Jens Färm

Forecasting power of neural networks in cryptocurrency domain

Forecasting the prices of Bitcoin, Ethereum and Cardano with Gated Recurrent Unit and Long Short-Term Memory

-

School of Accounting and Finance
Master's thesis in Finance
Master's Programme in Finance

Vaasa 2022

UNIVERSITY OF VAASA
School of Accounting and Finance

Author:	Jens Färm		
Title of the thesis:	Forecasting power of neural networks in cryptocurrency domain : Forecasting the prices of Bitcoin, Ethereum and Cardano with Gated Recurrent Unit and Long Short-Term Memory		
Degree:	Master of Science in Economics and Business Administration		
Programme:	Master's Programme in Finance		
Supervisor:	Mikko Ranta		
Year:	2022	**Pages:**	44

ABSTRACT:

Machine learning has developed substantially during the past decades and more emphasis has gone to deeper machine learning methods, i.e., artificial neural networks, computer-based networks seeking to mimic how the human brain functions. The groundwork for ANN research was established already in the 1940s and the advancement of ANNs has been extensive. Price prediction of different financial assets is a broadly studied field, as researchers have been trying to create models to predict the volatile and noisy environment of financial markets. Also, ANNs have been placed for these hard prediction tasks, as their advantage is the ability to find non-linear patterns in uncertain and volatile setting.

Cryptocurrencies have made their way to the common audience in the past years. After Nakamoto (2008) presented the first proposal for an electronic cash system, Bitcoin, the number of different cryptocurrencies has exceeded over 8 000. Also, the market capitalization of all cryptocurrencies has grown rapidly, in November 2021 the aggregate market capitalization topped 3 000 billion U.S. dollars. Cryptocurrencies are not a small concept for closed groups of tech-people, but a phenomenon that concerns also in the governmental level.

This study utilizes recurrent neural networks, GRU and LSTM, in the prediction task regarding cryptocurrencies. In addition to trading data, this study uses Google trend-based popularity score to try to better the ANNs accuracy. In addition to the sole prediction task, the study compares the two used RNN architectures and presents the performance and accuracy with selected performance measures.

The results show that recurrent neural networks have potential in prediction tasks in the cryptocurrency domain. The constructed models were able to find coherent trends in the price fluctuations but the average differences on actual and predicted prices were comparatively high, with the introduced simple RNN models. On average, the LSTM model was able to predict the cryptocurrency prices more accurately, but the GRU model showed also great evidence of prediction accuracy in the domain. All in all, the cryptocurrency prediction task is a hard task due to its volatile nature, but his study shows great evidence for ANNs ability to predict cryptocurrency prices. Considering the findings, further research could be applied to more optimized and complex ANN models as the models used in the study were relatively simple one-layer models.

KEYWORDS: Machine learning, ANN, Cryptocurrencies, LSTM, GRU

Contents

Figures

Tables

Abbreviations

AI	Artificial Intelligence
ML	Machine learning
ANN	Artificial Neural Network
MLP	Multi-layer perceptron
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
ReLU	Rectified Linear Unit
LSTM	Long Short-Term Memory
CEC	Constant error carousel
GRU	Gated recurrent unit
MAD	Mean absolute deviate
MSE	Mean squared error
RMSE	Root mean squared error
BTC	Bitcoin
ETH	Ethereum

1 Introduction

We often hear that artificial neural networks are the next big thing in machine learning, but what is an artificial neural network and why are they so important? Neural networks are a subset of machine learning. Artificial neural networks are designed to simulate the human brain's ability to learn. They are also known as neural networks, connectionist networks or connectionist systems. The human brain is a network of neurons that process sensory information and transmit it to other parts of the brain, where it is further processed and used to make decisions. The artificial neural network model is a simplified model of the human brain. Instead of neurons, artificial neural networks have connections between the neurons, and they receive input data and produce output data. (Gao et al. 2020).

The development of machine learning algorithms, moreover artificial neural networks, has been fast during their lifetime, starting from the first research on the topic in the 1940s. The power and advantages of ANNs are beyond doubt. Their ability to detect nonlinear patterns in noisy environment has made them attractive for several different fields, e.g., medicine and finance.

The first cryptocurrency entered the world when Nakamoto (2008) presented his/her study for the first decentralized electronic cash, Bitcoin. After Nakamoto's study, the cryptocurrency market has flooded with other alike electronic currencies, the current number being over 8 000 according to Liu & Tsyvinski (2021). Also, a lot of development has happened in the cryptocurrency world, e.g., the hash rate has increased substantially, and the adaptation of cryptocurrencies has not been wider ever. As cryptocurrencies gain more ground, these will be more affected by the macro-economic and political environment. Also, as cryptocurrencies are becoming more popular, the governmental stance is developing to regulate the cryptocurrency market. (Schaupp & Festa 2018).

Prediction of financial asset prices and cryptocurrencies is a challenging task due to high volatility, noisy environment, and non-linearity. Several researchers and research groups have been trying to develop sufficient models to predict asset and cryptocurrency prices. For the past decade, machine learning models have been implemented in the domain as several ML approaches have had great empirical evidence on tasks regarding noisy environments and non-linearity. For example, Chang, Wang & Zhou (2012) created a partially neural network combined with a global search algorithm and achieved 97 percent prediction accuracy in stock price index prediction.

Moreover, researchers have been keen on quantifying the impact of online discussion and social media presence on asset and crypto prices. For example, Li, Fong, Rizik & Wu (2018), showed first academic proof that social signals, in their case Twitter sentiment, have a statistically significant effect on cryptocurrencies. Additionally, Garcia & Schweitzer (2022), found a relationship between price returns and signals through Twitter sentiment and Google search query data. Their findings clearly show that sentiment through online media and trading volumes influence the price of Bitcoin.

The motivation for the study derives from the previously mentioned aspects. For cryptocurrencies, where the intrinsic value is highly doubtable, models which can efficiently and accurately predict the price fluctuation given only the available information, in this case, trading and online sentiment information, would yield into interesting further research opportunities within the domain.

1.1 Purpose of the study

The main purpose of the study is to find out how well ANNs are suited in prediction tasks in the domain of cryptocurrencies. The sample of cryptocurrencies is chosen based on the aggregate market cap of the current supply, per 31.12.2021 (https://www.coinmarketcap.com). The three biggest cryptocurrencies are chosen to get a sufficient overview of the whole cryptocurrency market and further analyze the possible differences in prediction accuracy. Choosing the three biggest cryptocurrencies gives over 60 percent cover of the cryptocurrency market.

ANNs work well in several prediction tasks regarding the financial domain. In 2007, Roh studied so-called hybrid models, ANN combined with an economical model, e.g., GARCH-model, in stock index volatility prediction and found excellent prediction accuracy for forecasting period of less than a month, 100 percent accuracy for a 10-day period, and 85 percent accuracy for a 20-day period. Kim et al. (2018) extended the study with more novel LSTM architecture combined with the same GARCH-models and found superior accuracy compared to the traditional economical models.

Guresen, Kayakutlu & Daim (2011) studied ANNs, hybrid models, and chosen GARCH-models to predict the movement of the NASDAQ stock index. Their study showed that the traditional MLP-model performed better compared to the chosen control models DAN2 and GARCH-MLP. The traditional MLP-model realized 2,52 percent MAD, whereas other models realized a MAD of 2,78 to 6,49 percent. They found also, like others, that ANNs are exceptionally well suited to predict the direction of movement in different financial time-series prediction assignments.

Galeschcuk (2016) found out that ANNs, specifically simple three-layer ANNs, work well predicting the exchange rate fluctuations in the short-term daily intervals. Galeschcuk realized an average relative prediction error of 0,2 to 0,3 percent with three different exchange rates, EUR/USD, GBP/USD, and USD/JPY. In the longer horizon, the simple three-layer ANN lost slightly its prediction accuracy and realized 1,9 to 3,5 percent prediction errors in quarterly prediction.

Additionally, there have been studies that have found empirical evidence of ANN prediction power on cryptocurrencies. Alessandretti, El Bahrawy, Aiello & Baronchelli (2018) created gradient boosting trees and LSTM-based models to predict the daily prices of different cryptocurrencies. Based on the prediction of the models, they created simple trading strategies. They found that gradient boosting trees and LSTM-based models were able to predict the daily prices of cryptocurrencies exceptionally well and create significant positive returns with different cryptocurrencies. Also, Yamak, Yiujan & Gadosey (2019) and Patel, Tanwar, Gupta & Kumar (2020) found empirical evidence of LSTM and GRU -based models' price prediction accuracy in the cryptocurrency domain.

Despite Alessandretti et al.'s (2019) study, the utilization of ANNs, especially more advanced and deeper ANNs, in the domain of cryptocurrencies is rather limited. Therefore, this study aims to provide insight into time series prediction accuracy of ANNs within cryptocurrencies. In addition to the prediction of cryptocurrency prices, the study compares two different ANN models in the domain, LSTM, and GRU, as they both have different advantages compared to each other. Moreover, the study will incorporate Google search query-based popularity variable together with the common trading variables, trying to expand the prediction power of the models.

1.2 Hypotheses

Referencing the previous studies where ANNs have been applied in different domains, there is evidence that ANNs have great prediction accuracy. Also, when compared to traditional statistical methods the accuracy is superior, on several occasions. Therefore, the first hypothesis on the study is as follows:

H1: *Neural networks can predict the daily prices of cryptocurrencies with sufficient accuracy*

In addition to the sole accuracy, the difference in prediction accuracy within two different ANNs will be studied. LSTM-models have experienced great empirical evidence on time-series prediction. For example, Karmiani, Kazi, Nambisan, Shah & Kamble (2019) found that LSTM-model outperforms other ANN-based models in the financial domain. Also, the LSTM structure allows the model to use more information in the decision-making, compared to GRU. The second hypothesis is formulated as follows:

H2: *LSTM-model is more accurate in predicting the daily prices of cryptocurrencies, compared to GRU*

To conclude, this study aims to shed light on how well ANNs can predict the price movements of cryptocurrencies. Additionally, the intent is to discover which types of ANN models are most efficient in predicting the price development of cryptocurrencies.

1.3 Structure of the study

The structure of the thesis is the following. The first chapter will introduce the topic and related subject, with an initiation to a few studies regarding the matter. Additionally, the first chapter will disclose the purpose, hypotheses, and structure of the study. Following this, the second chapter will lay out the theoretical background for the study. The first part of the chapter will focus on cryptocurrencies, which will work as the domain for the empirical part of the study. The second part of the chapter will focus on ANNs, laying the background and fundamentals for the reader.

The third chapter discloses the data collected for the study and the methodologies used in forecasting the daily prices of selected cryptocurrencies. The fourth chapter will introduce the empirical results of the study, with selected accuracy measures, and discuss broadly about the results. The study will conclude with the fifth and last chapter, which summarizes the empirical results and evidence with further research topics where to extend the study.

2 Theoretical background

The purpose of the section is to present the theoretical background for the study, cryptocurrencies, and ANNs. The first sub-chapter will go through the basics of cryptocurrencies, focusing on birth of the cryptocurrencies and the cryptocurrency market. The following sub-chapter will present the background and fundamentals of ANNs. In the chapter, the setting for ANNs is presented, following with the fundamental of ANNs and their architecture. The chapter will present more profoundly two recurrent neural networks, LSTM and GRU as these are used in the empirical part of the study.

2.1 Cryptocurrencies

This chapter will shed light on the birth of Bitcoin, the first cryptocurrency, familiarize the concept of cryptocurrencies, and present the main advantages of cryptocurrencies. The chapter will close to a low-level overview of the pricing of cryptocurrencies, which is at the center of this study.

The first proposition for blockchain-based cryptocurrency was done by Nakamoto (2008) when he/she released his/her paper *“Bitcoin: A Peer-to-Peer Electronic Cash System”*. The key thesis for electronic cash, i.e., cryptocurrency, was to eliminate the need for the third party, i.e., a bank or other financial institution, to verify the transactions.

13 years later, there are over 8 000 individual cryptocurrencies to date, 1 707 individual cryptocurrencies with a market capitalization of over one million USD (Liu & Tsyvinski, 2021). During the lifetime of cryptocurrencies and during the past years, a vast amount of these has been created with malicious intentions. As, for example, Schaupp & Festa (2018) pointed out, cryptocurrencies are on a tight spot regarding governmental stance, so many have introduced the idea of a need for governmental regulation. Yet, this stance battles with the main rationale behind Bitcoin’s and other cryptocurrencies’ being decentralized peer-to-peer verified system without any third-party involvement.

Bitcoin and other cryptocurrencies are based on a so-called distributed public ledger. The transactions are gathered and verified on the blockchain through digital signatures. Figure 1 shows the structure of the chain of digital signatures. The structure is based on digital signatures, i.e., public, and private keys. Firstly, when a unit of a cryptocurrency is moved, a new hash is formed with confirmation of the following owner. This block, or a transaction, is verified with the public key of the previous owner and signed with the private key of the same owner. Therefore, the new block consists of the hash, public key of the current owner, and private key of the previous owner.

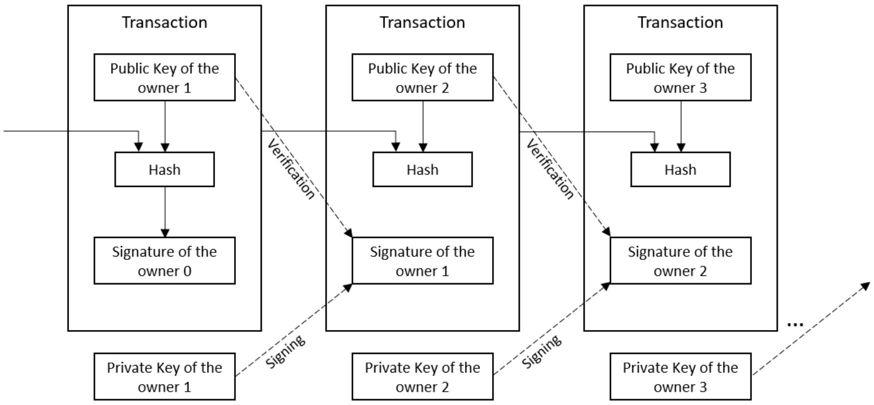

Figure 1. Transactions in cryptocurrency infrastructure (Nakamoto 2008).

Through Nakamoto (2008), Hanl (2018) and Corbet, Meegan, Larkin, Lucey & Yarovaya (2018) studies, four main advantages of cryptocurrencies can be detected. These are elimination of third-party verification, seamless and anonymous transactions, security, and accessibility. Cryptocurrencies are based on a decentralized system where the third party is the whole community of so-called miners. These miners verify the transactions based on public and private keys provided by the counterparties within the transactions.

When people are transferring cryptocurrencies through their digital wallets, cryptocurrencies and the transactions are available despite the time or place of the individuals. Also, the public and private keys or wallets are not linked to the individuals so the transactions can be done anonymously and cannot be tracked into any individual. Additionally, digital wallets can be accessed if one has only an internet connection and user interface (i.e., mobile phone, tablet, computer). Lastly, the blockchains aggregated through the transactions are always visible to anyone and cannot be retracted after the verification. (Nakamoto, 2008; Hanl, 2018; Corbet et al., 2018).

Also, the pricing of cryptocurrencies has been widely speculated during the lifetime of cryptocurrencies and more during the past five years. Mainly the cryptocurrencies pricing debate encounters the statement of cryptocurrencies having no intrinsic value. One of the earliest studies on cryptocurrency pricing is by Kristoufek (2015). Kristoufek used a continuous wavelet framework to examine the relationship between Bitcoin price movement and different parameters, e.g., technical drivers and search engine queries. The study found that the common fundamental factors (trade usage, price level) in fact have an effect on the price of Bitcoin in the long run. Also, it was found that the overall interest in Bitcoin was a key determinant of the price.

2.2 Artificial neural networks

This chapter will unveil the background and fundamentals of artificial neural networks. Moreover, the chapter will shed light on the basic structure of ANNs and present the most common types of neural networks. Later in the chapter, the two models used in the empirical section of the paper are introduced more deeply.

2.2.1 Background

In the context of machine learning, ANNs are a sub-category of deep learning. ANNs are made to resemble how the human brain functions. The first form of ANNs was already formulated during the 19th-century by a group of researchers, which study was based on McCulloch and Pitts' study from the 1940s. ANNs caught interest due to their ability to learn, respond, and capability to mimic human brain behavior. (Zilouchian & Jamshidi, 2001). As the human brain, ANNs take an input, process the information within the hidden layers, and then yield an output through the output layer.

In the later decades, the ANNs, their architectures, and applications have developed significantly. For example, already in the 1980s the first multi-layer ANN architectures were introduced but gained more ground in the early 1990s. (Schmidhuber 2014).

Table 1. Development of the research of ANNs (Schmidhuber 2014).

Date	Subject	Authors
1940s	Earliest studies on ANNs	McCulloch & Pitts, 1943
1950s to 1970s	Simple ANNs with supervised learning	Rosenblatt, 1958
		Widrow & Hoff, 1962
		Narendra & Thathatchar, 1974
1960s to 1980s	Simple ANNs with unsupervised learning	Grossberg, 1969
		Kohonen, 1972
		Hopfield, 1982
1960s to 1980s	Development of backpropagation	Bryson, 1961
		Linnainmaa; 1976
		Rumerhalt et al., 1986
1980s to 2000s	General improvements in ANNs	Hecht-Nielsen, 1989
		Elman, 1990
		Jaeger, 2001
2000s to 2010s	Further improvements in ANNs, especially Deep ANNs (RNNs and CNNs)	Hochreiter & Schmidhuber 1997
		Neal 2006
		Cho et al., 2013

Table 1 presents the development of ANN research from the early days in the 1940s to this date. The first studies on ANNs are dated in the 1940s when McCulloch & Pitts published their paper *"A logical calculus of the ideas immanent in nervous activity"*, these were not ANNs with the ability to learn. Although, this laid the groundwork for the further development of ANNs. A few years after, Hebb 1949, published a paper regarding unsupervised learning, which can be regarded as the first solid ANN. (Schmidhuber, 2014).

Through the 1950s to 1970s development focused on simple supervised learning ANNs by, e.g., Rosenblatt (1958) and Narendra & Thathatchar (1974). In the 1960s to 1980 the development focused on unsupervised learning and a concept of backpropagation was presented. Regarding unsupervised learning Grossberg (1969), studied general structures of unsupervised learning, and Kohonen (1972) focused on data classification. The concept of backpropagation, propagation of the loss backward to the system to optimize the error, was first mentioned in the 1960s by, e.g., Kelley (1960) and Bryson (1961), and it developed through the 1970s and 1980s. The concept of backpropagation improved the accuracy and prediction power of ANNs but as it made the architecture more complex, the computational power needed to train the ANNs increased significantly. (Schmidhuber, 2014).

The 1980s to 2000s were a time for more general development in the ANN space. For example, Hecht-Nielsen (1989) found evidence that simple three-layer ANNs are sufficient enough to estimate continuous multivariate function with great accuracy. Also, there was great development in RNN architectures by Elman (1990) and Jaeger (2001). Through the 2000s and 2010s, the development focused on deeper ANNs. For example, a study by Neal in 2006, on Bayesian ANN and Cho et al. 2013 study on stochastic gradient descent in 2013. Also, Kalchbrenner et al. (2015) introduced more novel architectures of LSTM, a complex RNN.

2.2.2 Fundamentals

Three main types of ANNs are simple feed-forward neural networks, convolutional neural networks (CNN), and recurrent neural networks (RNN). As mentioned, the following chapter will go through the fundamentals of ANNs and briefly all the mentioned types of ANNs. Additionally, two ANN models GRU and LSTM, in the RNN family, are presented more profoundly as these are used in the empirical part of the study.

In figure 2 below, the basic structure of a neural network is shown as divided into layers. A simple one-layer feed-forward neural network has one input layer, one hidden layer, and one output layer. In more complex neural networks, there can be several hidden layers. As depicted in figure 2, the three different layers are fully interconnected, each input neuron is connected to each hidden layer neuron, and each hidden layer neuron is connected to the output layers neurons. (Zilouchian & Jamshidi, 2001; Thakur, 2021).

Firstly, the input layer gathers the information and feeds it to the hidden layer. Given there is only one hidden layer, the hidden layer processes the information and feeds it to the output layer. The output layer interprets the information from the hidden layer and indicates the result of the model. The links between the different nodes which are described as lines in the figure, pass the information through different layers and work like synapses on the human brain. (Zilouchian & Jamshidi, 2001; Thakur, 2021).

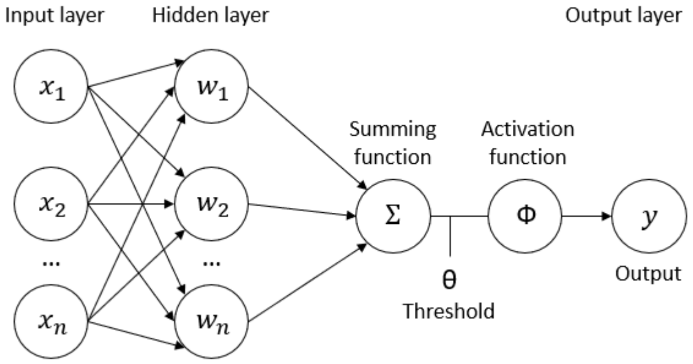

Figure 2. Basic structure of an artificial neural network – simple feed-forward.

In more detail, every ANN, with few exceptions, include at least the following five elements:

1. Input for the perceptron/synapsis

2. Synaptic weight
3. Summing function
4. Activation function
5. Threshold
6. Output

As mentioned in the previous paragraph, the common structure starts with the input layer with n inputs ($x_1, x_2 \ldots\ x_n$). These inputs are fed to the ANN and each node in the hidden layer assigns a weight to the input variable, which can be regarded as an importance of the input to the output ($w_1, w_2 \ldots\ w_n$). (Thakur, 2021).

Before passing the outcome to the activation function, the outcome is compared to the chosen threshold, θ. The threshold determines if the given neuron's outcome will be considered. Simply, if equation: $\sum_i w_i x_i > \theta$, holds true, the neuron's outcome will be considered for the output. (Rosenblatt, 1958).

After the threshold is considered, the result of the dot-product, given that w_i and x_i are vectors, the outcome is passed to the chosen activation function. The activation function scales the result of the dot-product to the desired output. The activation function can be binary, linear, or non-linear. Binary activation functions give the outcome a value if a given limit is reached. The below equation is an example of a binary activation function, which uses 0 as a limit. (Ramachandran, Zoph & Le, 2017).:

$$f(x) = \begin{cases} 1, when\ x \geq 0 \\ 0, when\ x < 0 \end{cases} \tag{1}$$

The second type of activation function is linear. In a linear activation function, the outcome of the function is directly proportional to the result of the hidden layer. The below equation is an example of a linear activation function. (Ramachandran et al., 2017).:

$$f(x) = ax \tag{2}$$

Thirdly, there are also non-linear activation functions, e.g., sigmoid, tanh, ReLU. These have been designed to work the best on different types of tasks. The below equation is an example of a non-linear sigmoid activation function. (Ramachandran et al., 2017).:

(3) $$f(x) = \frac{1}{1+ e^{-ax}}, 0 \leq f(x) \leq 1$$

The sigmoid function is one of the most used activation functions. As can be seen from the equation, the sigmoid function transforms the values between the range 0 and 1.

The common way to describe ANNs is as they resemble how the human brain works and they have the capacity to solve problems in a human-like manner. ANNs have a few common characteristics that make them great for resembling human-brain function and how to perform complex tasks. The common characteristics are listed below. (Thakur, 2021).

1. Adaptive learning
2. Fault tolerance
3. Prognosis
4. Self-organization

Adaptive learning refers to the ANNs ability to develop the accuracy of recognizing the patterns and prediction in the model. This yields, for example, to better model structures and the overall performance of the model. Fault tolerance refers to the ANNs ability to tolerate corrupted neuron results without affecting the generation of the output of the model. This has also a downside as fault tolerance can affect negatively in prediction accuracy. Prognosis refers to the predictive power of ANNs and self-organization to the ability to compile models with vast amounts of data. (Thakur, 2021). The main disadvantages of ANNs are listed below:

1. Unexplained functionality of the network

2. Hardware dependence
3. Uncertainty of the sufficient model structure
4. Duration

One of the most considerable disadvantages in ANNs is the unexplained functionality of the network. In more simple statistical models, one can identify how different parameters contribute to the output. Given that ANNs work as a black box, one cannot identify the why's and how's the model got to the output. This reduces trust in the network. In real life, this disadvantage has emerged, for example, in banking regulation and EU GDPR regulation. E.g., banks cannot base their assessment of individuals' or institutions' creditworthiness solely on ANN-based models as they are required to be able to disclose the basis of their decisions. (Zilouchian & Hamshidi 2001; Thakur, 2021).

Secondly, hardware dependence refers to the ANNs need for processors and parallel processing power. Also, as one creates an ANN model the model structure is chosen. During the learning, one can test different models and model parameters but there is no particular rule on how to choose the best one for the given task. Lastly, as ANNs require vast amounts of data to train on the model, it may require an excessive amount of time to process the data and train the model. Nowadays, the lastly mentioned disadvantage has lost slightly its context as the processing power of computers has increased substantially. (Zilouchian & Hamshidi 2001; Thakur, 2021).

Alongside the few disadvantages, there is one problem that may occur for less practiced users. Chang et al. (2012), explains over-fitting as the ANNs ability to predict output exceptionally well with the training data but lacks the same accuracy on the test data. Over-fitting may occur more likely when the complexity of the ANN is increased. For example, Chang et al. 2012., when studying evolving partially connected neural networks, clearly found that the increase in complexity of the network (e.g., number of neurons and layers) does not necessarily yield to better results, as their prediction accuracy fell over 10 percentage points, from 97,58 percent to 86,29 percent, when one additional hidden layer and double the number of neurons was added.

In addition to simple feed-forward ANNs, two main classes of ANNs are convolutional neural networks and recurrent neural networks. Albawi, Mohammed & Saad (2017) elaborate that CNNs differ from simple ANNs as they have a convolutional layer in the architecture, the name convolutional comes from matrix calculus. CNNs use at least one convolutional layer in the structure, and, as in regular ANNs, each neuron is connected to every neuron in the following and previous layer. In addition to the convolutional layer, CNNs have another two additional layers, pooling layer, and non-linearity layer. As in regular ANNs, the convolutional layer and fully connected layers have adjustable parameters, but the pooling layer and non-linearity layers do not. This makes the CNN model faster to train and compile. CNNs have great empirical evidence in several machine learning problems, e.g., image classification and natural language processing. Figure 3. below shows the simplified structure of a CNN.

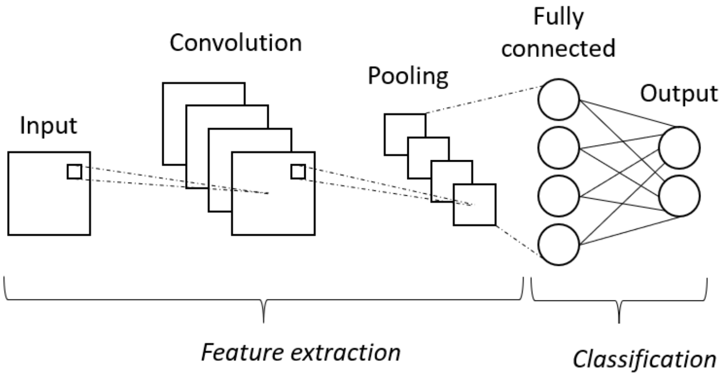

Figure 3. Simplified structure of CNN (Albawi et al. 2017).

RNNs are ANNs for processing sequences of data. As Medsker & Jain (2001) show, the main advantage of RNNs is dynamic layer processing, in RNNs the past or future information is represented with regular neurons, but with recurrent connections. Figure 4 shows the simplified structure in RNNs. The dashed lines represent information flow within a neuron itself or back to previous layer's neurons. RNNs work well in time series prediction due to the ability to learn non-linear dependencies on time series'. (Chen et al. 2021). Two more complex RNN architectures are gated recurrent unit and Long-Short-Term-Memory, which are used in the empirical part of the study and uncovered in the next section.

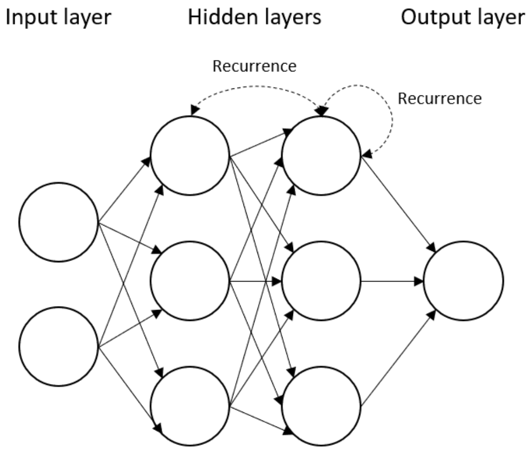

Figure 4. Simplified structure of RNN (Medsker & Jain 2001).

2.2.3 GRU & LSTM

GRUs introduces a slight modification to the regular RNN structure. Cho et al. (2014) introduced a novel structure to fix the vanishing gradient problem in RNNs. GRUs tackle the problem with so-called reset and update gates. The gates allow the network to decide what information is crucial for the network to process and what the network can forget. All in all, the network, including these gates, can be trained to hold on to the information for a longer period dynamically. Due to these features, GRUs are able to save and filter the information using the mentioned gate structures.

As mentioned, LSTM is slightly more complex in nature compared to the conventional RNN architectures. The first proposal of LSTMs was done by Hochreiter & Schmidhuber in 1997 when they elaborated on Hochreiter (1991) study on time consumption of recurrent backpropagation. Hochreiter & Schmidhuber (1997) presented the idea of constant error flow through CECs, constant error carousels. The greatest advancement was, in fact, in achieving a longer horizon of lags with faster learning time.

The LSTM architecture is composed to be able to detect longer potentially unknown dependencies in time-series data. (Chen et al. 2021). Due to this, the LSTM architecture is well suited for predicting, e.g., price development with non-linear and unknown cycles. The name of LSTM refers to the architecture's ability to detect patterns, in both, long-term and short-term. In recent years LSTM's have also seen development. For example, Kalchbrenner et al. (2015) made significant improvements on the LSTM architecture in their paper *"Grid Long Short-term Memory"*. They proposed addition of spatiotemporal dimensions of the data in addition to the regular cell connections between the network layers.

3 Data & Methodology

The chapter presents the data and methodology used for the empirical part of the study. The first part introduces the used data and the rationale for using the particular. The second part presents the methodology used in the study. Additionally, the chapter will introduce the used metrics for the model's performance evaluation.

3.1 Data

For the empirical part of the study, the three biggest cryptocurrencies by the aggregate market capitalization are used. The data set is set to include only cryptocurrencies which are not backed by any institutions (i.e., BNB is excluded as it is issued by the biggest cryptocurrency exchange Binance) or based on the value of another asset (i.e., Tether and USD Coin are excluded as their price is based on USD). Also, to have enough data, Solana is excluded as it was created less than two years ago. At the end of 2021, the three biggest cryptocurrencies were Bitcoin, Ethereum, and Cardano, considering the exclusions. The input parameters gathered for the neural networks are the common trading parameters: open, high, low, close, and volume. The trading data is gathered mainly from Yahoo Finance -database, but in the case of missing data, e.g., Yahoo Finance did not have data for the first 10 months of 2017 for Ethereum, the missing data is gathered from CoinMarketcap (https://www.coinmarketcap.com). The data is acquired in daily intervals. Also, search engine queries are used as one parameter for the model.

During the selection, a criterion for daily volume was incorporated. This is based on Alessandretti et al. (2019) study, and the chosen threshold is set as 100 000 USD. Although this study does not cover trading strategies where the daily volume is crucial for active trading, it is still used as sufficient trading volume contain information that would translate into the prices of which the ANN will try to find information. All the above cryptocurrencies met the criteria during the set period.

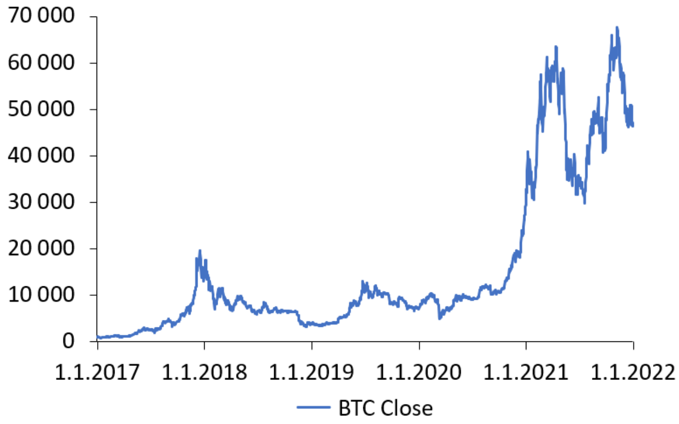

Figure 5. Bitcoin closing price development (Yahoo Finance, 2022).

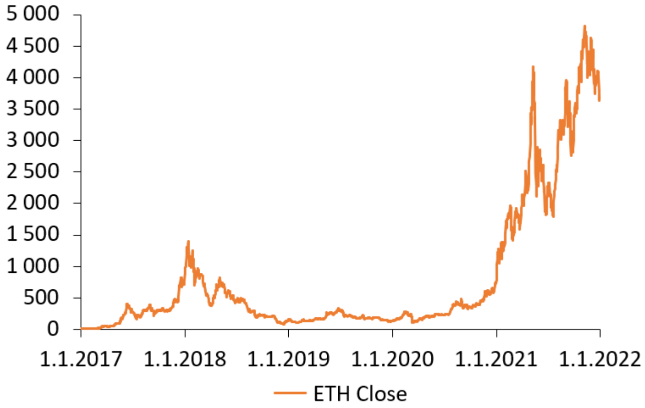

Figure 6. Ethereum closing price development (Yahoo Finance, 2022; Coinmarketcap, 2022)**.**

Figures 5, 6, and 7, show the closing price development for the chosen cryptocurrencies. As it can be seen, cryptocurrencies are heavily intercorrelated, at least in more intense bull and bear markets. The chosen period is rather interesting as it incorporates several great runs and falls for cryptocurrency price but also a more stable period.

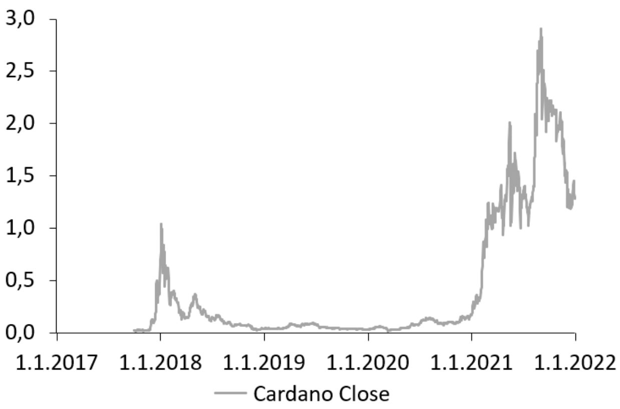

Figure 7. Cardano closing price development (Yahoo Finance, 2022; Coinmarketcap, 2022).

As mentioned, the search engine query data is used as a proxy for the popularity of the cryptocurrency. Google Trend data regarding the individual cryptocurrencies is used, as Kristoufek (2015) found that the popularity of Bitcoin is one of the determinants of the price development of the underlying cryptocurrency. This could be also the case in other cryptocurrencies. The Google Trend data can be fetched country-by-country or using the worldwide aggregate. In this study, the aggregated popularity is used as the cryptocurrency market is available almost regardless of the country. The popularity measure is derived from the Google search engine queries. The name of the individual cryptocurrencies is chosen as it is the most frequently used term regarding the different cryptocurrencies, compared to the acronyms, e.g., BTC, ETH. The normalized Google Trend data is available on weekly basis.

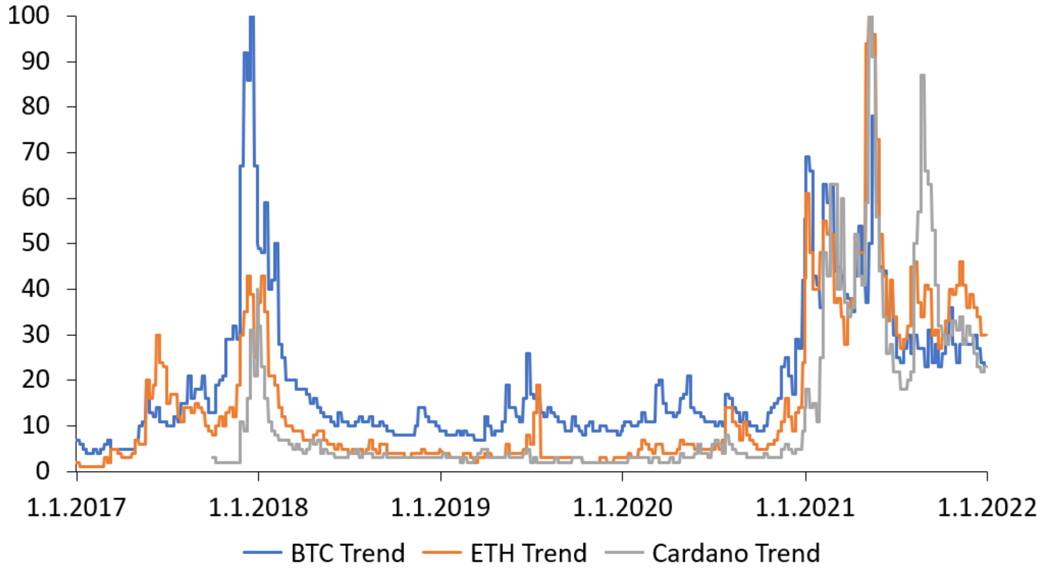

Figure 8. The development of the Google Trend popularity measure (Google, 2022).

Figure 8 shows the development of the popularity measure for the period. As it can be seen, the popularity of cryptocurrencies is also heavily correlated. Additionally, in some periods, it can be seen that the popularity of Ethereum lags slightly from the popularity of Bitcoin which could indicate that the popularity in Bitcoin drives the popularity of Ethereum. The trend development clearly shows the bull-cycle for Bitcoin, also other cryptocurrencies, at the end of 2017 and the bull-cycle in the midst of 2021, where the popularity of Ethereum was highest during the examination period.

The data is gathered from the beginning of 2017 until 2021. The period is chosen to get a sufficient amount of data to feed to the ANN, but also due to the highly increased popularity of cryptocurrencies during the time, especially after the start of Covid-19 03/2020. The data for Cardano is fetched from the beginning of October 2017 as it was not formed until then, therefore the data set is slightly smaller for Cardano. The interpretation of the used parameters is shown in table 2.

Table 2. Interpretation of the trading parameter.

Parameter	**Description**
Open	The opening trading price of the selected cryptocurrency
High	The highest traded price for the cryptocurrency given the day
Low	The lowest trade price for the cryptocurrency given the day
Close	The closing price of the selected cryptocurrency
Volume	The total USD value of trading done in the given day
Trend	Normalized trend-parameter for the popularity on Google searches for the given week

Firstly, the data is prepared and preprocessed. Table 3 presents the dataset sizes. Dataset for Bitcoin and Ethereum include 1 827 data points per parameter and for Cardano 1 552. This set is divided into a training set and a test set. The training set consists of 80 percent of the data points, 1462, and the testing set consists of the remaining 20 percent of the data points, 365. 1 241 and 311 respectively for Cardano.

Table 3. Summary of the dataset sizes.

	Training size	**Test size**	**Data set size**
Bitcoin	1 462	365	1 827
Ethereum	1462	365	1827
Cardano	1 241	311	1 552

As RNNs, especially LSTM, are scale sensitive, the data needs to be transformed into a common scale. Therefore, min-max scaling is used to scale the data into a range [0, 1]. (Witten et al., 2016).

(4) $$y_i = \frac{x_i - \min(x)}{\max(x) - \min(x)}$$

The y_i denotes the scaled value of x_i in data set x . $\max(x)$ denotes the maximum value in the data set, and $\min(x)$ the minimum value on the data set. From the equation, it can be seen that if the value x_i is out of the bound of $[\min(x)\,,\max(x)]$, the equation will not result a value in the bounds of [0, 1]. After the model has been trained and tested, the scaled values will be inverted back to the original scale to be more interpretable. (Witten et al., 2016).

3.2 Methodology

As the method for the study, two different recurrent neural networks are used: GRU and LSTM. These are chosen due to great results in time-series prediction in previous literature and to evaluate two different RNNs in this domain. The model construction is done in three steps:

1. Gathering and preprocessing of the data
2. Building and training the model
3. Evaluating the results

Firstly, the data is gathered from Yahoo Finance and Google Trends, and missing trading data for cryptocurrencies is gathered from Coinmarketcap. For the cryptocurrency trading data, the data from Yahoo Finance and Coinmarketcap are combined. As mentioned, the cryptocurrency data was acquired on daily intervals and the search engine trend data on weekly intervals. To combine the data sets, the weekly trend data is interpolated for the whole week, as the weekly data point are aggregated for the week. Following this, the data for each cryptocurrency is shaped uniformly.

The ANN models are done with Python and during the model construction, several different open-source libraries are used. For the preprocessing of the data Pandas and Numpy are used. Sklearn is used to obtain the performance measures and MinMaxScaler used in the model. For the model building, Tensorflow and Keras are used.

After the model has been trained and tested, Plotly and Matplotlib are used for visualization purposes.

Table 4. Summary of the parameters of the models.

Parameter	**Value**
Number of hidden layers	1
Number of neurons	36
Number of epochs	100
Loss method	MSE
Validation split	0,25
Optimizer	Adam
Activation function	Linear

Firstly, the data set is divided into the training set and test set as mentioned in the previous chapter. After the division, both GRU and LSTM-models are built and compiled. Table 4 presents the parameters used in the models. The models are simple one hidden layer ANNs with 36 neurons in the hidden layer. The loss method used is mean squared error and a linear activation function is used.

After the model building, the testing set is fed to the constructed and trained model and the results are visualized. Lastly, the results will be evaluated with two different accuracy parameters root mean squared error and relative form of the mean absolute deviate. (Guresen et al. 2011).

Below is illustrated the formulas for the performance measures:

(5) $$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{k}^{n} (r_k - y_k)^2}$$

(6) $$\text{MAD-\%} = = \frac{1}{n} \sum_{k}^{n} \frac{|r_k - y_k|}{y_k} \; x \; 100$$

The above equations r_k refers to the obtained prediction from the model for kth observation, y_k is the actual value for kth observation and n is the size of the dataset, number of observations. (Guresen et al. 2011).

4 Empirical results

This chapter will present the empirical results of the study. The results for the chosen cryptocurrencies, Bitcoin, Ethereum, and Cardano, are presented in their own sub-chapter. Every sub-chapter will go through the results, relevant insights, show the comparison of the predicted and actual prices and present the performance measures used in the study.

4.1 Bitcoin

Table 5 shows the results for Bitcoin, using both GRU and LSTM. Overall, the neural networks show great potential in Bitcoin price prediction. The chosen models experience 3.76 and 4.85 MAD-%. The LSTM-model is able to predict the Bitcoin prices more accurately, with less than four percent average deviation from the actual price. As previous studies and logic suggest, the GRU model was faster in terms of computational time, although with worse accuracy.

Table 5. Summary of the results for Bitcoin.

	Bitcoin	
# of obs. (train/test)	1 461/365	
	GRU	LSTM
RMSE (USD)	2 855.1	2 188.5
MAD-% (%)	4.85	3.76
Time (sec)	6.65	8.36

As the below figures show, both models were able to capture the clear trends in the prices. Although, from both models, one can see that the predictive power diminishes in the latter half of the prediction period, e.g., after 200 days of prediction.

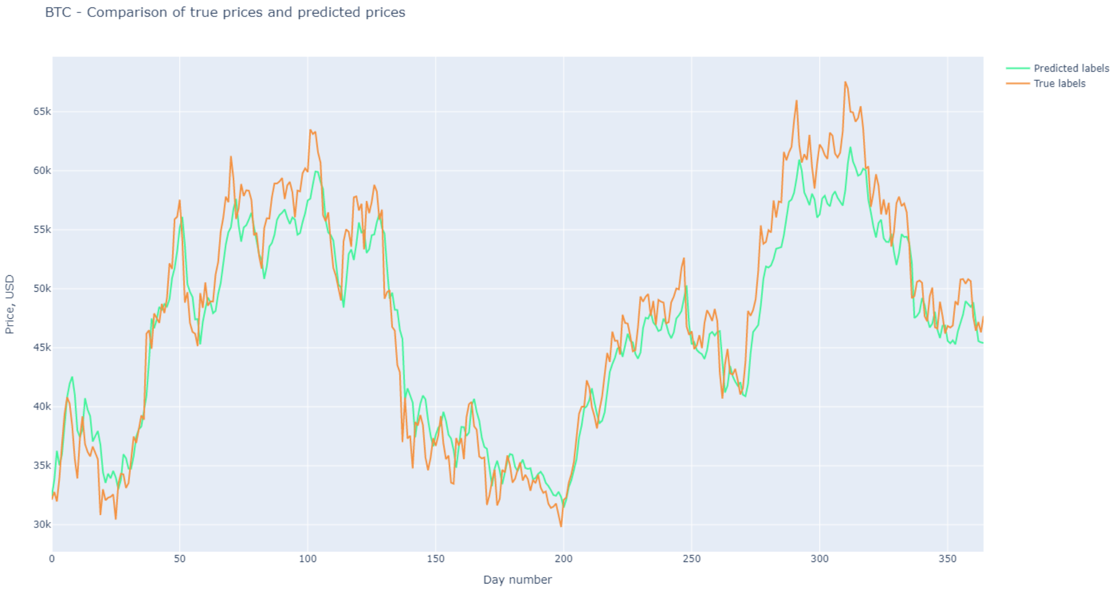

Figure 9. Comparison of the actual and predicted prices of Bitcoin - GRU.

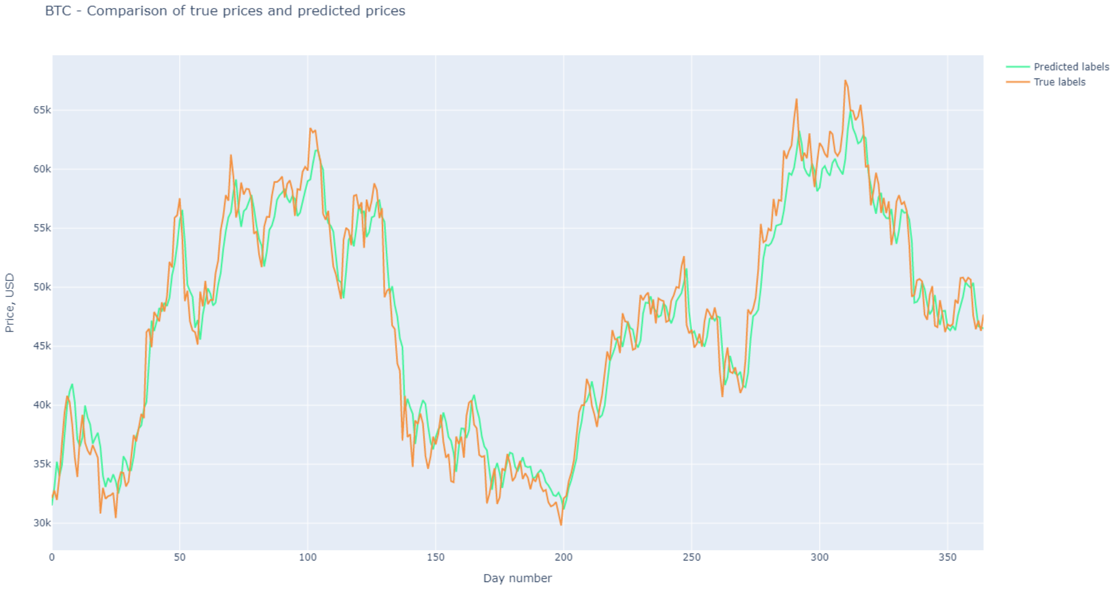

Figure 10.Comparison of the actual and predicted prices of Bitcoin - LSTM.

4.2 Ethereum

For Ethereum, the results are in line with Bitcoin. The models experienced better accuracy with LSTM than with GRU, with 4.76 MAD-% for GRU and almost 9 percent better MAD-%, 4.34, for LSTM. Comparing the computational times, the GRU model was clearly faster, as expected.

Table 6. Summary of the results for Ethereum.

	Ethereum	
# of obs. (train/test)	1 461/365	
	GRU	LSTM
RMSE (USD)	170.24	157.37
MAD-% (%)	4.76	4.34
Time (sec)	5.98	9.4

Through the below figures, it is clear that the LSTM had better prediction accuracy compared to GRU. Also, as in the case of Bitcoin, it can be stated that the prediction accuracy decreases in the latter half of the prediction period for Ethereum also.

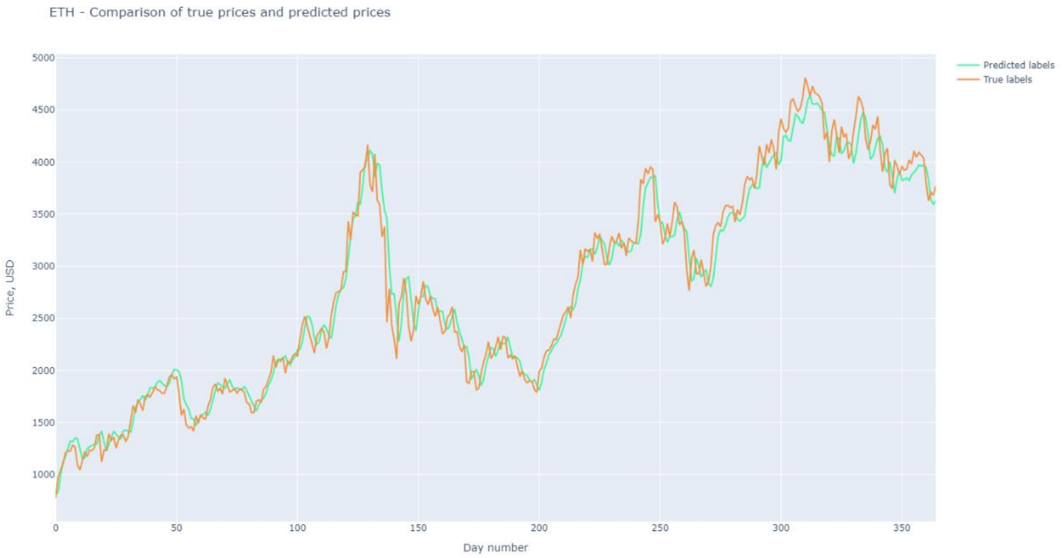

Figure 11. Comparison of the actual and predicted prices of Ethereum - GRU.

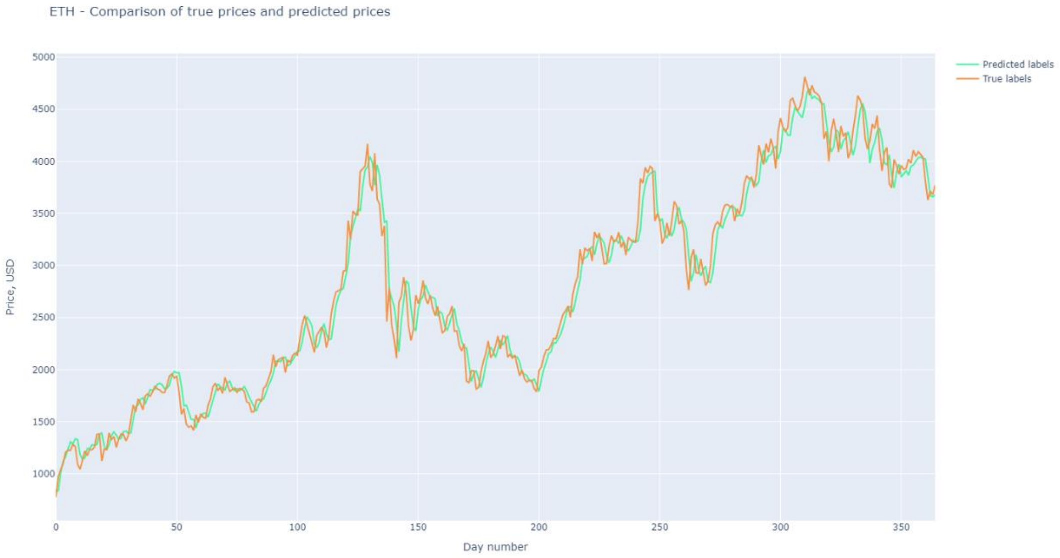

Figure 12. Comparison of the actual and predicted prices of Ethereum - LSTM.

4.3 Cardano

Contrarily to the previous cases, Cardano experienced better accuracy with GRU than with LSTM. The MAD percentages were 4.1 for GRU and 5.28 for LSTM. The prediction accuracy for Cardano with GRU was the second most accurate of all, which is an interesting finding. Also, the computational time using GRU was faster, as expected.

Table 7. Summary of the results for Cardano.

	Cardano	
# of obs. (train/test)	1 241/311	
	GRU	LSTM
RMSE (USD)	0.104	0.135
MAD-% (%)	4.1	5.28
Time (sec)	7.33	7.66

The below figures, figure 12 and figure 13, show the predicted and actual prices for Cardano. In the case of Cardano, the decrease in accuracy is not as well seen as in the case of Bitcoin or Ethereum. Overall, the ANNs were again able to detect the clear trend in the price fluctuation but the exact predictions were slightly off.

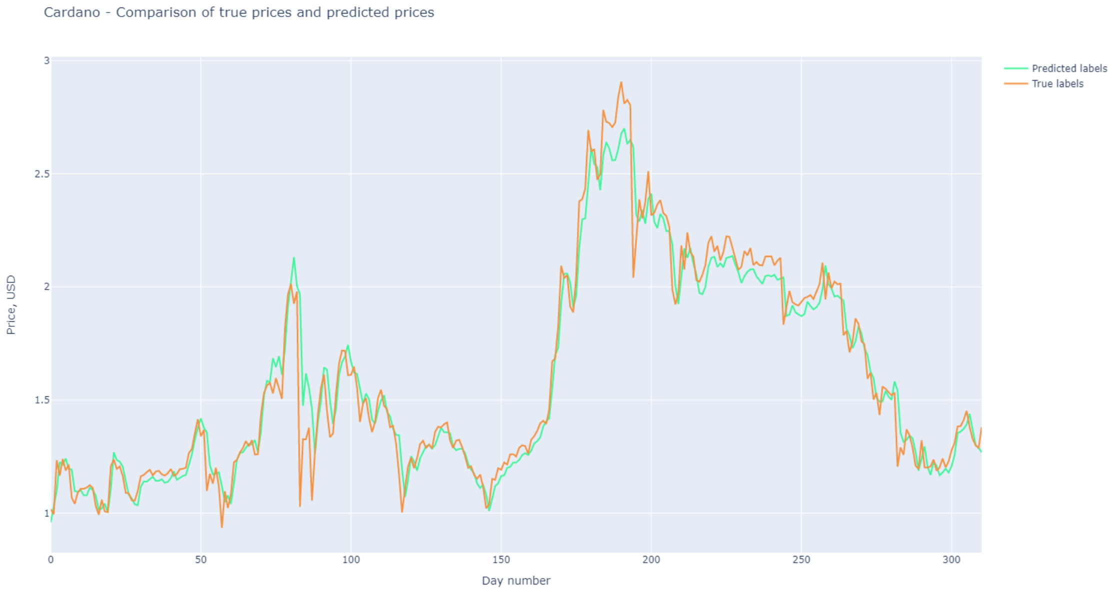

Figure 13. Comparison of the actual and predicted prices of Cardano - GRU.

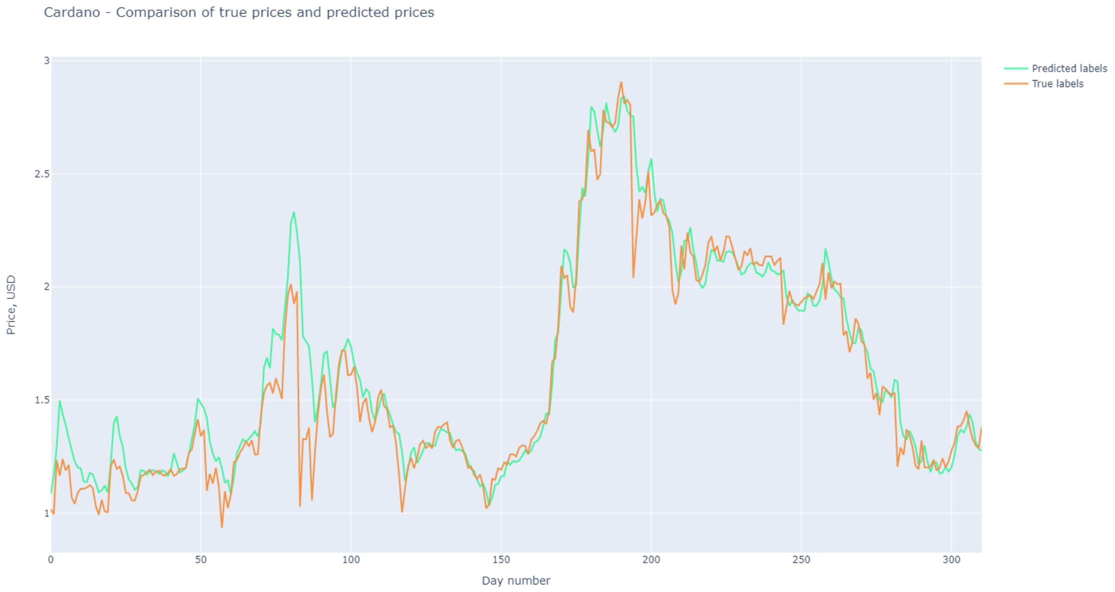

Figure 14. Comparison of the actual and predicted prices of Cardano - LSTM.

4.4 Discussion

The study's aim was to implement two RNN architectures on the cryptocurrency price prediction tasks. More detailed, GRU and LSTM architecture, were implemented on the three most popular cryptocurrencies by market size, Bitcoin, Ethereum, and Cardano. Moreover, the aim was to analyze the predictive power of ANNs and additionally compare the two mentioned RNN architectures among themselves. The performance measures used to compare the results and models were RMSE and relative MAD.

The below table summarizes the results for all cryptocurrencies. All in all, it can be stated that the simple ANNs, with given only trading information and Google search query data, perform relatively well in the price prediction task. The best achieved MAD percentages were 3.76, 4.1, and 4.34, which shows that the ANNs were able to detect the trend patterns well, but sufficient accuracy in the exact daily prices was not achieved.

Table 8. Summary of the results.

	Bitcoin		**Ethereum**		**Cardano**	
# of obs. (train/test)	1 461/365		1 461/365		1 241/311	
	GRU	LSTM	GRU	LSTM	GRU	LSTM
RMSE (USD)	2 855.1	2 188.5	170.24	157.37	0.104	0.135
MAD-% (%)	4.85	3.76	4.76	4.34	4.10	5.28
Time (sec)	6.65	8.36	5.98	9.4	7.33	7.66

In the more popular cryptocurrencies, Bitcoin and Ethereum, the LSTM-model was more accurate in the prediction task. Contrarily, when predicting the daily prices for Cardano, the GRU model performed clearly better. Therefore, the results are not unanimous. Although, on average, the LSTM-model performed better compared to GRU.

As the study utilized only a relatively small amount of data, the computational times were not in the key position. Still, it can be seen that LSTM required more time to train and compile the model. From figure 10 can be seen that when the Bitcoins price rose to levels above 50 000 USD, the accuracy of the model decreased clearly. This is probably since during the training period Bitcoin did not experience these price levels, i.e., the model had not been trained to these situations. The same can be seen from figure 12 when Ethereum's prices were above 3 500 USD levels.

The results of the study do not support fully the first hypothesis of the study: *"Neural networks can predict the daily prices of cryptocurrencies with sufficient accuracy"*, as the average deviation from the prices was fairly high, 3.76 to 4.34 for the best performing models. Although, there is clear evidence that artificial neural networks do in fact have potential in predicting cryptocurrency prices. As can be seen from figures 9 to 14, it is clear that the models can detect the overall trends in the price fluctuations, and with more detailed analysis and modifications of the hyperparameters, more accurate predictions can be achieved. As mentioned, these RNNs were simple one-layer models.

Considering the second hypothesis, the results are not unanimous. The LSTM-model performed clearly better for more popular cryptocurrencies Bitcoin and Ethereum than the GRU model. Despite that, the GRU model performed better for Cardano. All in all, the LSTM models achieved better prediction accuracy on average, which supports the second hypothesis clearly.

5 Conclusions

The purpose of the study was to find out how well ANNs are suited in prediction tasks regarding cryptocurrencies. There are vast amounts of previous literature regarding financial time-series prediction, although cryptocurrencies have not been studied that extensively. This study contributes to the existing literature by assessing the forecasting power of recurrent neural networks in the cryptocurrency domain. Additionally, the study examines the differences between a few RNN architectures.

The first hypothesis of the study was not fully supported by the results. Both, LSTM- and GRU-model were able to detect rather firmly the trends in the cryptocurrency price fluctuations. The experienced relative deviations from the true price were still over 3 percent, so it cannot be stated that neural networks can predict the daily prices of cryptocurrencies with sufficient accuracy. Therefore, it is fair to say that RNN architectures have great potential in the price prediction tasks, but the models need to be more optimized for the task compared to the models used in this study. The used models were simple one-layer models and optimization for the hyperparameter, e.g., the number of neurons or hidden layers was not done.

The second hypothesis of the study concerned the differences between GRU and LSTM. The second hypothesis had better support from the results. On average, the LSTM-model was able to predict the prices more accurately. Although when predicting the prices of Cardano, the GRU model performed better. The case regarding Cardano is also supported by Yamak et al. 2019, who found that GRU model performed better in time-series prediction than LSTM.

The first limiting factor in the study is the used data. This attains from the characteristics of cryptocurrencies. As cryptocurrencies are not based on fundamentals, e.g., dividends or other cash flows, there is limited data to be used in the prediction tasks regarding cryptocurrencies. Also, the models used in the study could be extrapolated more. This study used simple one hidden layer RNNs and the hyperparameters were not optimized for the tasks. Introducing more complex models or optimizing the hyperparameters could lead to better prediction accuracy if the computational time is not restricted.

The further research possibilities are boundless and highly linked with the previously mentioned limitations of the study. The study clearly underlines the RNNs potential in cryptocurrency price and other time-series prediction. Next, it would be interesting to advance the used ANN models and see hybrid models in the domain of cryptocurrencies.

The idea of ANNs was briefly explained in the introduction by the following paragraph: *"We often hear that artificial neural networks are the next big thing in machine learning, but what is an artificial neural network and why are they so important? Neural networks are a subset of machine learning. Artificial neural networks are designed to simulate the human brain's ability to learn. They are also known as neural networks, connectionist networks or connectionist systems. The human brain is a network of neurons that process sensory information and transmit it to other parts of the brain, where it is further processed and used to make decisions. The artificial neural network model is a simplified model of the human brain. Instead of neurons, artificial neural networks have connections between the neurons, and they receive input data and produce output data.".* This paragraph was in fact formed by an AI called GPT-J when asked a question "What are artificial neural networks". The GPT-J is constructed by Gao et al. in 2020. The mentioned AI has been taught with over 800 gigabytes of data including but not limited to, e.g., medicine, economics, and history, and works as a "Google" through neural network structure. Today, the machines are rather efficient and the current level of development of ANNs is already rather broad. Next thing is to wait for what happens in the future. Who is right, Jack Ma or Elon Musk?

References

Albawi, S., Mohammed, T., A., & Saad, A. (2017). Understanding of a Convolutional Neural Network [Conference presentation]. 2017 International Conference on Engineering and Technology.

Alessandretti, L., El Bahrawy, A., Aiello, L. M., & Baronchelli, A. (2019). Anticipating cryptocurrency prices using machine learning. *Complexity*. 1 – 16. http://dx.doi.org/10.1155/2018/8983590

Bryson, A., E. (1961). A gradient descent method for optimizing multi-stage allocation processes. *Proc. Harvard University Symposium on digital computers and their applications*.

Campbell, P. K., Dale, M., Ferrá, H. L., & Kowalczyk, A. (1995). Experiments with neural networks for real time implementation of control [Conference presentation]. 8th International Conference on Neural Information Processing Systems. Colorado, USA.

Chang, P., Wang, D., & Zhou, C. (2012). A novel model by evolving partially connected neural network for stock price trend forecasting. *Expert Systems with Applications, 29*, 611 - 610. https://doi.org/10.1016/j.eswa.2011.07.051

Chen, L., Pelger, M., & Zhu, J. (2021). Deep Learning in Asset Pricing. https://dx.doi.org/10.2139/ssrn.3350138

Cho, K., Raiko, T., & Ilin, A. (2013). Enhanced gradient for training restricted Boltzmann machines. *Neural Computing, 25*(3), 805 - 831. https://doi.org/10.1162/NECO_a_00397

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *arXiv e-prints.*

Corbet, S., Meegan, A., Larkin, C., Lucey, B., & Yarovaya, L. (2018). Exploring the dynamic relationship between cryptocurrencies and other financial assets. *Economic Letters , 165*, 28 - 34. https://doi.org/10.1016/j.econlet.2018.01.004

Elman, J. L. (1990). Finding structure in time. *Cognitive Science , 14*(2), 179 - 211. https://doi.org/10.1016/0364-0213(90)90002-E

Galeshchuk, S. (2014). Neural networks performance in exchange rate prediction. Neurocomputing, 172, 446 - 452. https://doi.org/10.1016/j.neucom.2015.03.100

Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N., Presser, S., & Leahy, C. (2020). The Pile: An 800 GB Dataset of Diverse Text for Language Modeling. *arXiv e-prints.*

Garcia, D., & Schweitzer, F. (2015). Social signals and algorithmic trading of Bitcoin. Royal Society Open Science, 2(9). http://dx.doi.org/10.1098/rsos.150288

Grossberg, S. (1969). Some networks that can learn, remember, and reproduce any numer of complicated space-time patterns, I. *Journal of Mathematics and Mechanics, 19*(1), 53 - 91. https://doi.org/10.1512/iumj.1970.19.19007

Guresen, E., Kayakutlu, G., & Daim, T. U. (2011). Using artificial neural network models in stock market index prediction. Expert Systems with Applications, 38(8), 10 389 - 10 397. https://doi.org/10.1016/j.eswa.2011.02.068

Hecht-Nielsen, R. (1989). Theory of the backpropagation neural network. *International Joint Conference on Neural Networks,* 146 - 160. https://doi.org/ 10.1109/IJCNN.1989.118638

Hochreiter, S., & Schmihuber, J. (1997). Long Short-Term Memory. Neural Computation, 9(8), 1735 - 1780. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.676.4320&rep=rep1&type=pdf

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences, 79*, 2 554 – 2558. https://doi.org/10.1073/pnas.79.8.2554

Jaeger, H. (2001). The “echo state” approach to analysing and training recurrent neural networks. *GMD Report 148, German National Research Center for Information Technology*.

Kalchbrenner, N., Danihelka, I., & Graves, A. (2016). Grid Long Short-Term Memory [Conference presentation]. 4th International Conference on Learning

Representations 2016, San Juan, Puerto Rico. https://arxiv.org/abs/1507.01526v3

Karmiani, D., Kazi, R., Nambisan, A., Shah, A., & Kamble, V. (2019). Comparison of Predictive Algorithms: Backpropagation, SVM, LSTM and Kalman Filter for Stock Market [Conference presentation]. 2019 Amity International Conference on Artificial Intelligence. Dubai, UAE.

Kim, H. Y., & Won, C. H. (2018). Forecasting the volatility of stock price index: A hybrid model integrating LSTM with multiple GARCH-type models. *Expert Systems with Applications, 103*, 25 - 37. https://doi.org/10.1016/j.eswa.2018.03.002

Kohonen, T. (1972). Correlation matrix memories. *Computers, IEEE Transactions on, 100*(4), 352 - 359. https://doi.org/10.1109/TC.1972.5008975

Kristoufek, L. (2013). BitCoin meets Google Trends and Wikipedia: Quantifying the relationship between phenomena of the Internet era. Sci Rep, 3(3415). https://doi.org/10.1038/srep03415

Kristoufek, L. (2015). What Are The Main Drivers of the Bitcoin Price? Evidence from Wavelet Coherence Analysis. PLOS ONE, 10(4). https://doi:10.1371/journal.pone.0123923

Kröse, B., & Smagt, P. (1993). An introduction to neural networks. Journal of Computer Science, 48.

Li, T. R., Chamrajnagar, A. S., Fong, X. R., Rizik, N. R., & Fu, F. (2018). Sentiment-Based Prediction of Alternative Cryptocurrency Price Fluctuations Using Gradient Boosting Tree Model. Front Phys, 7(98). doi: 10.3389/fphy.2019.00098

Linnainmaa, S. (1976). Taylor expansion of the accumulated rounding error. *BIT Numerical Mathematics, 16*(2), 146 - 160. https://doi.org/10.1007/BF01931367

Liu, Y., & Tsyvinski, A. (2021). Risks and Returns of Cryptocurrency. *The Review of Financial Studies, 34*(6)*,* 2 689 – 2 727. https://doi.org/10.1093/rfs/hhaa113

McCulloch, W., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics, 9*, 115 – 133. https://doi.org/10.1007/BF02478259

Medsker, L., R., & Jain, L., C. (2001). Recurrent neural networks. *Design and Applications, 5,* 64 – 67.

Narendra, K., S., & Thathatchar, M. A., L. (1974). Learning automata – a survey. *IEEE Transactions on Systems, Man, and Cybernetics, 4*, 323 - 334. https://doi.org/10.1109/TSMC.1974.5408453

Ndikum, P. (2020). Machine Learning Algorithms for Financial Asset Price Forecasting. *arXiv e-prints*.

Neal, R. M. (2006). Classification with Bayesian neural networks. *Lecture Notes in Computer Science (3944*), 28–32.

Patel, M. M., Tanwar, S., Gupta, R., Kumar, N. (2020). A deep learning-based cryptocurrency price prediction scheme for financial institutions. *Journal of Information Security and Applications, 55,* 49 - 55. https://doi.org/10.1145/3377713.3377722

Ramachandran, P., Zoph, B., & Le, Q., V. (2017). Searching for Activation Functions. *arXiv e-prints.*

Roh, T. H. (2007). Forecasting the volatility of stock price index. *Expert Systems with Applications, 33*(4), 916 - 922. https://doi.org/10.1016/j.eswa.2006.08.001

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review, 65*(8). https://doi.org/10.1037/h0042519

Rumelhart, D. E., Hinton, G., E., Williams, R. J. (1986). Learning internal representations by error propagation. *Parallel Distributed Processing, 1*, 318 – 33.

Schaupp, L., C., & Festa, M. (2018). Cryptocurrency adoption and the road to regulation. *Proceedings of the 19th Annual International Conference on Digital Government Research: Governance in the Data Age, 78,* 1 - 9. https://doi.org/10.1145/3209281.3209336

Schmidhuber, J. (2015). Deep Learning in Neural Networks: An overview. Neural Networks, 61, 85 - 117. https://doi.org/10.1016/j.neunet.2014.09.003

Thakur, A., & Konde, A. (2021). Fundamentals of Neural Networks. *International Journal for Research in Applied Science & Engineering Technology*, 9(8).

Widrow, B., & Hoff, M. (1962). Associative storage and retrieval of digital information in networks of adaptive neurons. *Biological Prototypes and Synthetic Systems, 1*(10). https://doi.org/10.1007/978-1-4684-1716-6_25

Witten, I., Frank, E., Hall, F., A., & Pal, C. (2016). Data Mining: Practical Machine Learning Tools and Techniques (4th ed.). Morgan Kaufmann Publishers In.

Yamak, P. T., Yiujan, L., & Gadosey, P. K. (2019). A comparison between ARIMA, LSTM, and GRU for time series forecasting. *Proceedings of the 2019 2nd International Conference on Algorithms, Computing and Artificial Intelligence,* 49 - 55. https://doi.org/10.1145/3377713.3377722

Yudong, C. & Lenan, W. (2009). Stock market prediction of S&P 500 via combination of improved BCO approach and BP neural network. *Expert Systems with Applications, 36*, 8 849 – 8 854. https://doi.org/10.1016/j.eswa.2008.11.028

Zilouchian, A., & Jamshidi, M. (2001). Intelligent Control Systems Using Soft Computing Methodologies. CRC Press. http://twanclik.free.fr/electricity/electronic/pdfdone4/CRC%20Press%20-%20Intelligent%20Control%20Systems%20Using%20Soft%20Computing%20Methodologies.pdf