COMP 702 - MSc Project

Financial Market Prediction

Michael Tattersall - 201518853

Final Report

Contents

Outline and objectives

An outline of the project is as follows:

1. To perform a literature review of financial market prediction using artificial intelligence techniques
2. To review the application of such methods by money managers
3. To develop a state of the art artificial intelligence based prediction model for a financial market

Objectives:

- To compare and contrast existing artificial intelligence techniques by data scientists
- To research the application of such methods by money managers
- To design an artificial intelligence based prediction model for a financial market
- To identify and collect appropriate data for use in the model
- To evaluate and assess the performance of this prediction model
- To complete the deliverables of the project
- To make a contribution to the study of the problem that might be useful to others

Design

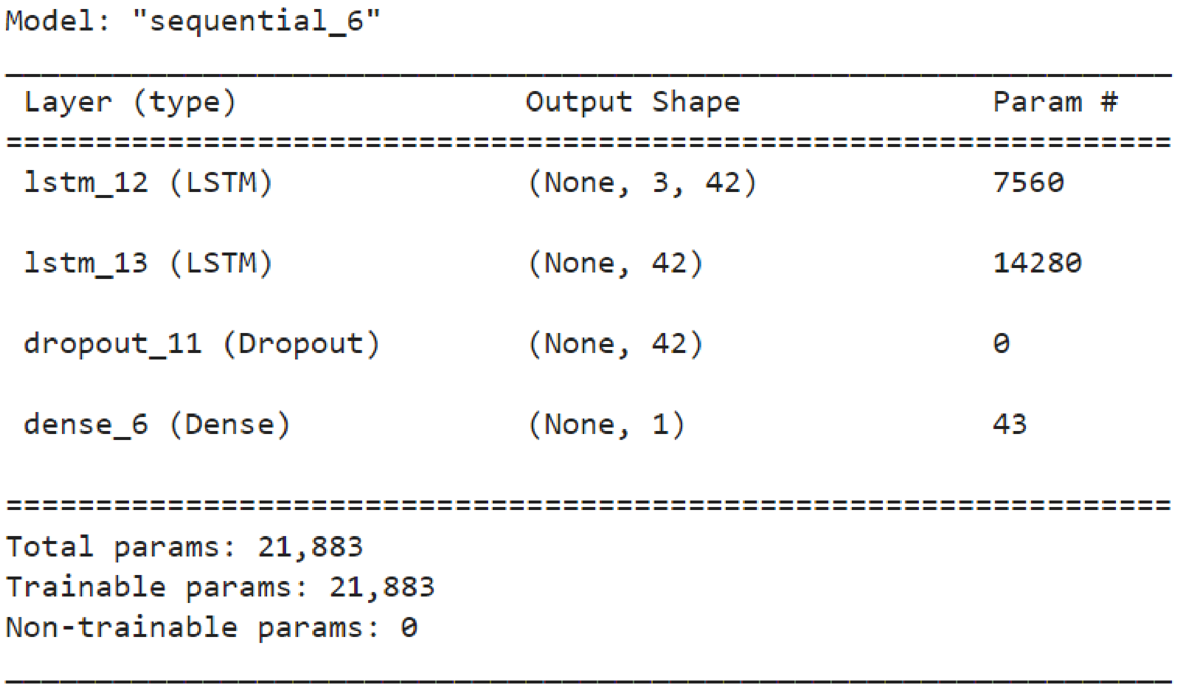
The developed model is an artificial neural network (ANN) designed to handle time-series data.

The model is a long short-term memory (LSTM) model, first developed by Hochreiter & Schmidhuber (1997), which is a type of recurrent neural network which enables efficient backpropogation through time.

The model has been created using the following tools:

- Python coding language
- Keras open-source software library
- Scikit-learn machine learning library
- Pandas library (data frames)
- Numpy library (arrays)
- Matplotlib library (visualisation)

The model is a deep recurrent neural network, with the following layers:



```
Model: "sequential_6"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm_12 (LSTM)              (None, 3, 42)             7560

 lstm_13 (LSTM)              (None, 42)                14280

 dropout_11 (Dropout)        (None, 42)                0

 dense_6 (Dense)             (None, 1)                 43

=================================================================
Total params: 21,883
Trainable params: 21,883
Non-trainable params: 0
_________________________________________________________________
```

There are two LSTM layers, each with a Dropout layer used on the last hidden layer in the network. The dropout layer is a regularization technique used in deep neural networks to minimise the effect of overfitting in a trained network. The Dense layer is the layer for producing the output, which in our case requires 1 unit and is programmed with the ‘relu’ activation function.

The LSTM layers use stochastic gradient descent to minimise loss, and have a built-in hyperbolic tangent activation function (‘tanh’) and a sigmoid recurrent activation function. The loss is specified to compute as mean squared error. An adam optimizer is used to enable adaptive learning rates and provide momentum for stochastic gradient descent to accelerate in the correct direction.

The hyperparameters of the model are:

- The number of units in each LSTM layer (42)
- The Dropout ratio (0.1)
- The learning rate (0.0002)
- The number of epochs (400)
- The Dense layer activation function ('relu')
- The batch size (16)

These hyperparameters have been tuned following extensive experimentation.

Data

The data used is structured historical time-series data from the financial market and is sourced from Yahoo! Finance.[1] The data downloaded as standard comma-delimited text files, with a separate file for every financial instrument. Each data file has fields as follows:

- Date - recorded in the "mm/dd/yyyy" format.
- Time - US Eastern Time (ET) time zone in the "hh:mm" format
- Price - the opening price, the highest price, the lowest price and the closing price of the financial instrument for that day on the exchange.
- Volume - the total number of instruments e.g. shares or contracts traded on that day on the exchange. This may be left out and not input to the model.

The data used is the 'Close' price. The data has been pre-processed using pandas dataframes, and has been 0-1 scaled using a scikit learn MixMaxScaler, which scales the data whilst preserving the original distribution.

In order to experiment with an additional feature, the daily log returns have been calculated. These are calculated by first calculating the daily return of the financial instrument ('Close' price (t+1)/'Close' price (t) - 1), then applying the numpy logarithm method (np.log). Such daily log returns are normally distributed, and have also been scaled when used as an additional feature in a multivariate time series (i.e. additional to the 'Close' price).

To enable evaluation and benchmarking, I have focussed on data used by Guresen, Kayakutlu & Daim (2011) in their study comparing the predictive performance of two distinct types of neural network - mutli-layer perceptron and dynamic artificial neural network. This is the 'Close' price of the NASDAQ Composite Index from 7 October 2008 to 26 June 2009, with a 80/20 split between the daily prices used for training and testing. The models developed in this study use the last four days of prices to forecast the fifth date, which I have adopted as the starting point for comparing the performance of my LSTM model.

Further data has also been used for additional testing of my LSTM model, which is:

[1] https://uk.finance.yahoo.com

- ‘Close’ price of the NASDAQ Composite Index from 1 January 2010 to 1 January 2020.
- ‘Close’ price of Bank of America Stock Price from 1 January 2010 to 1 January 2020.

For these tests, the number of days used to forecast has been extended to 30 days.

In the LSTM model, the 80/20 split between training and testing data has been maintained, with 10% of the training data used for validation purposes.

Money manager research

The literature review is augmented by researching how money managers use AI-based prediction models to gain trading/investment performance in financial markets. This research has been focussed on the world’s largest hedge fund management firms (ranked by assets under management)[2]:

This research will be presented formally in the dissertation.

Evaluation

As stated above, I have focussed the testing of the LSTM model on the data used by Guresen, Kayakutlu & Daim (2011) in their study comparing the predictive performance of two distinct types of neural network - mutli-layer perceptron (MLP) and dynamic artificial neural network (DAN) and hybrid neural networks.

To evaluate the LSTM model I have utilised the measures in Guresen, Kayakutlu & Daim (2011). These are:

1. Mean Square Error (MSE)
2. Mean Absolute Error (MAE)
3. Percentage Mean Absolute Error (MAE%)

These formulas will be fully specified in the dissertation. The inclusion of the ‘Percentage Mean Absolute Error’ is extremely useful for enabling the benchmarking of the LSTM model for different financial instruments and different datasets. This is because the ‘Mean Square Error’ and ‘Mean Absolute Error’ are both based on the price level of the particular financial instrument in question for the time period tested. However, the ‘Percentage Mean Absolute Error’ has as its denominator the average price level of the financial instrument for the time period tested, which enables comparison of the performance of models using different datasets as input.

The LSTM models tested each using ‘Close’ price data of the NASDAQ Composite Index from 7 October 2008 to 26 June 2009 are as follows:

[2] https://www.investopedia.com/articles/personal-finance/011515/worlds-top-10-hedge-fund-firms

- NASDAQ Close Price Only 4 Days (LSTM CP4)
- NASDAQ Close Price Only 3 Days (LSTM CP3)
- NASDAQ Close Price & Log Returns (LSTM CPLR4)
- NASDAQ Close Price & Log Returns (LSTM CPLR3)

The results of the LSTM models tested in comparison to the MLP and DAN tested by Guresen, Kayakutlu & Daim (2011) is as follows:

	MSE	MAD	MAD%
LSTM CP4	2248.98	40.54	2.27
LSTM CP3	2784.67	43.67	2.45
LSTM CPLR4	2806.86	45.81	2.57
LSTM CPLR3	1817.01	36.32	2.04
MLP	2227.42	36.91	2.32
DAN	2349.26	37.29	2.41

The results show that the LSTM model CPLR3 outperforms all other models on each of the specified measures, including the MLP and DAN models developed by Guresen, Kayakutlu & Daim (2011).

This best performing LSTM CPLR3 model has been used for additional testing, using extended datasets and with 30 days used to forecast the 31st day price. The first additional test uses the 'Close' price of the NASDAQ Composite Index from 1 January 2010 to 1 January 2020 (NASDAQ 10YEAR). The second additional test uses the 'Close' price of Bank of America Stock Price from 1 January 2010 to 1 January 2020 (BAC 10YEAR).

The results of the additional LSTM model tests are as follows:

	MSE	MAD	MAD%
LSTM CPLR3	1817.01	36.32	2.04
NASDAQ 10YEAR	26401.33	142.43	1.85
BAC 10YEAR	0.38	0.46	1.54

The results show that the LSTM model CPLR3 used over an extended time period and with 30 days of prices outperforms the same model when used with a short dataset and with only 3 or 4 days of prices.

The suggestion for future development is to research the effect of autocorrelation and credit assignment (i.e. assessing more exactly what components of the LSTM model are responsible for its performance).

Unfortunately, it does not look practical to test the predictive capability of the model with a live trading account. This is because the model does not appear to be a practical tool for risking real capital. Whilst the LSTM model has evaluated well in comparison to the models developed by Guresen, Kayakutlu & Daim (2011), the LSTM model is not sufficiently developed to the extent that it generalises well to predict future on the very dynamic financial markets. Nevertheless, the skills learned in this project may be used as a basis for developing predictive models in the money management industry.

Bibliography

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

Guresen, E., Kayakutlu, G., & Daim, T. U. (2011). Using artificial neural network models in stock market index prediction. *Expert Systems with Applications*, *38*(8), 10389-10397.

References

Dawson, C. W. (2015). *Projects in computing and information systems: a student's guide*. Pearson Education.