

---

# Stock Prediction using Deep Learning

---

August 19, 2021

Nefeli Tavoulari

## 1. Introduction

**Topic.** Forecasting stock prices can yield high profit to investors. There are many techniques to accomplish that, including Fundamental Analysis, Technical Analysis and Machine Learning. This paper analyzes the procedure of predicting TESLA stock prices, experimenting with different neural networks. These types of neural networks are not necessarily suitable for our goal, but they were chosen simply so that I can familiarize with all these different concepts and organize them in my brain.

**Challenges.** Predicting stock prices is a challenging task, due to the many factors that cause them to go up or down (e.g. national economic and social situation). Moreover, time-series forecasting is challenging because of the data not being stationary.

## 2. Methodology

In this section I would like to say a few words about each type of neural network used in this experiment, but also about some relative concepts.

**Artificial Neural Networks.** ANNs are composed of multiple nodes, which imitate biological neurons of human brain. The neurons are connected by links and they interact with each other. The nodes can take input data and perform simple operations on the data. The result of these operations is passed to other neurons. The output at each node is called its activation or node value. Each link is associated with weight. ANNs are capable of learning, which takes place by altering weight values.

**Multi-Layer Perceptron.** An MLP is a class of feedforward artificial neural network (ANN). It consists of at least three layers of nodes: an input layer, a hidden layer and an output layer. Except for the input nodes, each node is a neuron that uses a nonlinear activation function. MLP utilizes a supervised learning technique called backpropagation for

training. Its multiple layers and non-linear activation distinguish MLP from a linear perceptron.

**Convolutional Neural Network.** CNN are distinguished from other neural networks by their superior performance with image, speech, or audio signal inputs. They have three main types of layers, which are:

Convolutional layer

Pooling layer

Fully-connected (FC) layer

The convolutional layer is the first layer of a convolutional network. While convolutional layers can be followed by additional convolutional layers or pooling layers, the fully-connected layer is the final layer. With each layer, the CNN increases in its complexity, identifying greater portions of the image. Earlier layers focus on simple features, such as colors and edges. As the image data progresses through the layers of the CNN, it starts to recognize larger elements or shapes of the object until it finally identifies the intended object

**Recurrent Neural Network.** An RNN is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence. This allows it to exhibit temporal dynamic behavior. Derived from feedforward neural networks, RNNs can use their internal state (memory) to process variable length sequences of inputs. This makes them applicable to tasks such as unsegmented, connected handwriting recognition or speech recognition.

**Long Short-Term Memory.** LSTM is an artificial recurrent neural network (RNN) architecture used in the field of deep learning. Unlike standard feedforward neural networks, LSTM has feedback connections. A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell.

LSTM networks are well-suited to classifying, processing and making predictions based on time series data, since there can be lags of unknown duration between important

---

Email: <tavoulari.1977701@studenti.uniroma1.it>.

*Deep Learning and Applied AI 2021, Sapienza University of Rome, 2nd semester a.y. 2020/2021.*

events in a time series. LSTMs were developed to deal with the vanishing gradient problem that can be encountered when training traditional RNNs. Relative insensitivity to gap length is an advantage of LSTM over RNNs, hidden Markov models and other sequence learning methods in numerous applications.

**Gated Recurrent Unit.** GRUs are a gating mechanism in recurrent neural networks. The GRU is like a long short-term memory (LSTM) with a forget gate, but has fewer parameters than LSTM, as it lacks an output gate. GRU's performance on certain tasks of polyphonic music modeling, speech signal modeling and natural language processing was found to be similar to that of LSTM. GRUs have been shown to exhibit better performance on certain smaller and less frequent datasets.

**Rectifier Linear Unit.** In the context of artificial neural networks, the rectifier or ReLU activation function is an activation function defined as the positive part of its argument:

$$f(x) = x^+ = \max(0, x)$$

### 3. Tools

PyTorch - Pandas - NumPy - Plotly - DVC

### 4. Results

Table 1. Performance comparison.

	MLP	CNN	LSTM	GRU
Train RMSE	37.585734	5.817905	10.604635	5.883437
Test RMSE	66.905527	45.338950	70.779618	32.251542
Train Time	0.399081	1.096348	17.377151	10.162085

### 5. Conclusion

TODO

### 6. References

[https://www.tutorialspoint.com/artificial\\_intelligence/artificial\\_intelligence\\_neural\\_networks.html](https://www.tutorialspoint.com/artificial_intelligence/artificial_intelligence_neural_networks.html)

[https://en.wikipedia.org/wiki/Recurrent\\_neural\\_network](https://en.wikipedia.org/wiki/Recurrent_neural_network)

[https://en.wikipedia.org/wiki/Long\\_short-term\\_memory](https://en.wikipedia.org/wiki/Long_short-term_memory)

[https://en.wikipedia.org/wiki/Gated\\_recurrent\\_unit](https://en.wikipedia.org/wiki/Gated_recurrent_unit)

<https://www.ibm.com/cloud/learn/convolutional-neural-networks>

[https://en.wikipedia.org/wiki/Rectifier\\_\(neural\\_networks\)](https://en.wikipedia.org/wiki/Rectifier_(neural_networks))

<https://medium.com/swlh/stock-price-prediction-with-pytorch-37f52ae84632>

<https://colah.github.io/posts/2015-08-Understanding-LSTMs/>

<https://www.kaggle.com/faressayah/stock-market-analysis-prediction-using-lstm>