 RMSE results for all companies used in the testing phase

Company Code	Number of test days	RMSE
CABK	146	0.01615
JRCD	103	0.00339
EXFB	148	0.00516
JTEL	147	0.01061
JOPH	149	0.03493
RUMM	148	0.0115
JOEP	149	0.01105
JOPT	149	0.00455
AMAL	144	0.02045
IPCH	144	0.00634
JNTH	145	0.01316
JOST	146	0.00607
ULDC	147	0.00566
UINV	147	0.00642
MECE	148	0.00563
BOJX	148	0.03421
AHLI	148	0.01109
SPIC	148	0.01146
IDMC	149	0.00391
JOIB	149	0.00679
ARBK	149	0.02667
INVH	66	0.01275
REAL	63	0.00454
ISRA	57	0.01005
AIHO	55	0.00841
IHCO	51	0.03229
JPHM	86	0.00251
VFED	86	0.02975
DERA	79	0.01432

4.4 Evaluating RNN with LSTM and Tabu List Accuracy with DWT with RNN Accuracy

As mentioned in section 4.3 we used RME, MAE and RMSE to check prediction accuracy based on the research published by Jarrah, et al., (2019) and we compared our results with their results to determine if our model was successful or not.

The mean error criteria results comparison with results mentioned in (Jarrah, et al., 2019) are shown in table 7 below.

Table 7 Mean for MSE, MAE, and RMSE comparison between RNN with LSTM and Tabu List, and Discrete Wavelet Transform with RNN

	MSE	MAE	RMSE
DWT & RNN (DRNN)	0.15996	0.03701	0.19237
Autoregressive Integrated Moving Average (ARIMA)	6.60949	76.5758	8.75076
RNN with LSTM & Tabu List	0.00074	0.01473	0.01275

Chapter 5

Conclusion and Future Work

5.1 Conclusion

In this work, we proposed predicting stock market using RNN with LSTM and Tabu List. In order to test the effectiveness of the proposed model, one company stock data was selected per time to test the optimized configuration of the model.

The proposed RNN with LSTM and Tabu List model provided excellent results and predicted the increase & decrease in stock prices despite the size historical stock prices data set. Moreover, as shown in sections 4.1 and 4.2, that the results obtained from applying RNN with LSTM and Tabu List were much better than the results obtained from applying RNN with LSTM alone, this means the model we proposed was effective and provide better prediction accuracy.

In addition, we compared the mean of errors in the results produced from applying RNN with LSTM and Tabu List with the results mentioned in Jarrah, et al., (2019) research results by using Discrete Wavelet Transform with RNN (as shown in section 4.4), the comparison outcome that our model have mean of errors less than the model proposed by Jarrah, et al., (2019) research.
In conclusion, using RNN with LSTM and Tabu List to predict stocks prices is effective and reliable and can help the Jordanian companies to anticipate the upcoming changes in the Jordanian Market.

5.2 Future Work

The plan for the future is to make the developed model is applicable for more stock markets in the Middle East region to anticipate future stock market changes and provide a reliable forecast feature for companies in different sectors and businesses.

References

Alhroob, A., Tarawneh, H., & Ayob, M. (2018). A Hybrid Simulated Annealing with Tabu List and Allowable Solutions Memory to Solve University Course Timetabling Problem.

Alturki, F. A., & Aldughaiyem, A. M. (2020). Trading Saudi Stock Market Shares using Multivariate Recurrent Neural Network with a Long Short-term Memory Layer. Machine learning, 11(9).

Al-Radaideh, Q. A., Assaf, A. A., & Alnagi, E. (2013, December). Predicting stock prices using data mining techniques. In The International Arab Conference on Information Technology (ACIT'2013).

Anirudh Rao, (2019) Recurrent Neural Networks (RNN) Tutorial - Analyzing Sequential Data Using TensorFlow In Python, https://www.edureka.co/blog/recurrent-neural-networks/

Azoff, E.M. "Neural Network Time Series Forecasting of Financial Markets". (1994). Chichester; New York: Wiley.

Carlos Serrano-Cinca. "Self organizing neural networks for financial diagnosis". (1996). Decision Support Systems, 17:227–238.

Christopher Olah. (2015). Understanding LSTM Networks. https://colah.github.io/posts/2015-08-Understanding-LSTMs

Chung, Hyejung & Shin, Kyung-shik. (2018). Genetic Algorithm-Optimized Long Short-Term Memory Network for Stock Market Prediction.

DiPietro, R., & Hager, G. D. (2020). Deep learning: RNNs and LSTM. In Handbook of Medical Image Computing and Computer Assisted Intervention (pp. 503-519). Academic Press.

Fama, E.F. and K.R. French. “Permanent and Temporary Components of Stock Prices”. (1988). Journal of Political Economics, vol. 96, no. 2, pp.264-273.

Ghosh, A., Bose, S., Maji, G., Debnath, N., & Sen, S. (2019, September). Stock Price Prediction Using LSTM on Indian Share Market. In Proceedings of 32nd International Conference on (Vol. 63, pp. 101-110).

Glover, F. (1990). Tabu search: A tutorial. Interfaces, 20(4), 74-94.

Glover, F., & Laguna, M. (1998). Tabu search. In Handbook of combinatorial optimization (pp. 2093-2229). Springer, Boston, MA.

Hiemstra, Y. “A stock market forecasting support system based on fuzzy logic”. (1994). Proceedings of the Twenty-Seventh Hawaii International Conference on System Sciences, Vol. III: Information Systems: Decision Support and Knowledge-Based Systems, Volume 3, 4-7 Page(s):281 – 287.

Hochreiter, S. (1998). The vanishing gradient problem during learning recurrent neural nets and problem solutions. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 6(02), 107-116.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.

Hsieh, T. J., Hsiao, H. F., & Yeh, W. C. (2011). Forecasting stock markets using wavelet transforms and recurrent neural networks: An integrated system based on artificial bee colony algorithm. Applied soft computing, 11(2), 2510-2525.

Ikkai, Y., Oka, H., & Komoda, N. (2005, September). A high-speed integrated scheduling system with tabu search for large-scale job shops problems with group constraints. In Proceedings of the 5th WSEAS International Conference on Applied Informatics and Communications (pp. 221-226). World Scientific and Engineering Academy and Society (WSEAS).

Jarrah, M., & Salim, N. (2019). A recurrent neural network and a discrete wavelet transform to predict the Saudi stock price trends. Int. J. Adv. Comput. Sci. Appl, 10(4), 155-162.

Jahan, I., & Sajal, S. (2018). Stock Price Prediction using Recurrent Neural Network (RNN) Algorithm on Time-Series Data. In 2018 Midwest Instruction and Computing Symposium.

Józefowska, J., Waligóra, G., & Węglarz, J. (2002). Tabu list management methods for a discrete–continuous scheduling problem. European Journal of Operational Research, 137(2), 288–302

Karpathy, A., & Fei-Fei, L. (2015). Deep visual-semantic alignments for generating image descriptions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3128-3137).

Khoa, N. L. D., Sakakibara, K., & Nishikawa, I. (2006, October). Stock price forecasting using back propagation neural networks with time and profit based adjusted weight factors. In 2006 SICE-ICASE International Joint Conference (pp. 5484-5488). IEEE.

Kim, K. J., & Han, I. (2000). Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. Expert systems with Applications, 19(2), 125-132.

Kim, H. J., & Shin, K. S. (2007). A hybrid approach based on neural networks and genetic algorithms for detecting temporal patterns in stock markets. Applied Soft Computing, 7(2), 569-576.

Koutnik, J., Greff, K., Gomez, F., & Schmidhuber, J. (2014). A clockwork rnn. arXiv preprint arXiv:1402.3511.

Leonardo dos Santos Pinheiro, Mark Dras, "Stock Market Prediction with Deep Learning: A Character-based Neural Language Model for Event-based Trading", (2017), Proceedings of Australasian Language Technology Association Workshop, pages 6-15.

Lin, X., Yang, Z., & Song, Y. (2009). Short-term stock price prediction based on echo state networks. Expert systems with applications, 36(3), 7313-7317.

Lipton, Z. C., Berkowitz, J., & Elkan, C. (2015). A critical review of recurrent neural networks for sequence learning. arXiv preprint arXiv:1506.00019.

Minami, S. (2018). Predicting equity price with corporate action events using lstm-rnn. Journal of Mathematical Finance, 8(1), 58-63.

Moghar, A., & Hamiche, M. (2020). Stock Market Prediction Using LSTM Recurrent Neural Network. Procedia Computer Science, 170, 1168-1173.

Pascanu, R., Mikolov, T., & Bengio, Y. (2012). Understanding the exploding gradient problem. CoRR, abs/1211.5063, 2, 417.

Piech, C., Bassen, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L. J., & Sohl-Dickstein, J. (2015). Deep knowledge tracing. In Advances in neural information processing systems (pp. 505-513).

Qaralleh, E., Abandah, G., & Jamour, F. T. (2013). Tuning recurrent neural networks for recognizing handwritten arabic words.

Qian, B., & Rasheed, K. "Stock market prediction with multiple classifiers". (2007). Applied Intelligence, 26(1), 25-33.

Qiu, Jiayu & Wang, Bin & Zhou, Changjun. (2020). Forecasting stock prices with long-short term memory neural network based on attention mechanism. PLOS ONE. 15.

R. Battiti and G. Tecchiolli, "Training neural nets with the reactive tabu search". (1995). in IEEE Transactions on Neural Networks, vol. 6, no. 5, pp. 1185-1200, Sept.

Recurrent Neural Network, DeepAI. (2019), https://deepai.org/machine-learning-glossary-and-terms/recurrent-neural-network

Roondiwala, M., Patel, H., & Varma, S. (2017). Predicting stock prices using LSTM. International Journal of Science and Research (IJSR), 6(4), 1754-1756.

Salhi, S. (2002). Defining tabu list size and aspiration criterion within tabu search methods. Computers & Operations Research, 29(1), 67-86.

Sangaiah, A. K. (Ed.). (2019). Deep Learning and Parallel Computing Environment for Bioengineering Systems. Academic Press.

Schmidhuber, J., & Hochreiter, S. (1997). Long short-term memory. Neural Comput, 9(8), 1735-1780.

Selvin, S., Vinayakumar, R., Gopalakrishnan, E. A., Menon, V. K., & Soman, K. P. (2017, September). Stock price prediction using LSTM, RNN and CNN-sliding window model. In 2017 international conference on advances in computing, communications and informatics (icacci) (pp. 1643-1647). IEEE.

Shah, D., Isah, H., & Zulkernine, F. (2019). Stock market analysis: A review and taxonomy of prediction techniques. International Journal of Financial Studies, 7(2), 26.

Shang, X., Lining, X., Ling, W., & Kai, Z. (2019). Comprehensive learning pigeon-inspired optimization with tabu list. Science China Information Sciences, 62(7), 07028.

Siegelmann, H. T., & Sontag, E. D. (1991). Turing computability with neural nets. Applied Mathematics Letters, 4(6), 77-80.

Sureshkumar, K. K., & Elango, N. M. (2012). Performance analysis of stock price prediction using artificial neural network. Global journal of computer science and Technology.

Tsubakitani, S., & Evans, J. R. “Optimizing tabu list size for the traveling salesman problem”. (1998). Computers & Operations Research, 25(2), 91-97.

Vargas, M. R., De Lima, B. S., & Evsukoff, A. G. (2017, June). Deep learning for stock market prediction from financial news articles. In 2017 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA) (pp. 60-65). IEEE.

Varsamopoulos, S., Bertels, K., & Almudever, C. G. (2018). Designing neural network based decoders for surface codes. arXiv preprint arXiv:1811.12456.

Vasant, P. M., Ganesan, T., & Elamvazuthi, I. “Hybrid tabu search Hopfield recurrent ANN fuzzy technique to the production planning problems: a case study of crude oil in refinery industry”. (2012). International Journal of Manufacturing, Materials, and Mechanical Engineering (IJMMME), 2(1), 47-65.

Wang, Z., & Bovik, A. C. (2009). Mean squared error: Love it or leave it? A new look at signal fidelity measures. IEEE signal processing magazine, 26(1), 98-117.

Wei, L. Y., & Cheng, C. H. (2012). A hybrid recurrent neural networks model based on synthesis features to forecast the Taiwan stock market. Int. J. Innov. Comput. Inf. Control, 8(8), 5559-5571.

Willmott, C. J., & Matsuura, K. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. Climate research, 30(1), 79-82.

Xie, X. K., & Wang, H. (2017). Recurrent neural network for forecasting stock market trend. In Computer Science, Technology and Application: Proceedings of the 2016 International Conference on Computer Science, Technology and Application (CSTA2016) (pp. 397-402).

Yao, K., Cohn, T., Vylomova, K., Duh, K., & Dyer, C. (2015). Depth-gated recurrent neural networks. arXiv preprint arXiv:1508.03790, 9.

Ye, J., Qiao, J., Li, M. A., & Ruan, X. (2007). A tabu based neural network learning algorithm. Neurocomputing, 70(4-6), 875-882.

Yoshihara, A., Fujikawa, K., Seki, K., & Uehara, K. (2014, December). Predicting stock market trends by recurrent deep neural networks. In Pacific rim international conference on artificial intelligence (pp. 759-769). Springer, Cham.

Zineb Zineb, Saaid Achchab. (2019). A new approach for Trading based on Long-Short Term memory technique. International Journal of Computer Science Issues, IJCSI Press.