Korea Advanced Institute of Science and Technology

Introduction to Financial Engineering

Stock Price Prediction with Machine Learning

Professor:
Woo Chang Kim

Student:
Federico Berto

Course ID:
IE471

ID number:
20204817

Introduction

The goal of this report is to describe the experimental process and result for a simple stock market prediction. In particular, in the first experiment we will predict Samsung Electronics Co., Ltd stock price from 2000 to 2020 [1].
The prediction model is based on the Long Short-Term Memory (LSTM) [3] module in Figure 1[2], which is able to store past information of the data and is thus suitable for time series prediction.

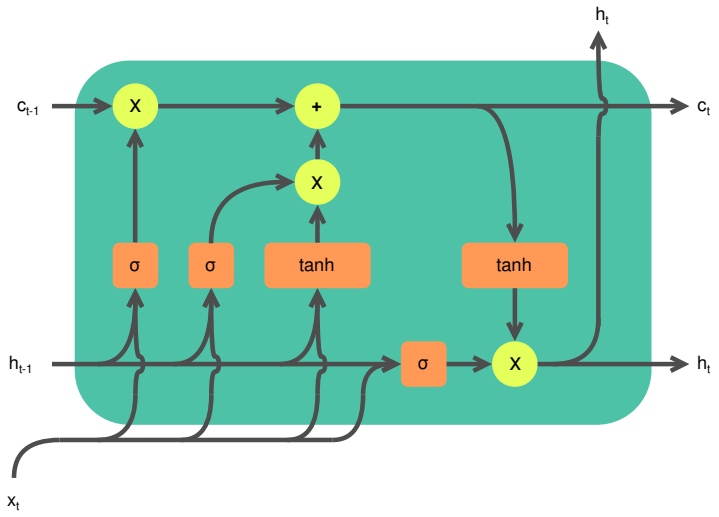

Figure 1: LSTM cell

Further details into the PyTorch [4] implementation can be found in the code, also available on the following Github repository.[3]

Experiments with LSTM

Predicting Samsung's stock price

In this section we provide the required data for the report.

1. Last five data rows of the original dataset:

[1]Source: Yahoo Finance
[2]Source: Wikimedia
[3]Link: `https://github.com/Juju-botu/financial-engineering-ai`.

Date	High	Low	Open	Close	Volume	Adj Close
2020-12-23	74000.0	72300.0	72400.0	73900.0	19411326.0	72085.835938
2020-12-24	78800.0	74000.0	74100.0	77800.0	32502870.0	75890.093750
2020-12-28	80100.0	78200.0	79000.0	78700.0	40085044.0	76768.000000
2020-12-29	78900.0	77300.0	78800.0	78300.0	30339449.0	78300.000000
2020-12-30	81300.0	77300.0	77400.0	81000.0	29417421.0	81000.000000

2. Tensor shape of training sets and test sets:

$$\begin{aligned} &\text{Shape of Training Input Data} = [4780, 1, 6] \\ &\text{Shape of Training Output Data} = [4780, 1] \\ &\text{Shape of Test Input Data} = [493, 1, 6] \\ &\text{Shape of Test Input Data} = [493, 1] \end{aligned} \tag{1}$$

3. Mean Squared Error every 100 epochs until 1000:

Epoch	Loss
100	0.1599
200	0.0392
300	0.0143
400	0.0099
500	0.0071
600	0.0051
700	0.0037
800	0.0026
900	0.0019
1000	0.0013

4. Actual and predicted data: we show them graphically in Figure 2, 3, 4.

5. *Comparison of the test and training set.* We can see from Figure 4 that the model has been properly fitted, which can be also seen in the loss of Table . The test set in Figure 4 shows that the model is still able to predict *well* unseen data, even though the gap between actual and predicted data grows wider at the end of 2020. This may also be due to unseen events, i.e. the COVID-19 pandemic.

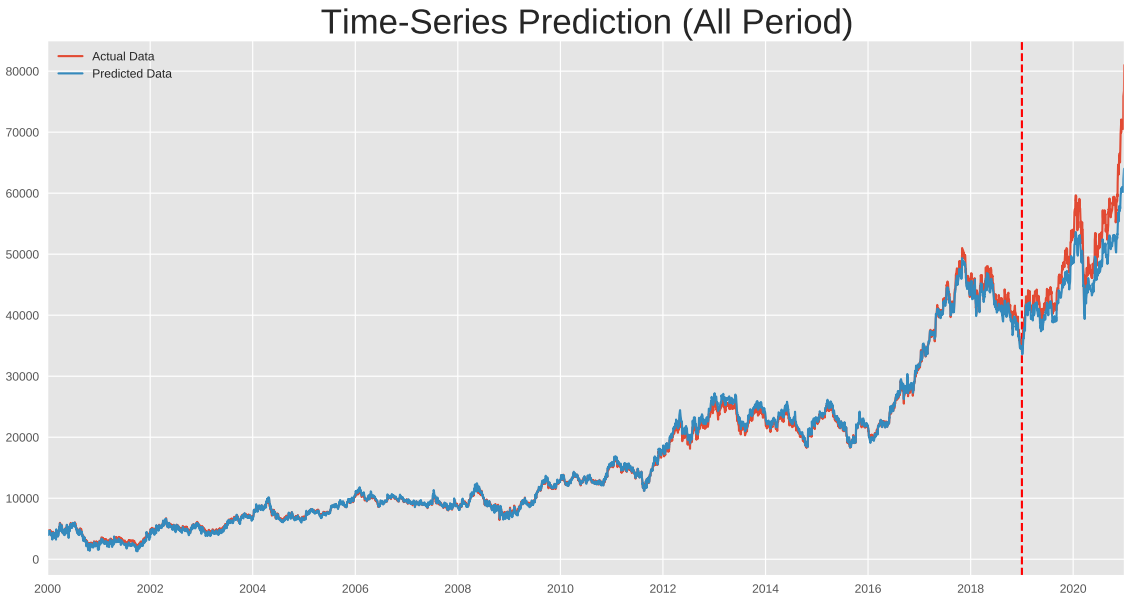

Figure 2: Actual and predicted adjusted closing price of Samsung's stocks the next day for all the time period

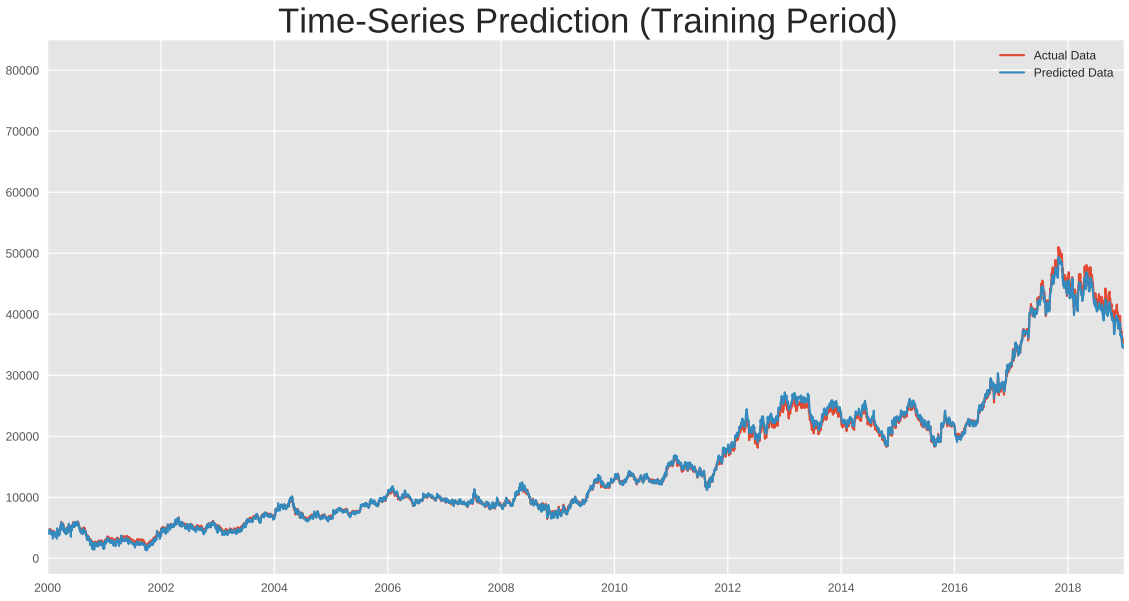

Figure 3: Actual and predicted adjusted closing price of Samsung's stocks the next day for the training period

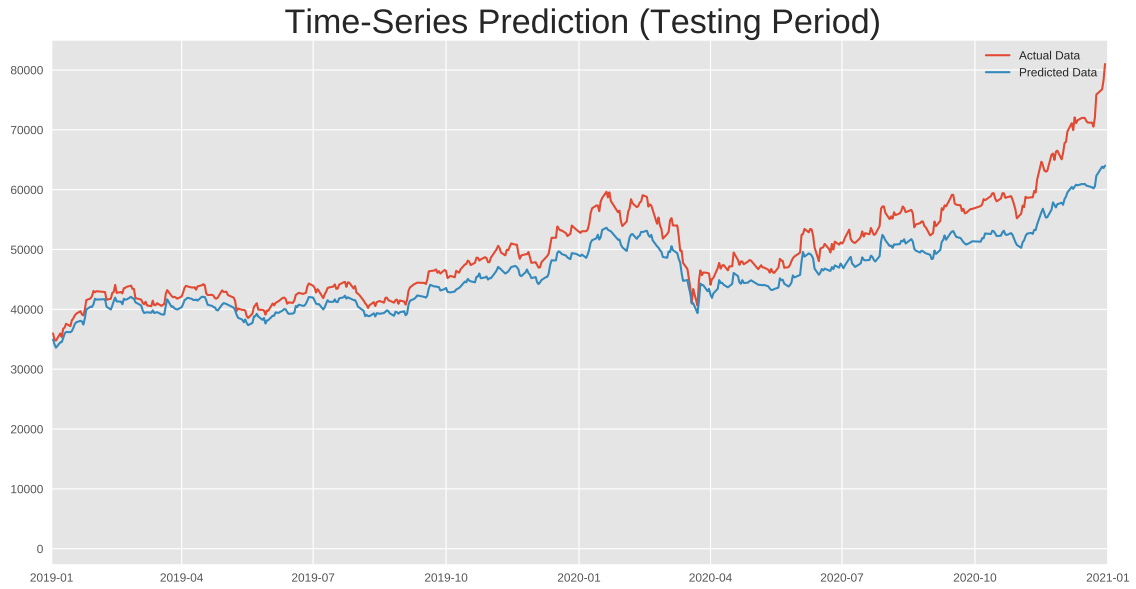

Figure 4: Actual and predicted adjusted closing price of the next day for the test period

6. *Mean Squared Error for both predicted and test data.* The Mean Squared Error (MSE) can be written as:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2}$$

where n is the number of data points, y_i are the actual values and $\hat{y}_i$ are the predicted ones. The values obtained are the following:

	MSE
Training Data	234,280.4
Test Data	18,589,588.0

Predicting Microsoft's stock price

We will extend the experiments with LSTM by predicting the Microsoft Corporation stock price [4]. As we can see in Figure 5, the network was able to fit well the training data, while in the test region of 6 it shows comparable results to Samsung's stock price behavior, which may be due to the COVID-19 era.

[4]Source: Yahoo Finance

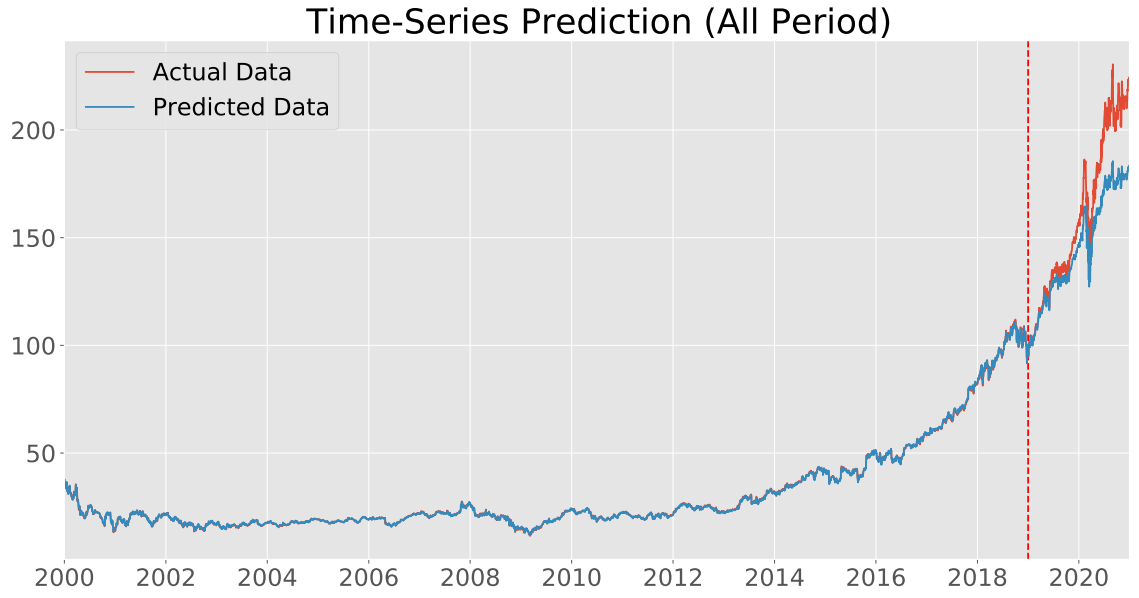

Figure 5: Actual and predicted adjusted closing price of Samsung's stocks the next day for all the time period

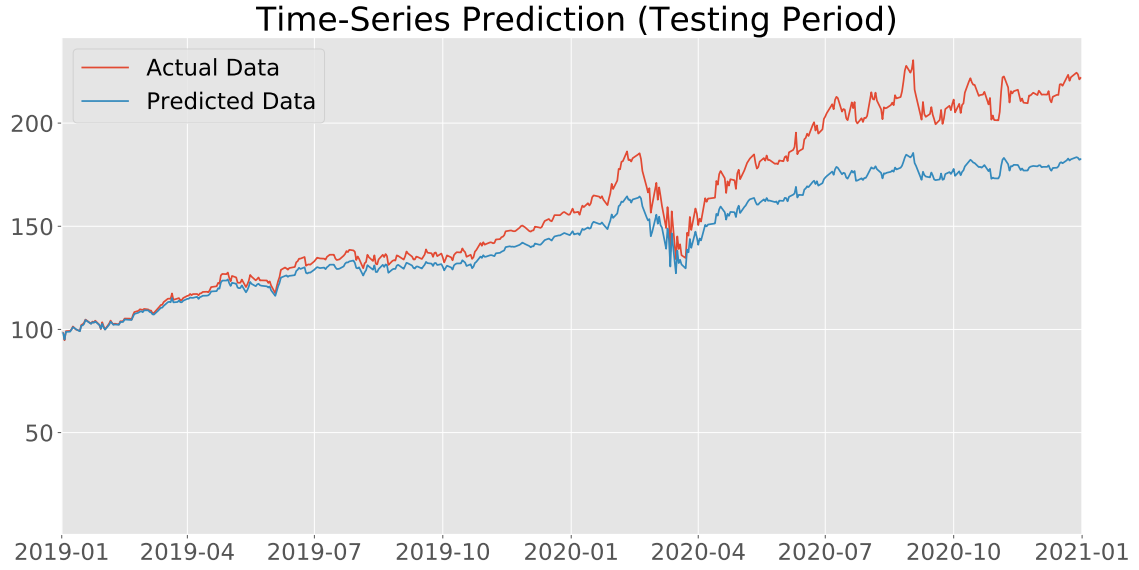

Figure 6: Actual and predicted adjusted closing price of Microsoft's stocks the next day for all the test period

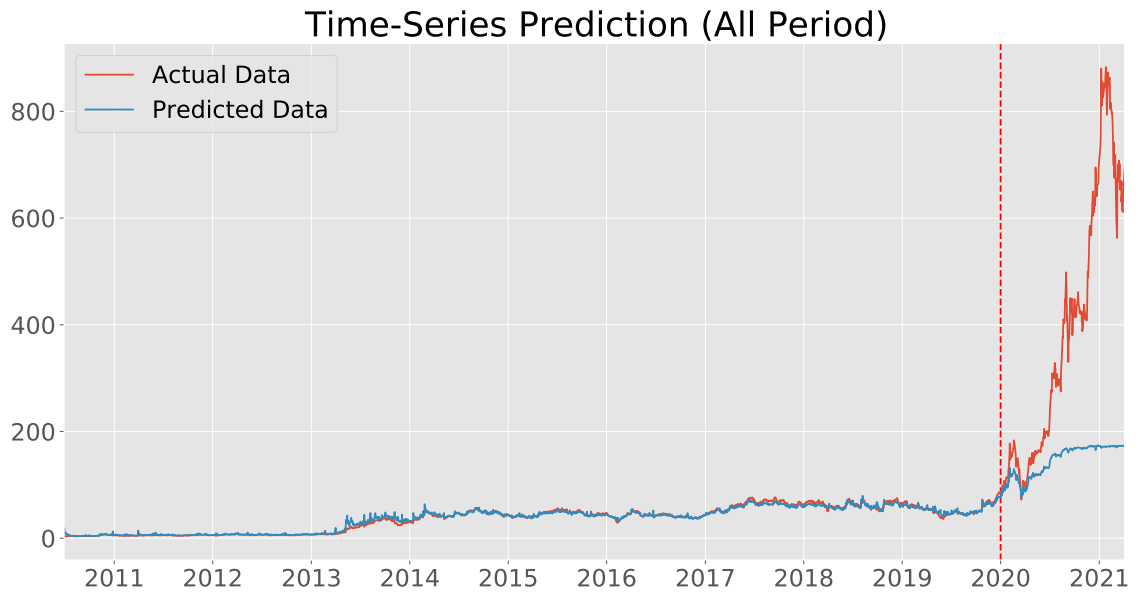

Figure 7: Actual and predicted adjusted closing price of Tesla's stocks the next day for the whole period

Predicting Tesla's stock price

As another extension, we predict the Tesla Inc. stock price [5]. This dataset is different from the previous two since it starts in 2010 and we made the prediction up to April 2021. Figure 7 shows that the LSTM could fit very well the training set, while Figure 8 clearly shows the network could not predict the stock price at all due to overfitting to values lower than around 200. Moreover, the COVID-19 era, coupled with the rising of Elon Musk's projects i.e. SpaceX and Gigafactory just to cite two, may be responsible for the skyrocketing of Tesla's stock prices.

Improving the Predictions

In this section, we show how to improve the algorithm in three ways:

1. Revising the codes with Pytorch Lightning
2. Using a different AI algorithm, namely Gated Recurrent Unit (GRU)
3. Improving the MSE error results

[5]Source: Yahoo finance

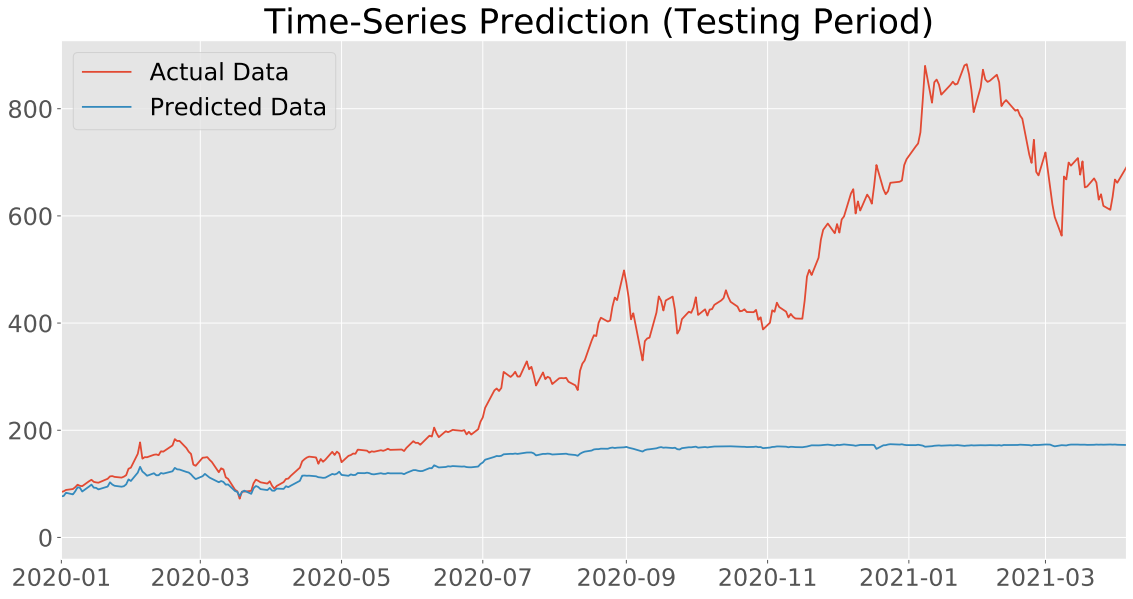

Figure 8: Actual and predicted adjusted closing price of Tesla's stocks the next day for the test period. The network is not able to predict well past 2020 due to the skyrocketing of stock prices which were unseen in the training set

Using Pytorch Lightning

We revise the code by using Pytorch Lightning [2]: this PyTorch framework provides a high-level interface by which it is easier to control architectures, results, logging and more. More importantly, it makes the process of moving models to GPU, Tensor Processing Units (TPUs) and even multiple GPUs and TPUs easier while having a negligible overhead. This open-source library is actively maintained by a community highly focused on efficiency and code readability. We refactor the code to be used with Pytorch Lightning.

The GRU model

Gated Recurrent Units (GRU) [1] were proposed as an RNN variant in 2014: they are similar to LSTMs without an output gate and has fewer parameters. Figure 9 shows an overview of the architecture. Having fewer parameters than LSTM, GRUs

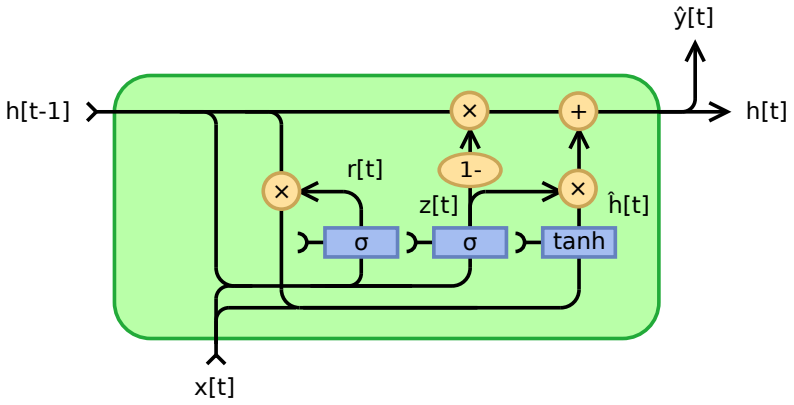

Figure 9: Gated Recurrent Unit (GRU) base model

have been shown to perform better on certain dataset and have better generalization capabilities with fewer data.

Predicting Samsung's stocks with GRU

We implement GRU on the same dataset as the one used with LSTM. Figure 10 shows that GRU successfully fits the training dataset and the predicted values are close to the actual ones.

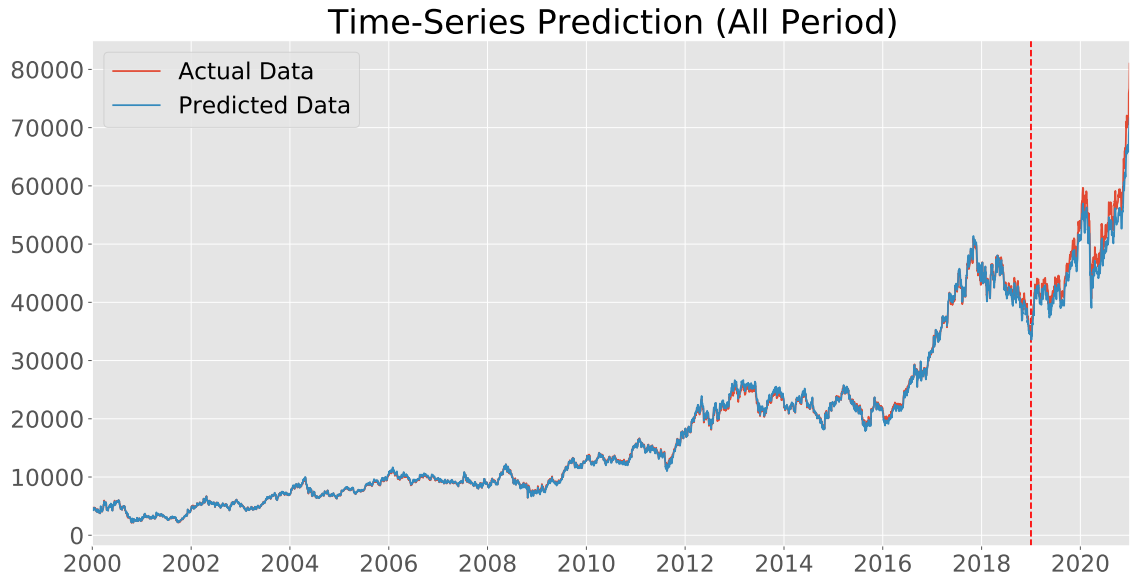

Figure 10: Actual and GRU-predicted adjusted closing price of Samsung's stocks the next day for the whole period

Comparing LSTM and GRU

Samsung's stock price

We show in the Table the results of the MSE, while Figure 11 we compare the results graphically.

	LSTM - MSE	GRU - MSE
Training Data	234,280.4	51,174.5
Test Data	18,589,588.0	5,497,948.0

As we can see, GRU is able to achieve better results than LSTM, which we confirmed with multiple experiments as well. As suggested in the seminal paper, this could be due to less parameters which guarantee better generalization properties when compared to LSTM; especially considering the fact that the dataset provided is not high-dimensional.

Tesla's stock price

We repeated the experiment of Tesla's stock prices prediction over 2020 onwards: this dataset is particularly difficult to predict due to the unseen behavior of stocks in the test set: even though LSTM achieved a good fit on the training data, it

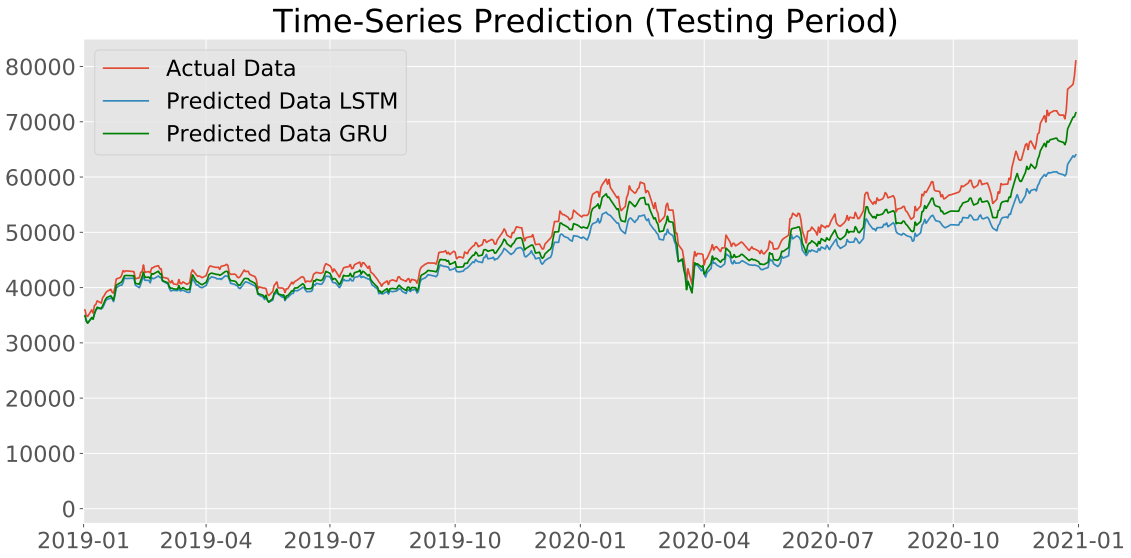

Figure 11: Actual, LSTM-predicted and GRU-predicted adjusted closing price of Samsung's stocks the next day for the test period. GRU has better generalization properties on this dataset.

could not generalize to unseen data. Figure 12 shows the comparisons of LSTM and GRU: while the former is not able to generalize to the unseen data and the solution diverges, GRU is able to consistently keep track of stocks behavior until around November 2020, while after that it still performs quantitatively better than LSTM.

References

[1] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[2] William Falcon et al. Pytorch lightning. *GitHub. Note: https://github.com/PyTorchLightning/pytorch-lightning*, 3, 2019.

[3] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[4] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Rai-

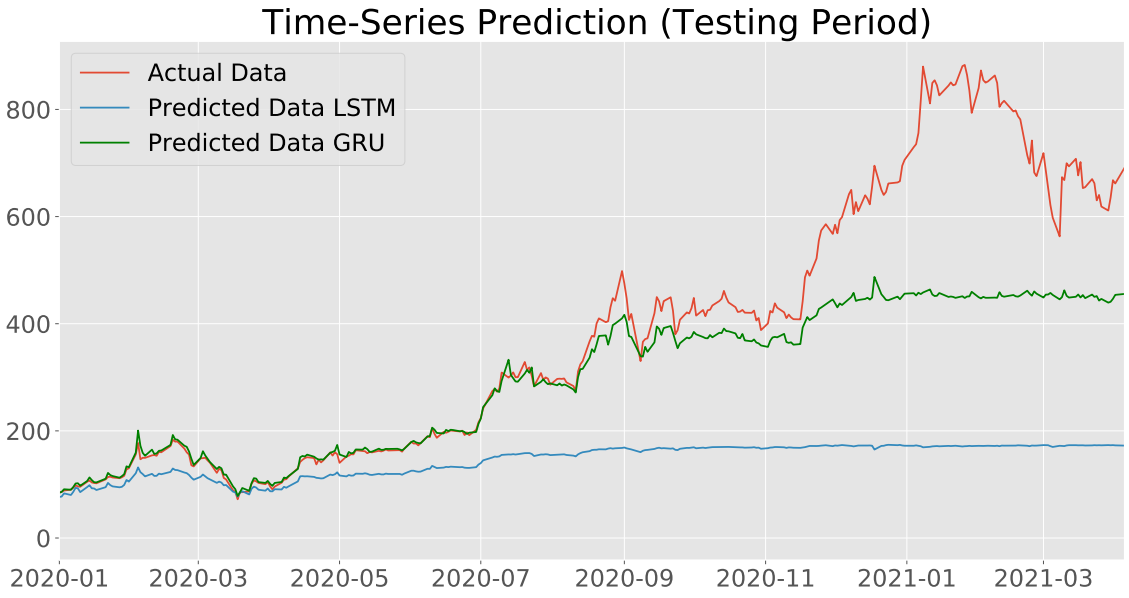

Figure 12: Actual, LSTM-predicted and GRU-predicted adjusted closing price of Tesla's stocks the next day for the test period. GRU has better generalization properties on this dataset, even though the test data differs greatly compared to the past observations due to 2020's skyrocketing of Tesla's stock prices.

son, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.