Market Forecasting Using Deep Learning Model

Abstract

In recent years, deep learning has developed rapidly. More and more researchers have applied deep learning to different fields, like data analysis, predictions. In terms of predictions, many neural networks, which are an important part of deep learning, have been used in predicting stock price. In this project, we will focus on predicting the revenue of companies. We test and compare the performances of Recurrent Neural Network (RNN) and Multilayer Perceptron (MLP) and try to figure out a better neural network for predicting the revenue of companies. The result indicates that MLP is a better neural network for making predictions on the revenue of companies in the next quarter.

1 Introduction

Market forecasting is important for companies since it can guide important decisions such as the prices of new products and new product launch. How to make rational decisions based on the previous information is a necessary part of a company. As machine learning develops rapidly, Najafabadi et al. [1] demonstrate that deep learning has been used for analyzing large volumes of data for business analysis and decisions. Using neural networks to learn the pattern in the past quarters and make the right decisions to improve the revenue and profit of the next quarter.

Both ANN and RNN have been contributing to stock market forecasting for a few years. As indicated by Vaisla and Bhatt [2], they use ANN, which is Multilayer Perceptron (MLP) neural network, to predict stock market price and figured out that the Neural Networks prediction works better than the Statistical technique. Hiransha el al. [3] launched experiments on testing the performances of predicting stock market prices on 4 types of neural networks including Multilayer Perceptron (MLP), Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN). And the result indicated that RNN has good performances on stock market prediction.

In this paper, we are going to discuss the performances of Multilayer Perceptron (MLP) and Recurrent Neural Network (RNN) in terms of predicting the revenue and profit of companies based on the past financial report.

The rest of the report is organized as follows: an approach section, we will introduce MLP and RNN models including the architectures and formulations of neural networks; in the experiment section, we evaluate the performances of our models quantitatively and qualitatively; in conclusion, we will point out the future work.

2 Methodology

Inputs of both neural networks are 13 features in the dataset, which will be discussed in section 3. The output is the quarterly net income of Apple (millions of US dollar).

2.1 Artificial neurons

Figure 1 is an artificial neuron which is a basic part of artificial neuron networks. The artificial neuron is a simple processing unit that computes an activation using an input function based on inputs and then passes the activation to an activation function to get the output of this neuron.

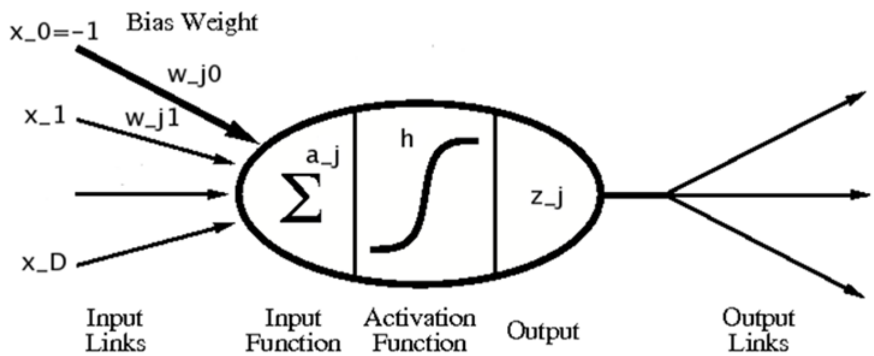

Figure 1: Artificial neuron.

Figure 2 indicates how to get activations. Inputs are multiplied by their own weights and plus a bias. And then the neuron sums up all the values computed by the mentioned approach. During the learning process, weights will be updated to learn the model.

$$a_j = \sum_{i=1}^{D} \left(w_{ji}^{(1)} x_i + x_{j0}^{(1)} \right)$$

Figure 2: Activations.

Activations are passed through an activation function to get outputs of neurons.

$$z_j = h(a_j)$$

Figure 3: Activation function.

2.2 Multi-layer perceptron

MLP is an example of a simple feed-forward network. MLP consists of three kinds of layers including input, hidden and output layers. And each layer consists of neurons and each neuron receives inputs from the previous layer and is connected to all the neurons in the next layer, which means the neural network is fully connected. This project uses an MLP which has three layers including one input layer, one hidden layer, and one output layer. The input layer includes 13 nodes that are features of the input vector, which will be discussed in the dataset section. The hidden layer includes 10 hidden nodes. And the output layer includes one node.

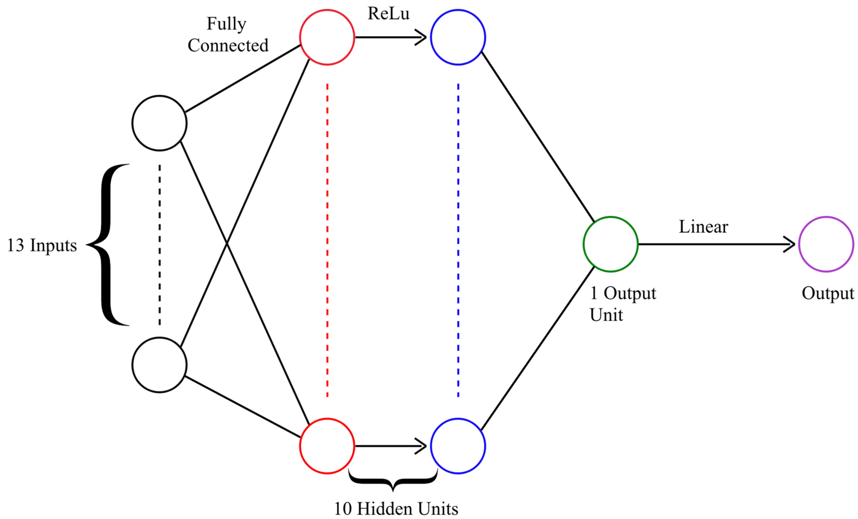

Figure 4: Multi-layer perceptron.

2.2.1 Activation function

This neural network uses the ReLU activation function for the hidden layer and uses a linear activation function for the output layer. The advantage of using linear activation function for the output layer is that the output is straight forward and is easy to compare with the actual data without additional computation. But the disadvantage is that the outputs will be very large values and computing losses using the loss function will take much more time than using activation functions which will take more time to finish the learning process. The reason why we use a linear activation function is that we do not have a large dataset so the computation time will not be too long and the output will be straightforward for us to monitor the network.

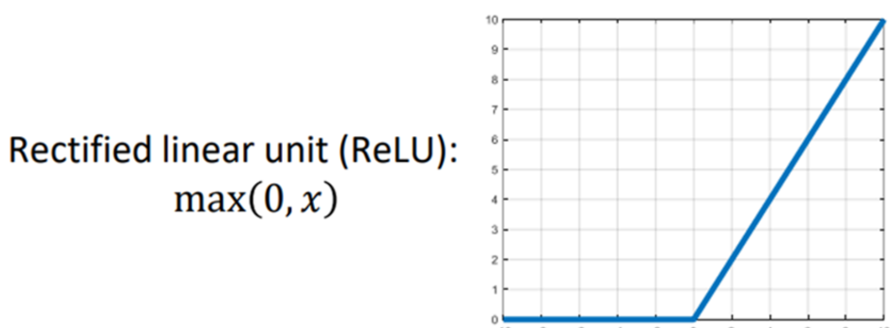

Figure 5: ReLU activation function.

2.2.2 Loss function

MLP uses mean squared error (MSE) to compute the loss for the network. The below equation is to compute the mean squared error. The equation computes the average squared values of the difference between the predicted values and the corresponding predicted values. In this project, the losses are large since we use a linear activation function for the output layer.

$$MSE = \frac{1}{n}\Sigma\left(\underbrace{y - \widehat{y}}_{\text{The square of the difference between actual and predicted}}\right)^2$$

Figure 6: Loss function: mean squared error.

2.3 Recurrent Neural Network

RNN takes inputs from two sources, one is from the dataset and another from the previous hidden state. In this project, the hidden state has 256 dimensions since the hidden state with 256 dimensions converge most of the time. The inputs from the dataset and the hidden state are multiplies by their own weights to compute the new hidden state. The output is computed by multiplying the current hidden state with its own weights. And these weights are what the network is going to learn.

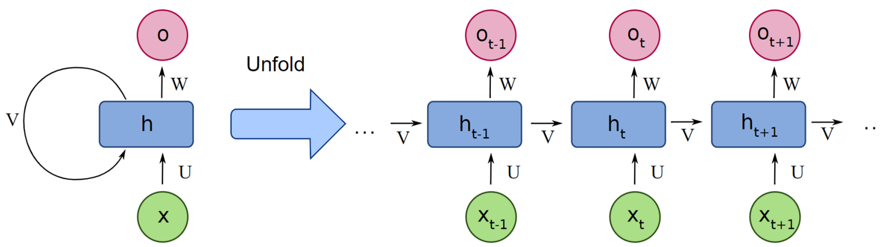

Figure 7: Recurrent neural network.

2.3.1 Loss function

We also use MSE loss function to compute the losses in order to monitor the learning situations of both MLP and RNN so that we can compare the results while both networks have similar losses.

3 Experiments, results and analysis

3.1 Dataset

Dataset is based on Apple's financial report from 2005 to 2019 including 60 quarters. Features include the cost of sales, operating expenses, operating income, net sales by iPhone, net sales by iPad, net sales by Mac, net sales by service, net sales by Wearables, Home and Accessories, if a new product comes out (iPhone), if a new product comes out (iPad), if a new product comes out (Mac), if a new product comes out (wearables, home, and accessories), new revolutionary technique.

3.1.1 Changes to the dataset

According to situations in the past years, Apple released new products around the end of September, which is the end of the fourth quarter of that year. However, after consideration and looking into the data, we think the release of new products will have more influence in the next quarter which is the quarter from October to the end of the year. So, we change the feature 'if a new product comes out' to 'if a new product comes out last quarter'.

3.1.2 Training set

Since the size of the dataset is not large, the training set consists of the first 50 quarters. The training set is selected consecutively rather than randomly because of the purpose of this project to compare

Table 1: Information about MLP and RNN

FEATURES	MLP	RNN
Maximum epochs	5000	50000
Loss function	MSE	MSE
Optimizer	Adam	Adam
Learning rate	0.0005	0.2
The loss after training	40980.80859375	41371.1796875

the performances of neural networks. With consecutive data, we can produce a plot that compares the actual data and the predicted data, which can be straightforward to figure out the performance of the neural network.

3.1.3 Testing set

The testing set consists of the data of the last 10 quarters. It is enough to reflect the trend of the predictions.

3.2 Method

In order to compare the performances of MLP and RNN, we decided to train both networks until the losses for both networks are similar. Table 1 includes the learning rates, epochs and the losses at the end of the training and so on.

3.3 Results

Figure 8 and 9 are the results of MLP and RNN respectively.

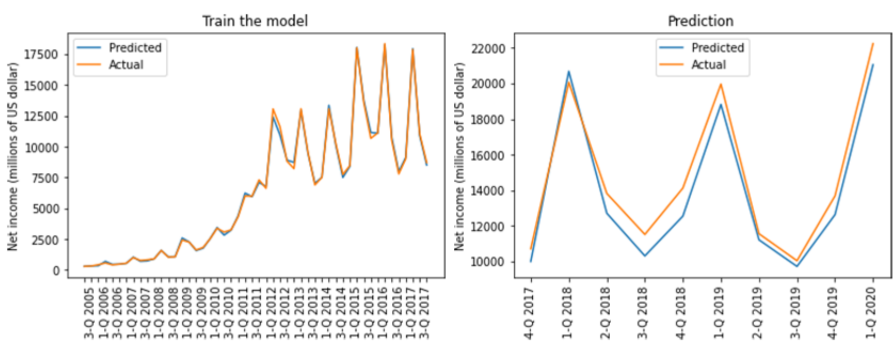

Figure 8: The results of MLP. Left:how well the model learns. Right: how well the predictions are

3.4 Analysis

From the results, we can see that when both neural networks have been trained properly and similarly, Multilayer Perceptron has a better prediction than Recurrent Neural Network. The predictions by MLP have the same trends and similar values as the actual data. However, although the predictions by RNN have the same trends as the actual data, there are huge gaps between the actual values and the predicted values The results also indicate that the profit and revenue of companies do not heavily depend on the profits from the previous quarter but depend on the decisions that the company will make in the next quarter such as whether it releases new technologies or a new version of devices.

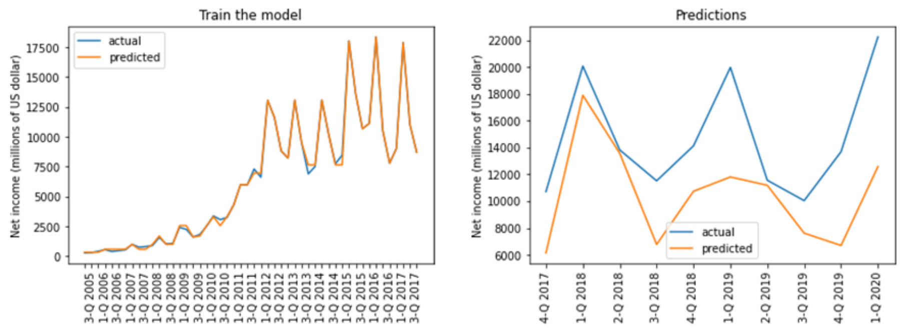

Figure 9: The results of RNN. Left:how well the model learns. Right: how well the predictions are

3.5 Problem

The losses of MLP and RNN are not always at the same level. Due to the construction of the RNN, RNN is easy to converge to a local minimum which is much higher than the loss of MLP. This takes us a lot of time to train RNN until it has a similar loss as the loss of MLP.

4 Conclusion

RNN performs better on stock market prediction because the price highly depends on the previous days. However, regarding the revenue of companies, the revenue highly depends on the decision they make in the quarter. This is also because we changed the dataset so that whether new products come out has been directly connected to the revenue of the corresponding quarter.

In conclusion, Multilayer Perceptron neural network is a better choice for forecasting companies' profits.

5 Future work

So far, many neural networks have been used in stock price prediction, such as Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN). And for predicting stock prices, CNN is the best choice and both LSTM and RNN have better performances than MLP since stock prices heavily depend on the price of the previous days.

For the future work, we are going to test the performances of more neural networks like LSTM. Since LSTM is a special type RNN, we expect that LSTM will have a similar performance as RNN. Also, CNN is another choice for this project. We assume CNN will have the best performance on market profit forecasting since it did well in stock forecasting and it does not depend on the previous data.

Contributions

Repository

https://github.com/infinitusposs/CMPT419-Project-by-Triple-A

References

[1] Najafabadi, M. M., Villanustre, F., Khoshgoftaar, T. M., Seliya, N., Wald, R., & Muharemagic, E. (2015). Deep learning applications and challenges in big data analytics. *Journal of Big Data*, 2(1), 1.

[2] Vaisla, K. S., & Bhatt, A. K. (2010). An analysis of the performance of artificial neural network technique for stock market forecasting. *International Journal on Computer Science and Engineering*, 2(6), 2104-2109.

[3] Hiransha, M., Gopalakrishnan, E. A., Menon, V. K., & Soman, K. P. (2018). NSE stock market prediction using deep-learning models. *Procedia computer science*, 132, 1351-1362.