



EÖTVÖS LORÁND UNIVERSITY
FACULTY OF INFORMATICS

ALGORITHM FOR CRYPTOCURRENCY PRICE DISCREPANCY DETECTION

IMRE LENDAK

DATA SCIENCE DPT. AT ELTE

MARTA PATIÑO MARTÍNEZ

COMPUTER SYSTEMS DPT. AT UPM

GERGELY TYUKODI

DIGITAL TRANSFORMATION AT OTP

DANIEL MINGUEZ CAMACHO

DSc & ENTREPRENEURSHIP MSc

BUDAPEST, 2020

STATEMENT OF THESIS SUBMISSION AND ORIGINALITY

I hereby confirm the submission of the Master Thesis Work on the Computer Science MSc course with author and title:

Name of Student: Daniel Mészáros Csikvári
Code of Student: 6NY33U
Title of Thesis: ALGORITHM FOR CRYPTOCURRENCY PRICE DISCREPANCY DETECTION
Supervisor: Imre Lendák
.....

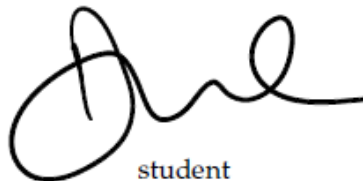
at Eötvös Loránd University, Faculty of Informatics.

In consciousness of my full legal and disciplinary responsibility I hereby claim that the submitted thesis work is my own original intellectual product, the use of referenced literature is done according to the general rules of copyright.

I understand that in the case of thesis works the following acts are considered plagiarism:

- literal quotation without quotation marks and reference;
- citation of content without reference;
- presenting others' published thoughts as own thoughts.

Budapest,



student

Contents

1	Introduction	1
1.1	Previous concepts	3
1.1.1	Cryptocurrency	3
1.1.2	Machine learning	3
1.1.3	Financial time series analysis	4
1.1.4	Limit Order Books	4
1.1.5	Arbitrage trading	5
1.1.6	HFT	5
1.2	Research questions	6
1.3	Previous research	7
2	Cryptocurrency Markets Overview	8
2.1	Marketplaces overview	8
2.2	Marketplaces types	10
2.3	Selected marketplaces	11
3	Dataset formation	15
3.1	Introduction	15
3.2	Problems and limitations	16
3.3	Data gathering & solution architecture	18
3.4	Process details	20
3.4.1	Data acquisition and storage	20
3.4.2	Dataset structure	20
3.4.3	Labels definition	22
3.5	Exploratory data analysis	23
4	Predictive modelling	28
4.1	DeepLOB	29

CONTENTS

4.2	Pre-trained model	30
4.3	Training process	30
4.4	Final results	31
5	Conclusion	33
	Appendices	43
A	Exchanges connections details	44
A.1	Binance	45
A.2	Bitfinex	45
A.3	Bithumb	46
A.4	Bitstamp	46
A.5	Coinbase	46
A.6	Kraken	47
A.7	Huobi	47
B	LSTM and CNN notes	48
B.1	LSTM	48
B.2	CNN	50

List of Figures

3.1	Architecture diagram	18
3.2	Dataset structure (Left) vs model input (Right)	21
3.3	Average ETH-BTC LOB price	24
3.4	Maximum price difference across exchanges	24
3.5	Exchanges with higher and lower prices	25
3.6	Average ETH-BTC average LOB price zoom	26
3.7	Exchanges with higher and lower prices zoom	26
3.8	Labels distribution	27
4.1	DeepLOB variation used	29
4.2	Metrics evolution through the different models	31
B.1	LSTM cell	48
B.2	CNN example	50

List of Tables

2.1	Exchanges characteristics	14
3.1	Records by pair and exchange	23
3.2	Labels	27
4.1	Walk forward validation	30
4.2	Average results	31
4.3	Models comparison	32
A.1	Last trade collection structure	44
A.2	Books snapshots and differences structure	44

Acknowledgement

Thank you very much to Eszter Kiss and Imre Lendák for letting me work on the topic I chose for this master thesis.

To M.V., for your patience and comprehension.

Abstract

As a result of the increasing usage of Bitcoin and other cryptocurrencies as investment and speculative assets, many marketplaces have emerged offering trading services, as well as price aggregators.

Different research papers have analysed cryptocurrency prices in general and Bitcoin prices in particular. One common factor of most of them is that they use data extracted from a small subset of marketplaces or even one alone, so the results obtained are only applicable to a subset of the market. It is needed to test whether these results can be generalized by applying the same methods to the rest of the market.

This Master Thesis tries to address this problem by making a first step towards the creation of a dataset combining prices from different exchanges to the lowest time precision possible. This lower time precision data is more difficult to obtain due to the fact that marketplaces only offer it in a close to real time manner.

A method is described throughout this document, together with the limitations encountered, in order to construct such dataset with data of the pair eth-btc between 31-05-2020 and 02-06-2020. An analysis is provided regarding the information obtained and possible price discrepancies detected across exchanges. Finally, a machine learning model already used in similar scenarios is implemented in an attempt to predict such discrepancies.

List of Abbreviations

AI	Artificial Intelligence
API	Application programming Interface
ARIMA	Autoregressive integrated Moving Average
ARMA	Autoregressive Moving Average
BTC	Bitcoin
CNN	Convolutional Neural Network
DAO	Decentralized Autonomous Organization
DEX	Decentralized Exchange
DNN	Deep Neural Network
ETH	Ether
FIX	Financial Information Exchange
FPGA	Field-programmable gate array
FSA	Financial Services Agency
GARCH	Generalized Autoregressive Conditional Heteroskedasticity
HFT	High Frequency Trading
ICO	Initial Coin Offering
IT	Information Technology
KYC	Know-Your Customer
LOB	Limit Order Book

LIST OF ABBREVIATIONS

LSTM	Long-Short-Term-Memory
ML	Machine Learning
OHLC	Open-High-Low-Close
OTC	Over-the-Counter
P2P	Peer-to-Peer
REST	Representational state transfer
SEC	Securities and Exchanges Commission
SVM	Support Vector Machines
USD	United States Dollar

Chapter 1

Introduction

Bitcoin (BTC) [1] is used by speculative investors to profit from its high volatility. This, among other factors, like its anonymity [2], adoption of cryptocurrency along the business environment [3], the slow evolving regulation [4], imperfect news media or the use of bitcoin for illicit purposes [5, 6]; has created a hype in the cryptocurrency field which led to the creation of several new cryptocurrencies and blockchain applications.

Dozens of cryptocurrency marketplaces have emerged to offer services during the last years, since other more regulated institutions cannot work with them. These marketplaces are present across a large number of countries, each one with different characteristics and it is not clear whether the services offered by some of them are secure or reliable [7].

With the data availability that these marketplaces provide through their public application programming interfaces (APIs), several businesses have grown around them by offering aggregated data [8, 9, 10]. It also allowed the development of algorithmic trading and, to some extent, High-Frequency-Trading (HFT). Nevertheless, there are arguments that can be made against the benefits and possibilities of using HFT in cryptocurrency markets. For example, due to the huge number of marketplaces, the low volume and some restrictions like usage limits it cannot be as effective as it is in other fields like stock exchanges or Forex exchanges. On the other hand, the existence of many different exchanges is prone to present more arbitrage opportunities.

This availability has also helped the study and experimentation with different models and with different types of data like Open-High-Low-Close (OHLC) [11], executed trades data or limit order books (LOB) [12] as an example. These models have been developed and tested against different datasets or sources in almost all of the cases. The datasets were usually obtained through marketplaces APIs, private companies [13, 14] or data aggregators [15].

These datasets are usually in time frequencies like hours or days, although some of them are low-latency related. As far as we are concerned, there is not a well-defined benchmark cryptocurrency low-latency dataset to test these models against in order to be able to generalize results and reproduce their analysis. We think that there should be an open source dataset in the cryptocurrency field for this type of studies and also a standard method for generating this dataset in order to adjust it to different time frames. Some alternatives exist, like in the case of the data provided by data aggregators, but their data use to be further aggregated (hour-days) and their aggregation process is not totally clear, although some of them share part of their methodology¹.

Therefore, during this master thesis we will briefly describe these marketplaces and some of the security risks and regulations affecting them. It is quite relevant to understand these topics due to the several scandals related to the theft of cryptocurrencies from marketplaces or individuals [16]. After this, we will extract market price data from each of the selected marketplaces in order to create a standard dataset. Finally, we will analyse this dataset and try to model the behaviour of this data.

Regarding the structure of the document, after this overview and the introduction of some concepts we use during the thesis, we present our research questions and previous research in the field. In the second chapter we give an overview of cryptocurrency marketplaces in general and the ones included in the analysis in particular. During the third chapter, we describe the method we used for gathering the data from the different marketplaces and put it together. We also detail the limitations and challenges we faced and possible future workarounds. We end this chapter with an exploratory data analysis. In chapter four we introduce possible models that could be used to study price differences based on the last trades and order book data and detail the results of one of them. Finally, in chapter five we answer the research questions and finish with the conclusion.

¹<https://www.coingecko.com/es/metodolog%C3%ADa#:~:text=Methodology,the%20integrity%20of%20the%20data.>

1.1 Previous concepts

1.1.1 Cryptocurrency

Cryptocurrency can be defined as a type of digital currency that use cryptography to ensure transfers, coin ownership and the creation of additional units. It has to have different properties in order to be classified as such [17]:

- Absence of central authority.
- Overview of cryptocurrency units and their ownership.
- Process of generation of new cryptocurrency units.
- Ownership can only be proved cryptographically.
- The system allows transactions to be performed, with a method of distinguish how to proceed if two are performed at the same time.

The most famous cryptocurrency is Bitcoin, created in 2009, but since then thousands of new cryptocurrency have been created based on the concepts developed by Bitcoin.

1.1.2 Machine learning

Machine learning (ML) is a field within artificial intelligence (AI). It studies the development of algorithms that allow computers to perform certain tasks without being explicitly programmed for it. ML has a long history [18] and, since the last 15 years, it has experienced a rapid growth due to the availability of more data and computing power. There are different types of ML algorithms, such as neural networks (NN) and their different architectures (Deep Neural Networks - DNN, Convolutional Neural Networks - CNN, Long-Short Term Memory - LSTM, etc.), random forests, support vector machines (SVM) or association mining rules algorithms among others.

These algorithms can be furthermore classified as supervised, semi-supervised or unsupervised, depending on how they reach the solution of the problems they are faced with. The main difference is that supervised algorithms are trained with labelled data, while unsupervised algorithms are trained with not labelled one. Given the improvement in these techniques and their increase in use for a broad variety of tasks, ML algorithms started to being applied across very different problems like natural language processing, image processing, signal processing, etc.

These problems are common to a broad number of disciplines as well, like computer science, medicine, biology, law, economics or finance. In finance we can find applications such as credit analysis, insurances claims analysis and the application we are focused in this document that is financial time series.

1.1.3 Financial time series analysis

A time series is a time ordered sequence of data points. There are different fields where time series analysis is relevant because they rely in time-ordered data like the prediction of geological events, meteorology, telecommunications engineering, biology or physics. Some examples of techniques used for time series analysis are:

1. Statistical and mathematical analysis and modelling. That is, studying different characteristics of the time series for pattern recognition for example.
2. Signal processing, like ARMA, ARIMA or GARCH methods.
3. Machine learning, with the use neural networks or alike.

Financial time series are those that come from events like trades in an exchange or credits offered through time. This is quite a extensive field covering everything from the data acquisition to the methods for analysing and predicting the behaviour of these time series. In this master thesis we will work with financial time series, specifically with an event-based time series that will include data from seven different exchanges.

1.1.4 Limit Order Books

Limit order books [19] (LOB) are created by exchanges in order to match asset offers and demands from investors. When an order arrives to the marketplace, it is recorded in the LOB and if there is any buy and sell orders that match each other, a trade is performed. Limit order book usually define the so-called price of their asset by averaging the best bid and ask prices.

LOB can be defined by three different elements: the time at which an order arrives to the LOB, the number of units that a trader is willing to commercialize and the price the trader is willing to perform the trade. Units and prices are detailed in each order.

Many different types of orders exist like limit orders or stop orders, that remain in the LOB until some condition is satisfied, they can also be cancelled by the trader usually at any time. Market orders are another example, these types of orders are executed instantly.

Limit order books are a type of financial time series and they have been modelled in different ways like machine learning techniques [20, 21, 22, 23] or more theoretical mathematical models [24, 25, 26].

1.1.5 Arbitrage trading

Arbitrage trading consists in making profit through trading with a close to zero risk involve in the transactions made. For example, by purchasing some asset in an exchange and sell it in another one at the same time at a higher price. There are different types of arbitrage trading like:

- **Cross-market arbitrage** or latency arbitrage [27, 28], inside this type of arbitrage we can find others like triangular arbitrage, this consists in taking advantage of currency pair prices between three or more currencies.
- **Statistical arbitrage** consists in finding trading opportunities by the means of statistical models.
- **Merger arbitrage** involves the acquisition of stocks involves in a merger & acquisition process by buying the target stocks while being short on the buyer side.
- **Convertible arbitrage**, when purchasing a convertible security and then being short in its underlying stock.

These are some examples, but there are other types like regulatory arbitrage or cross-border arbitrage. The work performed during this master is related with cross-market arbitrage along the different markets under analysis. These types of arbitrage can be also be found in the cryptocurrency ecosystem [14, 29].

1.1.6 HFT

High-Frequency-Trading (HFT) is a technique that consists in using computer power, low latency technology and algorithms to perform automatic trades in the market, buying and selling assets at high speed depending of different time series behaviours and market signals. [30]

Algorithmic trading started when electronic exchanges begun to operate in the late 90's and developed into HFT as technology evolved and allowed the kind of operations that can be performed nowadays. Exchanges historically have offered specialized services for those companies that use HFT techniques, like co-location, sliding fees, institutional accounts or specialized APIs.

HFT is more profitable in volatile markets, traders profit for intra-day trades and they don't use to maintain a position for more that trade-day. Many cryptocurrency exchanges already offer these services or are starting to offer them. The cryptocurrency market is highly volatile and its 24h, so it offers high profits for this kind of strategies.

HFT is a controversial topic, with many supporters as well as many others who claim that is unfair and causes negative market externalities. Therefore, they claim that it needs to be more regulated. Some examples of activities performed by HFT are spoofing, momentum ignition or layering. HFT have also led to movement of trade volume over-the-counter or into dark pools.

Specific software and hardware exist in order to work in these high frequencies, for example companies use field-programmable gate arrays (FPGAs) and low-level programming languages like C/C++ in order to execute faster trades and analyse data.

1.2 Research questions

Based on the topics described in the introduction, the research questions we will address during this master thesis will be:

- **Question 1:** Is it feasible to develop an efficient data acquisition tool capable to harvest relevant cryptocurrency data (price, volume, book depth, response time) from multiple marketplaces and APIs suitable for creating a normalized dataset as a basis for a comparative (marketplace) analysis?
- **Question 2:** Is it possible to design a methodology to spot significant price discrepancies based on those datasets?
- **Question 3:** Which method to use with these datasets in order to forecast future price discrepancies with state-of-the-art machine learning algorithms, e.g. Random Forest, LSTM?

1.3 Previous research

The study of the cryptocurrency ecosystem has grown in the previous years as the field consolidated itself. There is a quite extensive bibliography of cryptocurrency price dynamics [15, 31, 32]. These studies are classified in different fields, one is cryptocurrency price formation and its characteristics [33, 34], specially Bitcoin [35] given its predominance in the market. Another example is the comparison of prices of different cryptocurrencies in order to test if different types of cryptocurrencies influence each other [36].

Many research has been focused in cryptocurrency price prediction as well, using data from different time frequencies. Mathematical [37] and machine learning models [38, 39, 40, 41] like Convolutional Neural networks [42, 43, 44, 45] or Long-Short Term Memory networks [46, 47, 38, 39, 40, 41] have been applied with different grades of success. Several studies have developed models for predicting Bitcoin price from LOB [48, 49, 50, 51], historical prices and social media reactions [52, 53]. Agents have also been developed by using these approaches [54, 55, 56].

All above analysis has been applied in both long and short-term scopes, but almost all of them use data either provided by a third party [57], extracted from just one marketplace, or a small subset of marketplaces with higher intervals (hours - days) [15]. Benchmarking in the time series domain is a difficult task [58].

As far as we are concerned, few research has tried to apply models and analysis over lower frequency data from several marketplaces [11, 59] and at the same time explained the methodology used to obtain the data, probably due to the difficulties of obtaining such data.

We think performing that analysis is necessary as a result of some of the problems that market division carries [60]. It is needed also in order to generalize our research as much as possible and test and compare different models.

In the field of high frequency trading, there are different public static datasets [61] either provided by organizations, by researchers, or purchased to companies. Data at such level of detail can only be obtained nowadays, as far as we know, through Websocket and representational state transfer (REST) APIs of the exchanges or purchasing it to private companies.

We want to set up the first steps towards the creation of a cross-market benchmark dataset of cryptocurrency prices, not only a static one, but a method in order to be able to test and reproduce studies.

Chapter 2

Cryptocurrency Markets Overview

2.1 Marketplaces overview

A cryptocurrency marketplace is an electronic platform that a company offers to users in order to buy and sell their cryptocurrencies at any time. There are hundreds of cryptocurrency exchanges operating in the market. At the time of writing this document, a quick view of the main cryptocurrency data aggregators [8, 9, 10], shows more than 700 marketplaces. This is only a portion of the whole cryptocurrency market, that includes also other methods of exchange like P2P sites or direct transactions.

The reasons why this is the case can be explained first of all for the low entry barriers of the field, these applications are relatively easy to create for a programmer with some experience with cryptocurrency wallets. There are even open-source code crypto-exchanges and companies that offer services for creating one.

Moreover, there are many different types of cryptocurrency users [62, 63, 64]. Each of them demands different services, so different marketplaces exists depending on the (non-)control that exchanges have over their customers. Some examples would be: know your customer (KYC) policies in place, the cryptocurrencies they offer, whether or not it is necessary to store the cryptocurrencies in the exchange, security measures in place, the technology facilitators they offer (APIs or co-location for example), etc.

The presence in the market of a huge number of different players also allows investors to distribute funds and reduce the risk of lose everything in a single hacker attack. It also goes in line with the idea of decentralization of the currency. This big number of competitors facilitates a variety of financial services like different fees or trade alternatives such as futures or options. All of this gives differentiation to all players in the market.

Some voices recall that because of the huge number of participants, the prices are less prone to market manipulation techniques since big investors cannot sell in all exchanges, neither algorithmic trading can reach all of them in an affordable way at least. Nevertheless, research have been done in the opposite direction as well, for example analyzing suspicious activity in the MT.Gox exchange [65].

Regarding security, several scandals have arisen since the creation of Bitcoin in 2008. Every year there has been some event where some cryptocurrency exchange have been hacked or some users have lost their funds. The rapid changes in price have also led to several individuals and companies to bankruptcy, and some topics like taxation are still being developed and not clear in every country. Because of it, exchanges are subject to security and regulatory instability. Companies have to adapt to these risks and clients have to deal with them too while governments struggle to keep track of the evolving technology [66].

Different governments try to regulate the cryptocurrency world in different ways [4]. Some of them, like the United States or Europe, try to control fiat-crypto exchanges or regulate Initial Coin Offerings (ICO) [67]. Others try to facilitate the development of companies related to the field like Estonia or Malta while there are countries like China that just ban or ignore them.

Among the different fields where regulation is being developed, some relevant ones are those related to anti-money laundering [68], accounting or financial specific topics like taxation. Some relevant organisms that are in charge of this regulation and are the reference for the different points of view are the Securities and Exchanges Commission (SEC) in United States, the Financial Services Agency (FSA) in Japan, the European Commission and countries like Estonia or Luxembourg in Europe or the Financial Supervisory Service in Korea (among many others). It is worth noticing that, as a result of all of the explained above among other factors, the price of the cryptocurrencies offered at these exchanges may vary. However, little research has been done in this field, where mainly private companies and experienced investors profit from it, but it could be interesting to analyse which factors influence these changes and how these differences evolve over time.

2.2 Marketplaces types

We can classify marketplaces[69] based on different criteria and characteristics:

- Whether they are **centralized** or **decentralized**, that is, if they rely on a centralized entity and there is an organization that hold the servers where the exchange operates or if they are distributed in a decentralized way.
- **Crypto-to-crypto** or **Crypto-to-fiat**, depending on the type of exchanges that can be made in the marketplace.
- Peer-to-peer (**P2P**) or not, if they offer users to create a negotiate their offers with other users or not. This type of marketplaces can offer escrow services as well.
- **Custodial** or not, depending whether they hold client funds or not.
- **Derivatives** or not, if they offer derivatives contracts or not (just spot markets).
- **Instant** or not, if they offer quick trades by operate as a nested service on top of other exchanges.
- Others, there are other ways of trade cryptocurrencies, mostly other Over the counter (**OTC**) [70] methods, cryptocurrency **Brokers** or **nested services** (Marketplaces or other parties like gambling services).

There are other characteristics that a marketplace can have or not, such as the possibility of access to leverage or margin trading. For example, Kraken is a centralized, custodial, crypto-to-fiat exchange; Binance is also centralized, custodial but opposite to Kraken is crypto-to-crypto, it also offers futures contracts and it has an OTC service. These types of exchanges offer high usability and speed.

On the other hand, Bisq is a decentralized exchange, this exchange also tends to be non-custodial by design, preventing funds for being stolen and offer more privacy to the users, although they have other risks like that there is no party moderating transactions.

Finally, ShapeShift is an instant exchange working on top of other exchanges. An example of a P2P exchange would be Remitano. Brokers also have an important place like eToro.

This overwhelming number of exchanges has led to the creation of different companies and projects that aggregate all this information and also "rate" the different exchanges based on some internal methodology [8, 9, 10].

However, it is difficult to determine the most important marketplaces due to their high number and sometimes shady volume reports since also the marketplaces have been accused of inflating their own numbers from where these aggregators take the information. Some of them are also owned by the exchanges themselves, so it could lead to some independence problems.

We have selected the marketplaces of our analysis based on the information obtained from them, selecting a group of marketplaces with more market share, trade options and better APIs.

2.3 Selected marketplaces

Several studies use different marketplaces as source of the data for their analysis [15]. As we discussed, the huge number of participants make it difficult to perform a global analysis, that is why we tried to focus in relevant players in order to get a relevant portion of the market. Nonetheless, the future objective would be to replicate the process we are going to describe including more marketplaces with the purpose of generalize the data we obtain.

We have selected the marketplaces by two metrics of coingecko, first adjusted volume for the day 21/03/2020 19:08, then we selected the 7 marketplaces following the below criteria:

1. Check in Coingecko those marketplaces with trustscore bigger than 9.
2. Verify that these marketplaces were also placed in top 20 on CoinMarketCap and Cryptowatch.
3. Analyse their APIs and remove those that were not detailed enough or did not include some information like timestamp in their messages.

Regarding the currency pair we focused on, eth-btc, we did so because it was the only one present in all the marketplaces selected, since some exchanges do not use fiat currencies and replace them by the so-called stablecoins like Theter or USD Coin. Eth-btc could not be the optimal pair to analyse price differences between exchanges due to its low price and volatility compared with others, but we decide it to use it as a proof of concept for the dataset formation.

Once the concept is proven, the same process can be used to create similar datasets with the same characteristics. Three different parameters need to be defined: time window of the data collection, pairs under analysis and exchanges. If there is a software that can:

1. Connect to the selected exchanges.
2. Extract the data from the pairs chosen with the structure that we are going to define.
3. Replicate the dataset formation process that we describe.

Then, we will be able to construct a similar dataset that could be used to detect price discrepancies and model them.

The final 7 selected marketplaces were the following:

1. **Binance:** it is a centralized, custodial, crypto-to-crypto marketplace that was first founded in China in 2017. Later that year it was moved to Japan due to changes in regulation. Finally it was moved to Malta in 2018 for the same reasons. Below the parent company there are other entities like Binance Labs or Binance DEX, a presumably decentralized exchange, and Binance futures. It has also a P2P OTC service and other secondary services like credit/debit cards that allow to pay with cryptocurrencies. Regarding cybersecurity attacks, in 2019 it suffered a security breach where hackers withdrew 7.000 Bitcoin (BTC). Binance covered the losses with the secure Asset Fund for Users. For a new user trading fees are 0.1%, withdrawal fees for Bitcoin are 0.0004 BTC.
2. **Coinbase:** it is a centralized, custodial, crypto-to-fiat platform created in 2011 and based in San Francisco, California. It started as a project presented in Y Combinator. It is associated with companies like Dell or Expedia and it has investors like BBVA or Bank of Tokyo and it has acquired several blockchain companies like Blockr or Kippt. Coinbase also offers other services like electronic wallets. Trading fees vary between 0.5% or an established fee between \$0.99 and \$2.99 whichever is higher depending on the amount, for digital currency sales and purchases. For digital currencies conversions the fees can be up to 2%. These fees also vary with the location of the buyer. In 2019 Coinbase revealed that the passwords of 3.500 users were made public in a password glitch.
3. **Huobi Global:** centralized, custodial, crypto-to-crypto, Singapore-based exchange. It was founded in 2013 in China, but it was moved to Singapore due to changes in relevant chinese regulation. Since 2018 it is a publicly listed Hong-Kong company. It has several branches like Huobi China and partnership with different institutions like Tsinghua University. Huobi hasn't had major security breaches, most of the stolen funds reported in Huobi were made due to compromised user private keys. Fees are 0.7% for makers in crypto-fiat trades and 0.2% in crypto trades. It has a margin interest rate of 0.098%.

4. **Kraken:** it is a centralized, custodial, crypto-to-fiat, US-based marketplace. It was founded in 2011 following MT.Gox breach and it was launched in 2013. It has made several acquisitions like BitTrade in 2020. Since 2019 it also offers futures trading. Kraken hasn't had major security breaches, but DDoS attacks have reported and also user losses due to private key mismanagement. Trading fees are between 0.16% and 0.26% and margin fees are between 0.01% and 0.02%.
5. **Bithumb:** it is a centralized, custodial, crypto-to-fiat, South Korean - based marketplace founded in 2014 under the holding company BTHMB. It also offers OTC trading and a decentralized exchange. It has partners like Chainalysis or BitMax. It has reported major cybersecurity breaches, like the one in 2017 where a computer from an employee was exposed and client information was stolen, following a loss of \$1 million. The company has also tax related problems due to the regulation in its country. Bithumb trading fees are 0.1% but they vary by pair with different amounts for withdrawal.
6. **Bitfinex:** it is a centralized, custodial, crypto-to-fiat marketplace owned by iFinex Inc., it is registered in the British Virgin Islands but it is headquartered in Hong-Kong. It was founded in 2012 as a peer-to-peer (P2P) exchange. This exchange is closely related with the cryptocurrency Tether and it has not reliable banking partners. Trading fees are 0.1% for makers and 0.2% for takers, withdrawal vary depending on the coin. It has had several security breaches, for example, in 2016 \$72 millions in bitcoin were stolen.
7. **Bitstamp:** it is a centralized, custodial, crypto-to-fiat marketplace. It was founded in 2011 in Slovenia and moved to UK in 2013 and to Luxembourg in 2016. Among its partners are Dukascopy or Swissquote. The trading fees are 0.5% with withdrawal fee varying depending on the currency and the withdrawal type. The company has suffered some cybersecurity attacks, like in 2015 when it suspended service due to a 19.000 BTC stolen.

Selected exchanges				
Name	Country Orig.(-Current)	Fees ¹	Sec.Breaches detected	Currencies
Binance	China - Malta	Trade: 0.1% Withdrawal:0.0004	Yes	Crypto-to-crypto
Coinbase	US	Trade:0.5%, Withdrawal:-	Yes	Crypto-to-fiat
Huobi	China - Singapore	Trade:0.2%, Withdrawal:0.0005	No	Crypto-to-crypto
Kraken	US	Trade:0.16%-0.26%, Withdrawal:0.0005	Yes	Crypto-to-fiat
Bithumb	South Korea	Trade:0.1%, Withdrawal:0.001	Yes	Crypto-to-fiat
Bitfinex	British Virgin Islands	Trade:0.1%-0.2%, Withdrawal:0.0004	Yes	Crypto-to-fiat
Bitstamp	Slovenia - Luxembourg	Trade:0.5%, Withdrawal:-	Yes	Crypto-to-fiat

Table 2.1: Exchanges characteristics

¹Fees vary in almost all exchanges depending of the type of user and pair or coin, we include trades fees for new users and withdrawal fees for Bitcoin. We do not include other types of fees that are not in scope for this document, like margin or futures fees. All fees were last accessed on 18-06-2020.

Chapter 3

Dataset formation

3.1 Introduction

In this chapter we will discuss the data gathering process, its challenges and its limitations. First, an analysis will be performed with data extracted from the different exchanges selected. In this analysis we will look for price discrepancies among the exchanges for the given eth-btc pair and we will study its possible effects taking into account fees and other factors.

It is worth noticing that there are some data and software providers as well as open-source libraries available that facilitate the extraction of data from different exchanges like CCXT [71], XChange [72], GoCryptoTrader [73], Shrimpy [74], Cryptowatch [10] or PyAlgoTrade [75] among others [15]. These tools make some tasks easier like performing data requests and executing orders in multiple exchanges, however, most of them focus on REST API calls and have limited implementation of the exchanges WebSocket APIs.

We decided to implement the connection ourselves since we were interested in obtaining the data from its original source rather than from a third party. We had also to store the data in a custom database and thus, we needed to modify the exchanges messages accordingly. By using a custom solution we could filter all those fields that we weren't going to need, avoiding processing them and improving performance. Also, as we said before, most of the libraries rely on REST API connections that have slower updates and more call limits than Websocket ones. As a future development we are considering to integrate and test some of these libraries and tools.

3.2 Problems and limitations

Gathering data from multiple different exchanges in real time is a challenge, we have faced several limitations during the data collection and manipulation process that we would like to comment here:

1. **Timestamp data aggregation**, one of the main problems we faced with the data structure is that we have no means to determine if the timestamp reported by the different exchanges is actually equivalent [76], that is, that the same timestamp reported by one exchanges meet exactly with the timestamp reported with any other exchange and that there is no significant time delay between them both. Since we have not access to any of the exchanges' servers, we have no means to corroborate this. Therefore, we have to assume that the timestamps are equivalent. This is a problem that cannot be solved without the collaboration of the different exchanges.
2. **Missing or lost data**, it can happen that we lose the connection with an exchange for whatever reason (our own internet connection problem, the exchange disconnect us based on its policies, etc.), since we were working with only one machine, we have to take into account the impact of those disconnections in our dataset formation. In summary, a disconnection would imply that we will carry on the last version of the order book through the time of disconnection and we will have to assume that some trades will be missing from our dataset. Once the connection is re-established, the order book will be updated with the new snapshot and the process will continue.
3. **Legal considerations**, it is not totally clear if the data collected by the different exchanges APIs can be released publicly, since some of them, like for example Bithumb have explicit sentences in their terms of service like *"Members must not {...} Transferring to another person the data obtained through this service, including investment information, market prices, accounts, settlement details, balances, etc"* [77].
4. **Data normalization**, since we are working with different exchanges, each one of them provides a different format for basically the same data that we are interested in. It has been necessary to standardize this data into a common form, putting together different nomenclatures and structures. Among specific difficulties we encountered were the different names for same cryptocurrencies pairs, different order books details in every exchange (like the order book depth) and different decimal approximations for prices.

5. **Processing speed**, the focus of this master thesis was data gathering, data engineering and modelling. We did not focus on processing speed, neither trading nor how to implement a real time application based on our findings.
6. **Model through time consistency**, time series models need to be adapted as time passes. One model fitted and perfectly working during a time window, probably won't work in a different one in the future. We then have to look for a process of obtaining standard data during different time windows, not for a unique static dataset. If we propose a unique static dataset, it could help as a benchmark for ML, but it won't be useful as a research tool for cryptocurrency analysis.
7. **Processing power and internet speed**, we performed our analysis with an average home computer Inter Core I7-4720HQ, 16GB RAM, 2.60GHZ 2594Mhz, 4 Cores, 8 Logical Processors. Therefore, we were not able to test other approaches like clusters or containers to process the calls to the different APIs. The average Internet connection was 700Mbps. We also believe that the whole process of creating the dataset can be further optimized from its current form, for example parallelizing some parts. This also limited our capabilities to develop more complex models and longer training periods.
8. **Frequency**, The APIs also had different frequencies for their data, so it may be delayed by some microseconds between one exchange and another (for example Binance reports changes in its order book every second, while Bitfinex report every change).
9. **Hidden orders**, some exchanges like Binance offer to their customers the service to place hidden orders. That means that those orders are not showed in the public order book. This is a limitation if we want to analyse the dynamics of the price in real time.
10. **OTC trading**, even if we were able to connect all the exchanges that exists today, we would not be able to capture the total volume and price of a cryptocurrency, as there are multiple other channels that are difficult to track, like OTC trading or brokers.

3.3 Data gathering & solution architecture

Exchanges offer different tools to obtain data. A considerable number of them provides WebSocket and REST APIs, some also offer the Financial Information Exchange (FIX) protocol as an alternative. In order to perform this analysis, we have developed an application that extract data from the different exchanges via WebSocket and REST API calls. The architecture then can be summarized as follows¹:

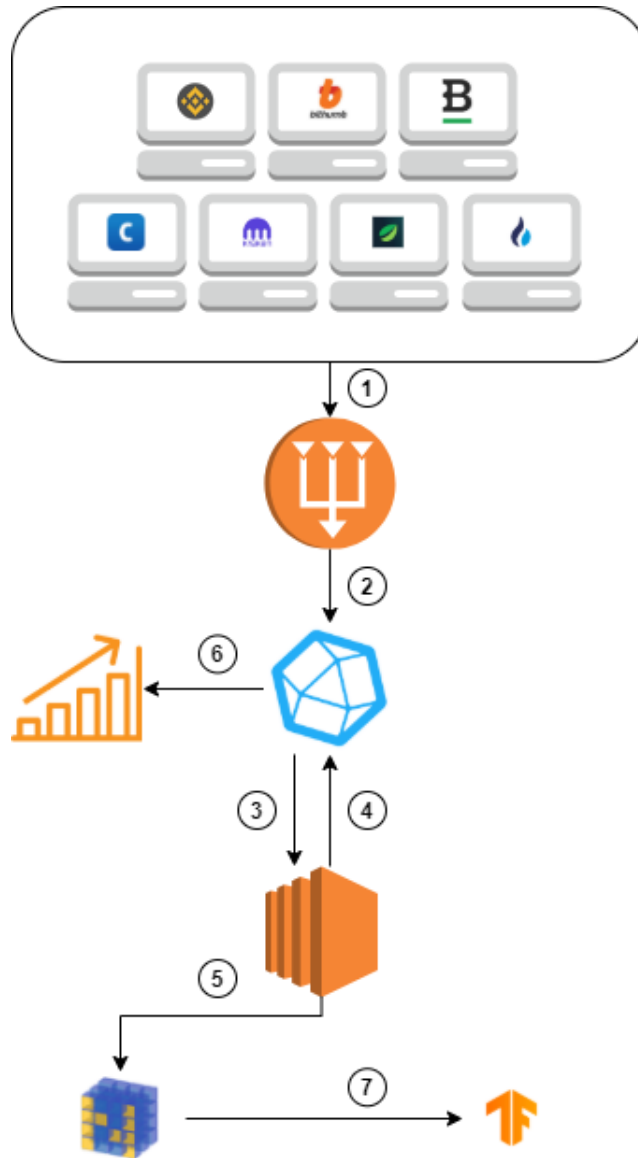


Figure 3.1: Architecture diagram

¹Code can be accessed at <https://github.com/DanielMCM/Thesis>

In order to summarize the whole process as steps, it works as follows:

1. Data gathering from the different exchanges ①.
2. Data storage into an InfluxDB database ②.
3. Reconstruction of order books for those exchanges that need it. This and the following steps are inside ③.
4. Merge of last trades feed with order books data for each exchange.
5. Merge of every group of trades-LOB data for every exchange together by timestamp. Here we have the proto-dataset already formed.
6. Propagation of last values to fill empty ones. We fill empty values between messages for each exchange.
7. Removal of duplicates in order to have an event-based dataset.
8. Computation of pertinent metrics, like mid order book price for every exchange, as well as the labels.
9. Saving of the dataset with numpy structure and its labels into binary files ⑤. We created one file for each hour we had with the purpose of use them as walk-forward validation.
10. Saving of the computed metrics back into the database ④ (mainly data like mean price of order books used for charts construction ⑥).
11. Models training, using the Numpy arrays, TensorFlow and TensorBoard ⑦. The details about the training procedure will be described in the following chapter.

3.4 Process details

3.4.1 Data acquisition and storage

We developed a python program ① that connects with the seven selected exchanges and through WebSocket and REST APIs calls we gathered the information we needed in order to construct the dataset. We deployed the program during different time intervals. We tested the program in previous dates and, in this report, we have used data from 2020-05-31 23:20:00 to 2020-06-02 19:20:00 for training and validating.

The data was processed as it arrives by the program and stored in InfluxDB ②, a specialized database for time series data. We created one measurement (table) for each topic we wanted to analyse, the different measurements (tables) created were ²:

- **Trades**, here we have stored the trades happening in each marketplace for the pair under analysis, whenever a trade arrived, we modified the structure of the message and save it in this measurement.
- **Books**, where we stored the snapshots of the order books for the different pairs. Some exchanges only use this measurement and they do not use the books updates one. This happens if they just release snapshots every second or we decided to proceed that way with them.
- **Books updates**, this includes the upcoming updates of order books snapshots happening at different time rates depending on the WebSocket API capabilities of each market.

3.4.2 Dataset structure

The data was extracted from the different measurements and put it together in a unique dataset using another python program ③. Here, we had to decide which structure we wanted to give to our final dataset, we decided to create an event-based dataset like in [61] rather than a time-based one. We chose this structure mainly due to storage and processing power restrictions since creating a millisecond-based dataset would have needed much more storage capacity and computer power capabilities. Given N the number of records in the dataset and M the number of exchanges, the final input and label for the model have the following structure $\{\{x_j^i\}_{j=1}^{j=M}; y^i\} \quad \forall i \in [0, N]$:

$$x_j^i = \{l, q, \{a_n, v_n^a, b_n, v_n^b\}_{n=0}^{10}\}$$

$$y^i \in 0, 1 ; x_j^i \in \mathbb{R}_{\geq 0}^{42}$$

²For more information about how the data was extracted for each exchange, refer to Appendix A

Where l is the last trade that happened in the exchange j , q the volume of the last trade and the pairs $\{a_n, v_n^a\}$ and $\{b_n, v_n^b\}$ the 10 best asks and bids and their volumes for every exchange. All those values are greater than or equal zero. The final dimension of the train data was $N \times 42 \times 7$, and for the labels N .

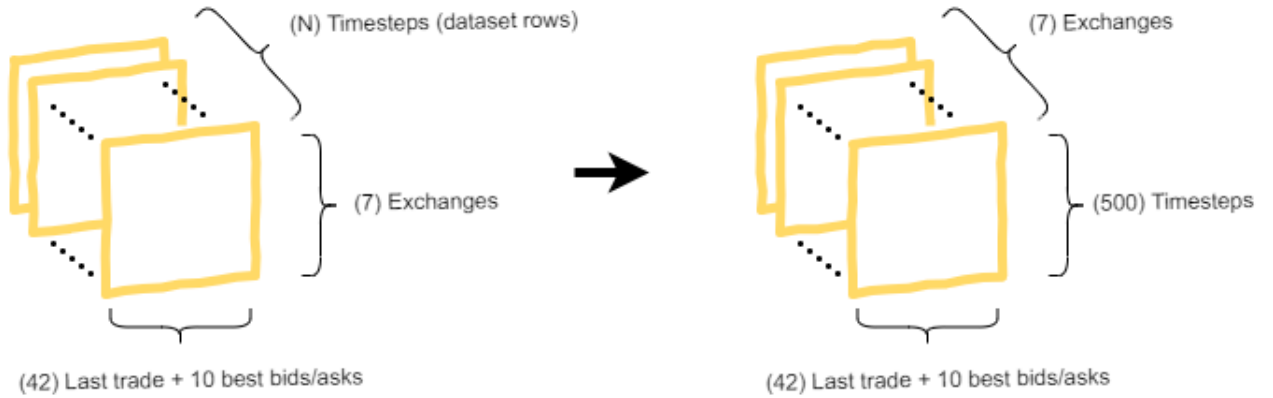


Figure 3.2: Dataset structure (Left) vs model input (Right)

Once this structure is saved, any model can easily transform this base dataset for training with its own generator to adjust the input according to its architecture.

During this step we also normalize the data we are working with. We have two different types of data: prices and volumes. Since prices can take values between $[0, \infty]$, we decided to divide the eth-btc prices by an historically upper limit of eth-btc, 0.16, so we used min-max scaling. For volumes (quantities), we used the following transformation in order to have values between $(0, 1)$:

$$f(x) = \frac{1}{1 + x}$$

3.4.3 Labels definition

Regarding the labels, we were not able to find any previous similar work through the documentation we were able to study, so we tried a similar approach as in [44] and defined a binary label. We wanted to define it as 1 when there was going to be a window of time where it would be possible to send an order to two marketplaces and execute it while existing a significant price difference.

In order to achieve that, we first established a reference where we could say that there is a significant difference between prices across two exchanges at any given time. Given $\mathbf{z}(t) = (z_t^j)_{j=1}^7$, with z_t^j the average order book price of eth-btc at time t in exchange j , we calculated the maximum difference of price among exchanges at any given event, as follows:

$$f(\mathbf{z}(t)) = \max(|z_t^j - z_t^i|) \forall j \in [0, 6]; i \in [j + 1, 7] \quad (3.1)$$

Now, if we want to decide when this difference is significant, we have to establish a reference that helps us to decide. After considered different possibilities, we selected this reference as 0.25% of the average order book price across all the exchanges (Average of the averages of the order books). We selected 0.25% as an approach of the fee that is paid when trading in an exchange. Nevertheless, we let for future work the analysis of the variation of this reference.

Then we defined an event with a significant difference all of those where the maximum price across exchanges was bigger than this reference. Being \bar{z}_t the average of the order books average price at time t . We can define a significant difference as:

$$\mathbf{d}(t) = \begin{cases} 1 & \text{if } f(\mathbf{z}(t)) > \bar{z} \cdot 0.0025 \\ 0 & \text{otherwise} \end{cases} \quad (3.2)$$

Finally, we defined an event as an opportunity event if there were more than 10 consecutive events with significant differences in the next 500 events.

We selected 10 because given the total number of events, the average number of events per second was around this number (9,95) and given the response times of the different exchanges [78] we decided to use this value.

With similar criteria, we selected the window of 500 because it is close to the average minute (50 seconds in average) and we could not work with bigger windows due to performance problems, since we run out of memory and had to use small batch sizes for training our models when choosing bigger windows.

In order to define the final label, we have to define if an event has a significant opportunity:

$$\mathbf{s}(t) = \begin{cases} 1 & \text{if } \sum_{k=t}^{t+10} \mathbf{d}(k) = 10 \\ 0 & \text{otherwise} \end{cases} \quad (3.3)$$

It returns one if there are 10 consecutive significant events, so we can define the final label as follows:

$$\mathbf{y}(t) = \begin{cases} 1 & \text{if } \sum_{k=t}^{k=t+500} \mathbf{s}(k) > 1 \\ 0 & \text{otherwise} \end{cases} \quad (3.4)$$

We will use $\mathbf{y}(t)$ as our label, it will mark if there is any chain of 10 consecutive events during the next 500 which maximum absolute difference is bigger than the average price times 0.0025. For a visual representation of this label for the dataset we constructed, refer to 3.8. Model training will be described in the next chapter.

3.5 Exploratory data analysis

In this section we are going to explore the dataset we just created in order to have an idea of the type of data we are working with. The following table shows the number of records gathered for every exchange:

Exchanges List			
Exchange	Trades records	LOB snapshots	LOB updates
Binance	232.940	404	158.339
Bitfinex	48.530	First: 676 Second: 676 Third: 678	First: 297.106 Second: 301.851 Third: 292.460
Bithumb	9.928	First: 6 Second: 202	First: 196.797 Second: 4.145
Bitstamp	5.558	48.445	-
Coinbase	16.931	First: 678 Second: 677 Third: 677	First: 518.666 Second: 510.324 Third: 509.484
Huobi	60.479	157.687	-
Kraken	91.484	First: 681 Second: 678 Third: 678	First: 549.432 Second: 540.714 Third: 540.817

Table 3.1: Records by pair and exchange

Looking at the table we can see that we made, for example, 678 connections to Coinbase exchange since each connection carries a new snapshot. The same can be said about Bitfinex or Kraken. As we detail in Appendix A, we had to do three different connections alternated in time in order to keep up with the number of messages per second these exchanges produce.

On the other hand, Binance only emit one message per second, so the LOB updates are lower. We also gathered snapshots periodically to cover possible missing updates.

Houbi emits its order book periodically, so we didn't need to store updates, the same for Bitstamp. For Bithumb we had a secondary periodic connection for storing snapshots in order to cover possible missing updates.

The only data that we could not update if lost were the last trades. This is because they are only emitted once when they happen, so if these messages fail to arrive, it is not possible to add them afterwards (unlike what happens with LOB snapshots).

Once we put all the data together organized by timestamp, we obtained a dataset with 1.576.370 entries. We created 44 pairs of input/labels binary files (one pair per hour).

Regarding the average price across the exchanges of the pair ETH-BTC, it was the following:

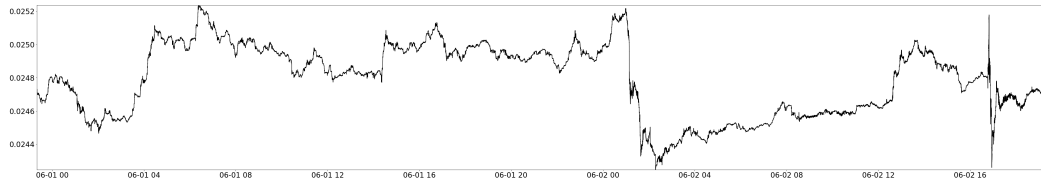


Figure 3.3: Average ETH-BTC LOB price

Regarding the average price over the period, it was 0.02480. Looking at the graph we can appreciate two "mayor" sudden changes in price between 01:00h - 03:00h and the 17:30h - 18:30h of the 2nd of June among other minor changes.

We also analysed the maximum difference across every exchange. Here we can find a plot with the maximum difference across time (3.1). The red line is the average price times 0.0025 in (3.2):

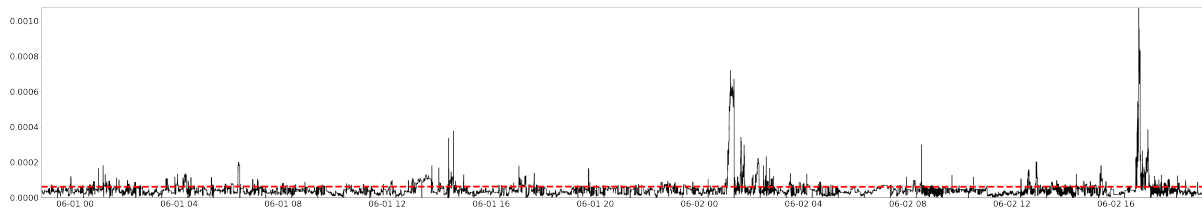


Figure 3.4: Maximum price difference across exchanges

We can appreciate how the periods with higher volatility and sudden changes are the ones where we can find higher differences across exchanges. Nonetheless, we have to be cautious with this assumption since part of the difference may be influenced by decimal approximations that some exchanges make or by the fact that exchanges update prices in different time frequencies, so sudden changes can have some delays in some of them and therefore resemble changes in prices across exchanges in our dataset. The differences can be even due to small disconnections periods, high volume of messages that could not be processed by the software or even periods where exchanges did not update their messages correctly.

We can have some insight about this by analysing which exchanges have the maximum and minimum values during these sudden changes, for doing that we can look at the following plot:

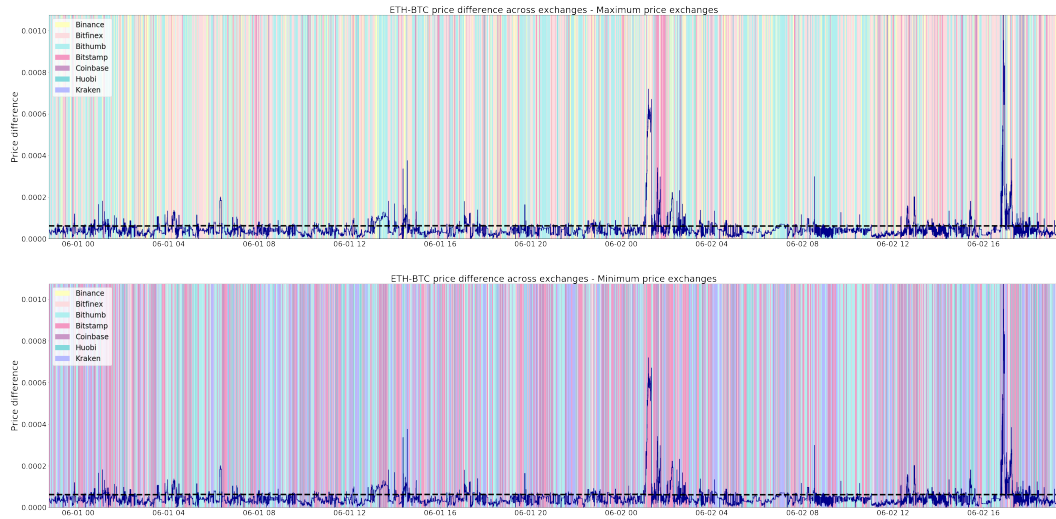


Figure 3.5: Exchanges with higher and lower prices

In this graph we represent the maximum difference across time 3.1, together with the average price time 0.0025 as before. It is also included a background color with one colour for each exchange under analysis. In the top graph we represent the exchanges with higher prices at any point in time, meanwhile, in the bottom one, we represent the exchanges with the lower prices.

We add these graphs in order to have a quick overview about which exchanges tend to be predominant in each one. We can see how Bithumb or Bitfinex tend to have higher prices and exchanges like Bitstamp, Coinbase or Huobi lower ones. Of all these exchanges, only Huobi and Bitstamp update their prices in a second-based manner, this could have some influence in the price difference and affect the graph and the dataset, as we stated before 8.

In the case of Bithumb we can also take into account the Kimchi Premium phenomenon that can maybe explain this situation. This phenomenon [79] is defined as a gap that tend to appear between Korean cryptocurrency exchanges and the rest.

However, we would need to continue the study during longer periods of time and perform quality tests over the data in order to corroborate these observations.

We can zoom into one of the periods where sudden changes appear:

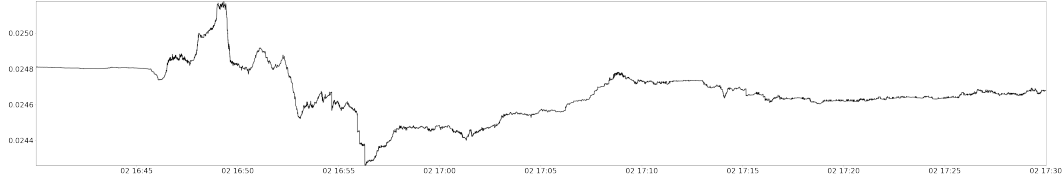


Figure 3.6: Average ETH-BTC average LOB price zoom

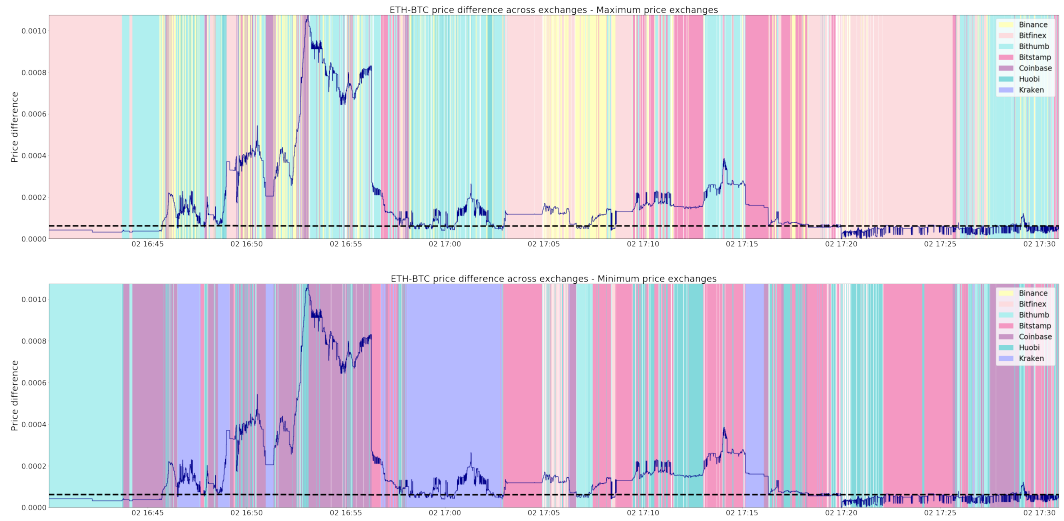


Figure 3.7: Exchanges with higher and lower prices zoom

Again, further analysis and data quality controls need to be established in order to verify that these changes are motivated by true discrepancies or are caused by connection or update frequency reasons.

Finally, we can also plot the labels 3.4 distribution across time and the number of occurrences:

Exchanges List	
Label	Records
0	1.204.388
1	371.982

Table 3.2: Labels

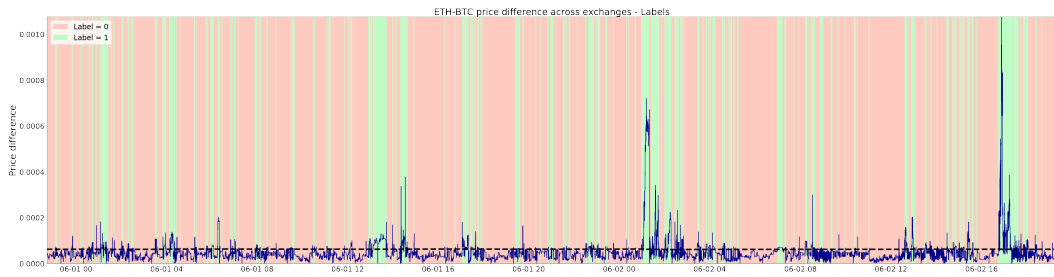


Figure 3.8: Labels distribution

We can see that the 0 class has more representation than the 1 class. We took this into account when training our model, implementing some techniques in order to balance our dataset.

Mainly we sampled elements from the both classes in order to have a balanced batch while training and also we kept occurrences of the underrepresented class from one hour to the next in order to use them to complete the data.

Chapter 4

Predictive modelling

Even though it is possible that the data we collected had different limitations, we decided to implement some models as a proof of concept. Even if differences are motivated by other means, we can use such a model for predicting when they will occur and anticipate it, for example by devoting more computing power to process messages or detecting that a sudden price change may occur.

After gathering, creating and analyzing the data, we have considered different models that could be adequate for our problem of analyzing price differences across exchanges in order to try to predict whether this price difference of eth-btc will go up from a predefined threshold.

As we briefly commented in the introduction, several techniques have been used to model cryptocurrency prices and machine learning models have shown relative better performance. This is the reason we decided to consider machine learning models like CNN and LSTM¹ [39] rather than traditional methods like ARIMA or GARCH.

Nevertheless, as far as we have researched, we have not found any case of studying price discrepancies across different marketplaces with low-frequency data, no model has been directly used for this purpose. Because of this, we decided to adapt one model used in a similar scenario to the problem we are facing with. We then compare the results obtained with this model with the results obtained with a well-known pre-trained model. We decided use a pre-trained model as a comparison, even though it was pre-trained for a different task, because of the lack of results in similar problems.

Other techniques could be also suitable for this problem like multi-layer convolutional LSTM, but, since our objective is to establish a first model as a benchmark, we let this for future research.

¹For more information refer to Appendix B

4.1 DeepLOB

Based on previous work with similar LOB data, as we detailed in the introduction, we considered that CNN and LSTM architectures could be better for addressing the problem we presented. We decided to use a modified version of DeepLOB [44] since it uses both architectures.

This model is a deep learning model designed to predict price movements from limit order book (LOB) data of cash equities. It uses convolutional filters to capture the spatial structure of LOB as well as LSTM modules to capture longer time dependencies.

The original model was tested against equities data, but since the base data structure used was LOB data, we decided to test if it could be extended also to cryptocurrency data. However, we had to overcome different limitations of the model in order to be able to apply it to our dataset.

First of all, the original model was designed to work with just one LOB. In the problem we are facing with, we have several ones, so we had to modify the number of input channels used in the original model and introduce one channel per exchange LOB data as an input. The authors also define a 3-dimensional output, but here we have a 2-dimensional one, so we also modified the output of the model. Using a similar representation as in the original paper, the final architecture would be:

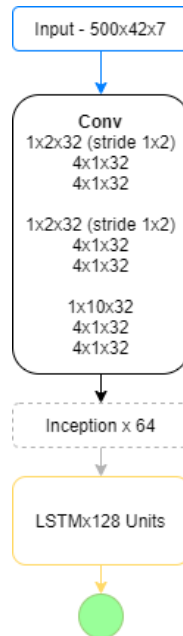


Figure 4.1: DeepLOB variation used

4.2 Pre-trained model

Due to the lack of results to compare the model performance with, we decided to use a pre-trained model as a reference. Since CNN were first developed, pre-trained models and specific architectures have been also published such as Inception [80], RESNET [81], Xception [82], etc.

After reviewing the models performance² and input size requirements, we opted for VGG16 [83]. We had to introduce at the beginning two 3x3 CNN layers with padding for maintaining shape. This was in order to adequate the input channels from seven to three. We also added three dense layers at the end after the last Max pooling layer with 1024, 512 and 2 units respectively. We trained two variations, one with all parameters (Complete VGG16, C-VGG16) and another one with VGG16 parameters frozen (Frozen VGG16, F-VGG16).

4.3 Training process

Regarding the training process, we used mini-batch stochastic gradient descent with momentum as optimizer [84] with learning rate of 0.0001 and momentum 0.9, we let for future work the analysis of the effect of different parameters and optimizers in the model. Regarding the loss function, since we used one-hot encoding for the labels, we selected categorical cross-entropy.

In the case of DeepLOB, we have followed a walk forward validation approach for the training process. We trained different models adding one hour at a time, using the next hour as validation:

Models		
Model	Training data	Validation data
Model 1	Hour 1	Hour 2
Model 2	Hour 1 + Hour 2	Hour 3
Model 3	Hour 1 + Hour 2 + Hour 3	Hour 4
...

Table 4.1: Walk forward validation

As some notes, we used Tensorboard to track the evolution of the trained models. Also, since the dataset was too big to fit it all in memory at once, we trained the model using Tensorflow generators.

²<https://keras.io/api/applications/>

In order to interpret the results, we have to take into account that when evaluating and training the model, we always made sure to create batches with half of entries of each class, as some studies recommend [85, 86, 87] so the input to the train and validation dataset was always balanced.

4.4 Final results

We decided to evaluate the performance of th different models using accuracy, precision, recall and F1. The final results for the DeepLOB model trained were:

DeepLOB Results				
Result	Accuracy %	Precision %	Recall %	F1 %
Average	60.61	65.79	43.94	49.42
Best	96.09	97.33	94.78	96.03

Table 4.2: Average results

In the following graphs we can see the evolution of the scores over the different models iterations:

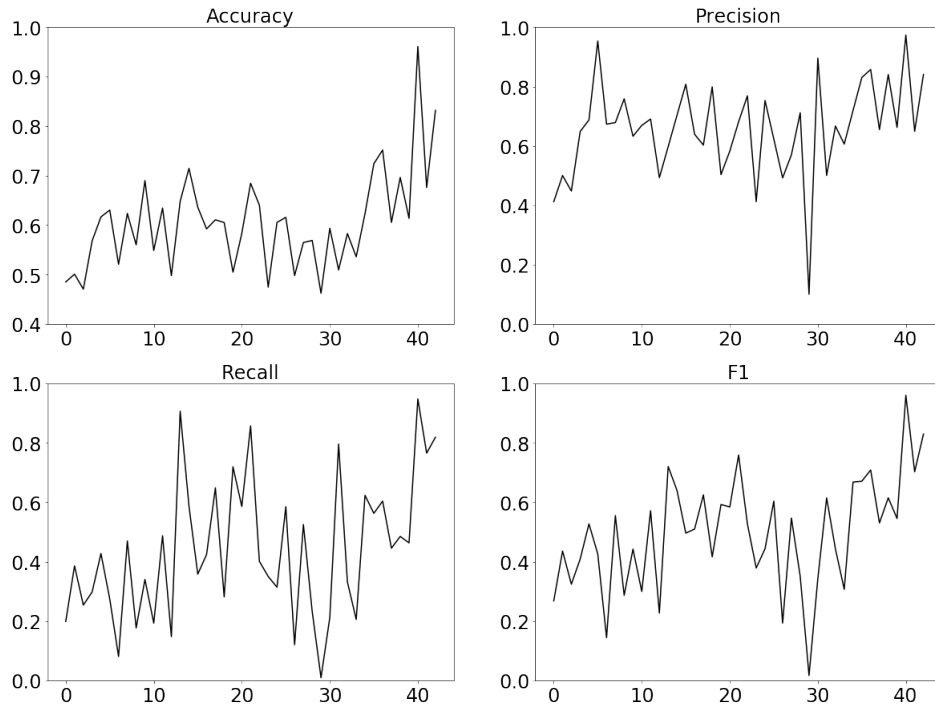


Figure 4.2: Metrics evolution through the different models

We can see that, even though prevision seems to be stable during time, the models accuracy and recall improves in the last ones when more data is available to train. This could signal that with more data models could improve performance.

In order to compare these results with other models, we trained the modified versions of VGG16 at four different points in time (Models 10, 20, 30 and 40), we obtained the following results (Averaged):

Models				
Model	Accuracy %	Precision %	Recall %	F1 %
DeepLOB	68.73	75.12	55.47	63.80
C-VGG16	56.49	68.75	26.12	37.36
F-VGG16	62.82	65.71	50.73	55.50

Table 4.3: Models comparison

We can see that it seems that DeepLOB performs better this can be reasonable since the architecture of this model was designed to fit a similar data architecture, whereas VGG was designed for a totally different problem, although it has showed good performance in different fields.

As we can see, none of the models shows meaningful performance. This can for a different number of reasons, like the need of adding more data to the models, the issues already commented that appeared during the dataset construction, the choice of the data normalization methods or the choice of parameters when training among others. The nature of the data itself is highly stochastic, so it also may influence the difficulty for relevant results. As a matter of reference, state-of-the-art models for mid LOB price prediction accuracy between 0.6146 and 0.8447 [44], depending on the dataset and the prediction horizon used for training the model.

Chapter 5

Conclusion

During this master thesis, we have proposed and developed a methodology for extracting relevant data related with cryptocurrency prices and aggregate it in a machine-learning-ready dataset, we also have presented the tools needed for doing so. After that, we have suggested a methodology for analyzing possible price discrepancies across exchanges. Even though if these discrepancies come from exchange APIs limitations or processing issues, this methodology could be used to spot the sudden changes in prices that create them. Finally, we discussed different models that could be used to predict price differences among these cryptocurrency prices, implementing one ML model, DeepLOB, and comparing it with pre-trained models performance for reference.

We consider this as a first step towards the development of a benchmark high-frequency dataset like FI-2010 but in the cryptocurrency field, without the collaboration of exchanges themselves. The use of such dataset could help to enrich many of the previous work in the field and could help to verify it, extend it and develop new research. This dataset, presented as a method, could create potentially unlimited data that could also be used in other problems like developing order book simulations or a universal testing model for the cryptocurrency market [88].

Nevertheless, we think there are still many challenges that can be addressed on the way of creating this dataset and the models applied to it. More data quality analysis needs to be performed over the data provided by exchange APIs and about how to deal with the difference in update times and other issues. Longer periods of analysis need to be used for this.

Regarding machine learning models, we can think about how to create a model that could handle the problem when having a variable number of exchanges, not a fixed one. It could be interesting as well to analyze how this difference in prices occurs between centralized and decentralized exchanges and study whether and how changes in prices propagate from one another.

Alternatives ways of creating the dataset could involve the creation of different features before training the model. These features could include, for example, information related with price formation for the pair in general and information related with marketplaces in particular [89, 90, 91].

Labels could also be created in many different ways, for example a model could be trained to predict the exchanges that will incur in a bigger price difference in the following window of time. Although it is possible that many more data would be needed in order to achieve meaningful results.

The involvement of the exchanges themselves could help to solve some of the problems we faced, like the timestamp limitation, and it could help to increase the accuracy of the data gathered avoiding interruptions in the data gathering process.

Once the issues commented during the document are resolved, the next step would be the development of a real time implementation of the models developed here. This would present additional challenges like how to create the model structure in an efficient way, probably using different languages and programming tools.

Finally, we provide our answer to the request questions presented at the beginning:

- **Question 1:** Is it feasible to develop an efficient data acquisition tool capable to harvest relevant cryptocurrency data (price, volume, book depth, response time) from multiple marketplaces and APIs suitable for creating a normalized dataset as a basis for a comparative (marketplace) analysis?

As we have showed during this master thesis, it is possible in fact to develop a data acquisition tool capable to harvest relevant cryptocurrency data from different marketplaces, even with a limited hardware like a home computer. Nevertheless, this process could take a lot of advantage of the computing power of a computer cluster for example, deploying API calls in a parallel manner and creating redundancy connections to the different exchanges in order to avoid data losses and disconnections. We have showed also that there are some limitations and assumptions we have to make in order to be able to work with the data that it is possible to obtain.

- **Question 2:** Is it possible to design a methodology to spot significant price discrepancies based on those datasets?

Even though we have not found previous open work directly related to the problem we were trying to address in this master thesis, we have presented a basis methodology to track this problem with an approach related to other similar techniques used in other previous work in similar scenarios. We hope this could help the future development of similar methods.

However, many challenges need to be addressed and many assumptions need to be done right now in order to apply the methodology we present, so future work will be needed in order to improve it.

- **Question 3:** Which method to use with these datasets in order to forecast future price discrepancies with state-of-the-art machine learning algorithms, e.g. Random Forest, LSTM?

Based on our study about similar problems in the field, we have gone through possible models that could be applied in this particular problem.

We have shown how CNN and LSTM are predominant in similar problems in the field and we have proposed and implemented one machine learning algorithm that combines both of them as a baseline for future work.

We also have compare the performance of this model with well-known pre-trained models in order to give a reference about how well it worked. This can be interpreted as a start point for future work.

Bibliography

- [1] Nakamoto S. Bitcoin: A peer-to-peer electronic cash system, 2008.
- [2] Benjamin Fabian, Tatiana Ermakova, Ulrike Sander. Anonymity in bitcoin – the users’ perspective, 2016.
- [3] Garrick Hileman, Michel Rauchs. Global cryptocurrency benchmarking study, 2017. Available at https://www.jbs.cam.ac.uk/fileadmin/user_upload/research/centres/alternative-finance/downloads/2017-04-20-global-cryptocurrency-benchmarking-study.pdf.
- [4] Library of Congress. Regulation of cryptocurrency around the world, 2020. Available at <https://www.loc.gov/law/help/cryptocurrency/world-survey.php>.
- [5] Robby Houben, Alexander Snyers. Cryptocurrencies and blockchain. legal context and implications for financial crime, money laundering and tax evasion, 2018.
- [6] Foley S, Karlsen J, Putniņš T, Goldstein I, Jiang W, Karolyi A, Weber M and Easley D. Sex, drugs, and bitcoin: How much illegal activity is financed through cryptocurrencies?, 2018.
- [7] Matthew Hougan, Hong Kim and Micah Lerner. Economic and non-economic trading in bitcoin: Exploring the real spot market for the world’s first digital commodity, 2019. Available at <https://www.sec.gov/comments/sr-nysearca-2019-01/srnysearca201901-5574233-185408.pdf>.
- [8] Coinmarketcap, top 100 cryptocurrency exchanges by trade volume. Last access 11/02/2020. Available at <https://coinmarketcap.com/rankings/exchanges/>.
- [9] Coingecko, cryptocurrency exchange ranking by trust score (spot). Last access 11/02/2020. Available at <https://www.coingecko.com/en/exchanges>.
- [10] Cryptowatch. Last access 09/05/2020. Available at <https://cryptowat.ch/>.

- [11] Vo Au, Yost-Bremm C. A high-frequency algorithmic trading strategy for cryptocurrency, 2018.
- [12] Mate Puljiz et al. Market microstructure and order book dynamics in cryptocurrency exchanges, 2018.
- [13] Takuya Shintate, Lukáš Pichl. Trend prediction classification for high frequency bitcoin time series with deep learning, 2018.
- [14] Igor Makarova, Antoinette Schoarb. Trading and arbitrage in cryptocurrency markets, 2019.
- [15] Fang F, Ventre C, Basios M, Kong H, Kanthan L, Li L, Martinez-Regoband D and Wu F. Cryptocurrency trading: A comprehensive survey, 2020.
- [16] Chainalysis. The 2020 state of crypto crime, 2020.
- [17] Jan. Possible state approaches to cryptocurrencies, 2018.
- [18] Haohan Wang and Bhiksha Raj. On the origin of deep learning, 2017.
- [19] Martin D. Gould, Masib A. Porter, Stacy Williams, Mark McDonald, Daniel J. Fenn and Sam D. Howison. Limit order books, 2013.
- [20] Nikolaos Passalis, Anastasios Tefas, Juho Kanninen, Moncef Gabbouj and Alexandros Iosifidis. Temporal logistic neural bag-of-features for financial time series forecasting leveraging limit order book data, 2019.
- [21] James Wallbridge. Transformers for limit order books, 2020.
- [22] Ulrich Horst and Dörte Kreher. Second order approximations for limit order book, 2019.
- [23] Paraskevi Nousia et al. Machine learning for forecasting mid price movement using limit order book data, 2019.
- [24] Johannes Bleher, Michael Bleher and Thomas Dimpf. The what, when and where of limit order books, 2020.
- [25] Dat Thanh Tran, Juho Kanninen, Moncef Gabbouj, Alexandros Iosifidis. Data normalization for bilinear structures in high-frequency financial time-series, 2020.
- [26] Matthias Schnaubelt, Jonas Rende and Christopher Krauss. Testing stylized facts of bitcoin limit order books, 2019.
- [27] Cristian Păuna. Arbitrage trading systems for cryptocurrencies. design principles and server architecture, 2018.

- [28] Tomasz Czapliński and Elena Nazmutdinova. Using fiat currencies to arbitrage on cryptocurrency exchanges, 2019.
- [29] Gina Pieters, Sofia Vivanco. Financial regulations and price inconsistencies across bitcoin markets, 2017.
- [30] Joel Hasbroucka, Gideon Saar. Low-latency trading, 2013.
- [31] Rodolfo C. Cavalcantea, Rodrigo C. Brasileiro, Victor L.F. Souza, Jarley P. Nobrega, Adriano L.I. Oliveira. Computational intelligence and financial markets: A survey and future directions, 2016.
- [32] Gagan Deep Sharma et al. Emergence of bitcoin as an investment alternative, 2019.
- [33] Nadarajah S and Chu J. On the inefficiency of bitcoin, 2017.
- [34] Roman Matkovskyy, Akanksha Jalan. From financial markets to bitcoin markets: A fresh look at the contagion effect, 2019.
- [35] Šurda, Peter. Economics of bitcoin: is bitcoin an alternative to fiat currencies and gold? Master’s thesis, WU Vienna University of Economics and Business, 2012.
- [36] Ciaian P, Rajcaniova M and Kancs d. Virtual relationships: Short- and long-run evidence from bitcoin and altcoin markets, 2018.
- [37] Devavrat Shah and Kang Zhang. Bayesian regression and bitcoin, 2014.
- [38] Laura Alessandretti, Abeer ElBahrawy, Luca Maria Aiello and Andrea Baronchelli. Anticipating cryptocurrency prices using machine learning, 2018.
- [39] Suhwan Ji, Jongmin Kim and Hyeonseung Im. A comparative study of bitcoin price prediction using deep learning, 2019.
- [40] Bruno Spilakr. Deep neural netowrks for cryptocurrencies price prediction. Master’s thesis, Humboldt-Universität zu Berlin, 2018.
- [41] Avraam Tsantekidisa, Nikolaos Passalisa, Anastasios Tefasa, Juho Kanninenb, Moncef Gabboujc and Alexandros Iosifidis. Using deep learning for price prediction by exploiting stationary limit order book features, 2018.
- [42] Omer Berat Sezer, Ahmet Murat Ozbayoglu. Algorithmic financial trading with deep convolutional neural networks: Time series to image conversion approach, 2018.
- [43] Ashwin Siripurapu. Convolutional networks for stock trading, 2015.

- [44] Zihao Zhang, Stefan Zohren and Stephen Roberts. Deeplob: Deep convolutional neural networks for limit order books, 2019.
- [45] Saul Alonso-Monsalve, Andres L. Suarez-Cetrulo, Alejandro Cervantes, David Quintana. Convolution on neural networks for high-frequency trend prediction of cryptocurrency exchange rates using technical indicators, 2020.
- [46] Fan Fang, Waichung Chung, Carmine Ventre, Michail Basios, Leslie Kanthan, Lingbo Lid and Fan Wu. Ascertaining price formation in cryptocurrency markets with deep learning, 2020.
- [47] Thomas Fischer and Christopher Krauss. Deep learning with long short-term memory networks for financial market predictions, 2017.
- [48] Justin A. Sirignano. Deep learning for limit order books, 2018.
- [49] Jian Wang. Ensemble methods for capturing dynamics of limit order books, 2017.
- [50] Andrea Barbon. Focusing at high frequency. an attention-based neural network for limit order books, 2019.
- [51] Eduard Silantsev. Order flow analysis of cryptocurrency markets, 2019.
- [52] Lamon C, Nielsen E, Redondo E. Cryptocurrency price prediction using news and social media sentiment, 2017.
- [53] Kim YB, Kim JG, Kim W, Im JH, Kim TH, Kang SJ, et al. Predicting fluctuations in cryptocurrency transactions based on user comments and replies, 2016.
- [54] Masafumi Nakano, Akihiko Takahashi and Soichiro Takahashi. Bitcoin technical trading with artificial neural network, 2018.
- [55] Jonathan Sadighian. Deep reinforcement learning in cryptocurrency market making, 2019.
- [56] Haoran Wei, Yuanbo Wang, Lidia Mangu and Keith Decker. Model-based reinforcement learning for predictions and control for limit order book, 2019.
- [57] Elie Bouri, Chi Keung Marco Lau, Brian Lucey and David Roubaud. Trading volume and the predictability of return and volatility in the cryptocurrency market, 2018.
- [58] Xinyuan Huang et al. Benchmarking deep learning for time series: Challenges and directions, 2019.

BIBLIOGRAPHY

- [59] A Sensoy. The inefficiency of bitcoin revisited: A high-frequency analysis with alternative currencies, 2018.
- [60] Maureen O’Hara, Mao Ye. Is market fragmentation harming market quality?, 2011.
- [61] Adamantios Ntakaris, Martin Magris, Juho Kannianenb, Moncef Gabbouj, Alexandros Iosidis. Benchmark dataset for mid-price forecasting of limit order book data with machine learning methods, 2017.
- [62] Yelowitz A, Wilson M. Characteristics of bitcoin users: an analysis of google search data., 2015. 22(13):1030–6.
- [63] Ahmed Zouhair, Dr. Noah Kasraie. *Disrupting Fintech: Key Factors for Adopting Bitcoin*, pages 33–34. Business and Economic Research, Macrothink Institute, vol. 9(2), 2019.
- [64] Jeremiah Bohr, Masooda Bashir. Who uses bitcoin? an exploration of the bitcoin community, 2014.
- [65] Gandal, N Hamrick, J Moore T and Oberman T. Price manipulation in the bitcoin ecosystem, 2018.
- [66] Hossein Nabilou, André Prüm. Central banks and regulation of cryptocurrencies, 2019.
- [67] Dr. Philipp Hacker, LL.M. and Dr. Chris Thomale, LL.M. (Yale). Crypto-securities regulation: Icos, token sales and cryptocurrencies under eu financial law, 2019.
- [68] CipherTrace. Cryptocurrency anti-money laundering report, 2019 q4, 2020.
- [69] The 5 key types of cryptocurrency exchanges [chainalysis blog]. Available at <https://blog.chainalysis.com/reports/cryptocurrency-exchange-types>.
- [70] Gutmann R, Knehr J, Rapoport P and Stevens R. Buying bitcoin, 2019.
- [71] Ccxt – cryptocurrency exchange trading library. Last access 25/05/2020. Available at <https://github.com/ccxt/ccxt>.
- [72] Xchange. Last access 25/05/2020. Available at <https://github.com/knownm/XChange>.
- [73] Gocryptotrader. Last access 25/05/2020. Available at <https://github.com/thrasher-corp/gocryptotrader>.
- [74] Shrimpy. Last access 25/05/2020. Available at <https://developers.shrimpy.io/docs/#introduction>.

BIBLIOGRAPHY

- [75] Pyalgotrade. Last access 25/05/2020. Available at <http://gbeced.github.io/pyalgotrade/docs/>.
- [76] Leslie Lamport. Time, clocks, and the ordering of events in a distributed system, 1978.
- [77] Bithumb api service terms and conditions. Last access 25/05/2020. Available at https://www.bithumb.com/resources/data/20180911_bithumb_APIService_EN.pdf.
- [78] Order speed analysis reveals the fastest cryptocurrency exchanges. Last access 02/06/2020. Available at <https://news.bitcoin.com/order-speed-analysis-reveals-the-fastest-cryptocurrency-exchanges/>.
- [79] Choi K, Lehar A and Stauffer R. Bitcoin microstructure and the kimchi premium, 2019.
- [80] Christian Szegedy et al. Going deeper with convolutions, 2014.
- [81] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun. Deep residual learning for image recognition, 2015.
- [82] François Chollet. Xception: Deep learning with depthwise separable convolutions, 2017.
- [83] Simonyan K and Zisserman A. Very deep convolutional networks for large-scale image recognition, 2014.
- [84] Sebastian Ruder. An overview of gradient descent optimization algorithms, 2017.
- [85] Johnson, Justin and Khoshgoftaar, Taghi. Deep learning and thresholding with class-imbalanced big data, 12 2019.
- [86] Laurikkala J. Improving identification of difficult small classes by balancing class distribution, 2001.
- [87] Wei Q and Dunbrack R. The role of balanced training and testing data sets for binary classifiers in bioinformatics, 2013.
- [88] Justin Sirignano and Rama Cont. Universal features of price formation in financial markets: Perspectives from deep learning, 2018.
- [89] Kercheval and A Zhang Y. Modeling high-frequency limit order book dynamics with support vector machines, 2013.
- [90] Adamantios Ntakaris, Giorgio Mirone, Juho Kanninen, Moncef Gabbouj, Andalexandros Iosifidis. Feature engineering for mid-price prediction with deep learning, 2020.

BIBLIOGRAPHY

- [91] Alameda Research. Investigation into the legitimacy of reported cryptocurrency exchange volume, 2019.
- [92] Sepp Hochreiter, Jurgen Schmidhuber. Long short-term memory, 1997.
- [93] Savvas Varsamopoulos, Koen Bertels. Designing neural network based decoders for surface codes, 2018.
- [94] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung. Convolutional lstm network: A machine learning approach for precipitation nowcasting, 2015.
- [95] Yann LeCun, Leon Bottou, Yoshua Bengio and Patrick Haffner. Gradient based learning applied to document recognition, 1998.

Appendices

Appendix A

Exchanges connections details

In this section we are going to go through the different exchanges and how we interfaced with them, describing which requests we used and how we merged them together. We created three structures for storing the data: "Trades", for storing the trades of every exchange; "Books", for storing the order books snapshots of every exchange; and "Books updates", for storing the order book updates for the exchanges that required it. All urls in this section were last accessed on 15-06-2020. The structure of the collections was the following:

Trades				
Timestamp	Host	Pair	Price	Q
...

Table A.1: Last trade collection structure

Book & Book differences					
Timestamp	Host	Pair	Follow-up info	Bids	Asks
...

Table A.2: Books snapshots and differences structure

A.1 Binance^{1, 2}

The base url for Binance websocket is `wss://stream.binance.com:9443/ws` and the REST API url is `https://www.binance.com/api/v1`:

- **Trades**, we used `<BASE_URL>/<symbol>@trade`.
- **Books**, as detailed in their documentation, we stored here snapshots obtained for the REST API at periodic intervals `<REST_BASE_URL>/depth?symbol=<symbol>&limit=<limit>`
- **Book updates**, here we used the depth socket, `<BASE_URL>/<symbol>@depth`

A.2 Bitfinex^{3, 4}

The base url for Bitfinex websocket is `https://api.bitfinex.com/v1`. In the case of Bitfinex, we launched three different calls for getting the book updates and reconnect them one by one periodically every few minutes in order to have redundancy and loose as few messages as possible. With this method we also had periodic order books snapshots every time we reconnected the websocket:

- **Trades**, we used the trades channel.
- **Books**, filled with raw books channel snapshot. We also made periodic requests to `<BASE_URL>/book/symbol` in order to be able to have a system timestamp, since the timestamp is not included in the raw books channel.
- **Book updates**, completed with raw books channel updates.

¹Main page: <https://www.binance.com>

²Documentation: <https://github.com/binance-exchange/binance-official-api-docs>

³Main page: <https://www.bitfinex.com/>

⁴Documentation: <https://docs.bitfinex.com/docs/ws-general>

A.3 Bithumb^{5, 6}

The base url for Bithumb websocket is `wss://global-api.bithumb.pro`. For the order book, we launched two websocket connections and we also reconnected them one by one periodically in order to have periodic order book snapshots.

- **Trades**, we used `<BASE_URL>/message/realtime?subscribe=TRADE:<symbol>`
- **Books**, filled with the snapshot send by `<BASE_URL>/message/realtime?subscribe=ORDERBOOK:<symbol>`
- **Book updates**, completed with the updates of the previous socket.

A.4 Bitstamp^{7, 8}

The base url for Bitstamp websocket is `wss://ws.bitstamp.net`

- **Trades**, filled with `live_ticker` channel.
- **Books**, completed with the `detail_order_book` channel.
- **Book updates**, due to connectivity problems with the `dif_order_book` channel we only were able to work with the snapshots provided in the Book collection.

A.5 Coinbase^{9, 10}

The base url for Coinbase websocket is `wss://ws-feed.pro.coinbase.com`. For Coinbase we also launched three different connections for gathering the order book updates.

- **Trades**, for this collection we used the `matches` channel.
- **Books**, in this case the collection was filled with the `level2` channel snapshot.
- **Book updates**, completed with `level2` channel updates

⁵Main page: <https://en.bithumb.com/>

⁶Documentation: <https://github.com/bithumb-pro/bithumb.pro-official-api-docs>

⁷Main page: <https://www.bitstamp.net/>

⁸Documentation: <https://www.bitstamp.net/websocket/v2/>

⁹Main page: <https://www.coinbase.com>

¹⁰Documentation: <https://docs.pro.coinbase.com/>

A.6 Kraken^{11, 12}

The base url for Kraken websocket is `wss://ws.kraken.com`. We also launched three connections for Kraken order book.

- **Trades**, completed with the trade channel data.
- **Books**, here we included the snapshot provided by the book channel.
- **Book updates**, filled with the book channel updates.

A.7 Huobi^{13, 14}

The base url for Kraken websocket is `wss://api.huobi.pro/ws`.

- **Trades**, we used the trade detail channel.
- **Books**, here we completed the collection with the market depth channel snapshots.
- **Book updates**, in this case we only used the snapshots due to problems reconstructing the order book with the differences provided by the API and connection problems.

¹¹Main page: <https://www.kraken.com/>

¹²Documentation: <https://docs.kraken.com/websockets/>

¹³Main page: <https://www.huobi.com/>

¹⁴Documentation: <https://huobiapi.github.io/docs/spot/v1/en/>

Appendix B

LSTM and CNN notes

B.1 LSTM

Long-Short Term Memory [92] is a type of recurrent neural network, these types of neural networks are able to extract patterns from sequences of data. LSTM are composed by memory cells and improve vanilla RNN by adding different mechanisms of how to handle the information at each time step.

Let's define \mathbf{x} as our input, \mathbf{y} as our labels and \mathbf{h} the outputs of the model at each step. We will use the notation W_{uv}^o to describe the weights between elements u and v at operator o .

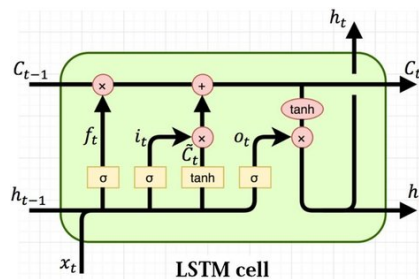
$$\mathbf{x} = \{x_t\}_{i \in [0, T]}, \quad x_i \in \mathbb{R}_a$$

$$\mathbf{y} = \{y_t\}_{i \in [0, T]}, \quad y_i \in \mathbb{R}_b$$

$$\mathbf{h} = \{h_t\}_{i \in [0, T]}, \quad h_j \in \mathbb{R}_b$$

Where T would be the training set size, a the dimension of the input, b the dimension of the output. As a visual representation of the cell and the equations we are going to describe we can follow the representation below [93]:

Figure B.1



There are different structures for LSTM cells, according to [93] equations for each of the units will be:

$$\begin{aligned}
 i_t &= \sigma(x_t W_{x_t c_t}^i + h_{t-1} W_{h_{t-1} c_t}^i) \\
 f_t &= \sigma(x_t W_{x_t c_t}^f + h_{t-1} W_{h_{t-1} c_t}^f) \\
 o_t &= \sigma(x_t W_{x_t h_t}^o + h_{t-1} W_{h_{t-1} h_t}^o) \\
 \tilde{C}_t &= \tanh(x_t W_{x_t C_t}^{\tilde{C}} + h_{t-1} W_{h_{t-1} C_t}^{\tilde{C}}) \\
 C_t &= \sigma(f_t * C_{t-1} + i_t * \tilde{C}_t) \\
 h_t &= \tanh(C_t * o_t)
 \end{aligned}$$

Where σ use to be the sigmoid function, although variations exists. \tanh the hyperbolic tangent:

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

In particular, one variation that could be interesting to try with our data model and that we let for future implementations would be Convolutional LSTM [94], the equations detailed before are modified in order to include convolution operations, bias vectors b^i and previous C_{t-1} in the different operators:

$$\begin{aligned}
 i_t &= \sigma(x_t \circledast W_{x_t c_t}^i + h_{t-1} \circledast W_{h_{t-1} c_t}^i + C_{t-1} \circ W_{c_{t-1} c_t}^i + b^i) \\
 f_t &= \sigma(x_t \circledast W_{x_t c_t}^f + h_{t-1} \circledast W_{h_{t-1} c_t}^f + C_{t-1} \circ W_{c_{t-1} c_t}^f + b^f) \\
 o_t &= \sigma(x_t \circledast W_{x_t h_t}^o + h_{t-1} \circledast W_{h_{t-1} h_t}^o + C_{t-1} \circ W_{c_{t-1} c_t}^o + b^o) \\
 \tilde{C}_t &= \tanh(x_t \circledast W_{x_t C_t}^{\tilde{C}} + h_{t-1} \circledast W_{h_{t-1} C_t}^{\tilde{C}} + b^{\tilde{C}}) \\
 C_t &= f_t \circ C_{t-1} + i_t \circ \tilde{C}_t \\
 h_t &= \tanh(C_t \circ o_t)
 \end{aligned}$$

By stacking two or more layers of these Convolutional LSTM we could feed snapshots of our order book data and create a model that would be able to extract information from the different dimensions, the 42x7 LOB dimension and the temporal dimension.

B.2 CNN

Convolutional Neural Networks (CNN), were developed in order to improve the models used for image recognition, but these have been also applied to a broad group of fields.

The architecture of a CNN can be pictured as follows [95]:

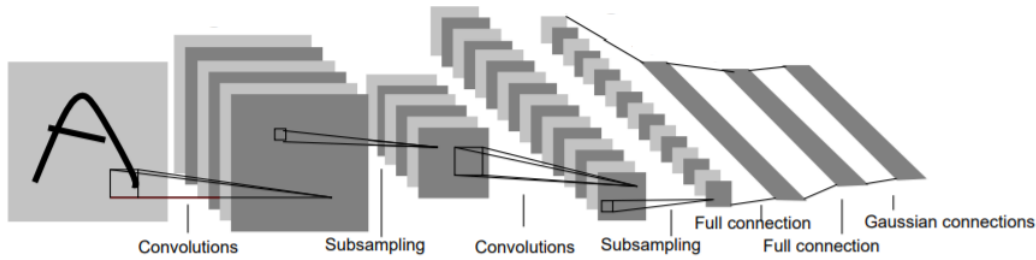


Figure B.2: CNN example

In order to define the convolution operation we have to define the number of layers L , denoted $\{l = 1, \dots, L\}$. In each layer l there are $m_1^{(l)}$ feature maps. Let's define $y_i^{(l)}$ as the feature map i output of layer $l - 1$, with $\{i = 1, \dots, m_1^{(l)}\}$. Each feature map $y_i^{(l)}$ has a dimension $m_2^{(l)} \times m_3^{(l)}$, and they are computed as:

$$y_i^{(l)} = b_i^{(l)} + \sum_{j=1}^{m_1^{(l-1)}} K_{i,j}^{(l)} * y_j^{(l-1)} \quad \forall i = 1, \dots, m_1^{(l)}$$

Where $b_i^{(l)}$ is the bias and $K_{i,j}^{(l)}$ the filter that connects the feature map j^{th} in layer $(l - 1)$ with the i^{th} feature map in layer l .