DigitalCurrency

Bohan Li, Zhekai Dong, Hyeseong Choi
CS445/545 Fund of Digital Archeology
bli43@vols.utk.edu; zdong7@vols.utk.edu; hchoi9@vols.utk.edu

abstract

In this project, we build a trading system based on DQN model with trading indexes including ADX, RSI, CCI, Vol to help trading digital currencies. We train model to do a long term daily trading for five currencies including bitcoins, ltc, eth, bch, eos and also to do a short term hour trading for bitcoins and ltc. In the long term trading results, DQN perform well at most cases but in the short term trading, it still need to be improved in future.
Keywords: DQN, trading indexes, digital currencies, long term trading, short term trading.

Introduction

Decisions in the stock trading process often depend on different perspectives. On the one hand, from the perspective of time, stock trading can often be divided into large cycle trading and small cycle trading. Large cycle trading are often low-frequency and small cycle trading are often high-frequency trading. On the other hand, stock trading analysis often involves new analysis, fundamental analysis and technical analysis. The news way tends to trade stock relying on good or bad news from newspapers or other medias and the fundamental analysis focus on business philosophy and financial status of a listed company related to a certain stock However, the technical way is more inclined to analyze stock and to help choosing trading strategies by using technical indicators of stocks. In this project, we only focus on the technical way to help trading and these technical indexes are ADX, RSI, CCI, Vol.

ADX: The average directional movement index is a combination of two other indicators: the positive directional indicator (+DI) and negative directional indicator (-DI). The ADX just added them together and take the average of them so that the conclusion is less various. In order to calculate these two indicators, the dataset must contains high, low, and closing prices each period. ADX = 100 times the smoothed moving average of the absolute value of (+DI - -DI) divided by (+DI + -DI)

RSI: The relative strength index is to graph the strength or weakness of a stock price or market based on the closing prices of a recent trading period. The relative strength factor is converted to a relative strength index between 0 and 100.

$$\text{RSI} = \frac{100}{1+RS} \times 100\%$$

CCI: The commodity channel index is used for technical analysis either. It will calculate a security's variation from the statistical mean. SMA in here is the simple moving average. MD in here is the mean absolute deviation. P in here is price.

$$CCI = \frac{1}{0.015}\frac{pt - SMA(pt)}{MD(pt)}$$

Vol is the trading volume per day.

DQN is a method to predict or approximate the Q-value function by deep learning. A Q-value function in reinforcement learning is a table recording the values of actions for an agent in different states. After the training of reinforcement learning, the Q-table value will converge and then the action that agent will take in a certain state will always be the action with max value in Q-table. In this way, we can tell the agent how to behavior in different states.

However, when the number of Q-table is huge, there are usually no space to store them. That is why we need deep learning to help reinforcement learning. The deep neural network will be trained by the current data or Q-value(Out) and the sates in Q-table(Input). After training of deep neural network, we can view the network as a Q-table and directly obtain the Q-value of a action according to the state.This is general work of reinforcement learning.

Objective

In this project, our experimental object is digital currency, which is an investment product similar to stocks and also has prevailed in recent years. We will try to build a trading system based on technical analysis(trading index) by reinforcement learning and deep learning models(DQN). After building model, we will train model by 5 different digital currencies include bitcoin, ltc, eth, bch, eos. The test will also include long term trading, short term trading, daily trading(low frequency) and hour trading(high frequency). The objective of this project is to build such a model that can make profit in the most cases above.

Motivation

Digital currency is a young and emerging market. Immature trading market means that in most cases, the effects of news analysis and fundamental analysis will be weakened and technical analysis will dominate in trading.
With the development of machine learning, deep learning, and reinforcement learning in quantitative trading, there are already many stock trend predictions based on LSTM models, and many reinforcement learning models do high-frequency trading. Models with LSTM or CNN do have a strong ability to do a prediction but it face a big problem of training and parameters tuning. At same time, the simple reinforcement learning have to require a huge space to store Q-table, which is also impossible to a problem with almost infinite states in trading. Therefore, the deep reinforcement learning or DQN with deep neural network but not CNN and LSTM become a good choice, which is also applied for stock trading in previous years and it has been proved that deep reinforcement learning methods can help trader makes profit.
Therefore, we believe that under pure technical indicator analysis, effective trading strategies and operational references can be proposed by DQN model that we can train by enough datasets.

Dataset

The data of this project will be selected by three ways. The 5 different digital currencies data will come from cointelegrah.com, which include the complete price history of each of them. The 2 hour price data of bitcoin and ltc are partially from two major digital currency exchanges in China.
They comes from github and this link is: https://github.com/speculatecat/BitcoinPriceHistoryInChina.
In addition to the available data, the link also provides two python files used to link the exchange's API to obtain data.
Secondly, the data contains the five-year historical price and trend of the five major currencies from 2013 to 2017 at Huobi and Okcoin exchanges.
Initially, here it should be mentioned that not all the data of the five digital currencies are complete but we complete them by using the data from cointelegrah.com. Third,

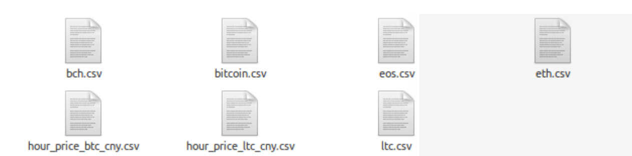

Figure 1: This is dataset of the digital currency will be used in this project. It includes btc, eth, ltc, eos, bcc five types of digital currencies. All have daily price data but only btc and ltc have extra hour price data that are used for high frequent trading. All data are saved in the form of .csv

among the five digital currencies, not all digital currencies have five-year transaction data due to the difference in the time of issuance.
In the data of btc.csv, there are almost 10 years data with daily price, which has exactly 3647 samples.
In the data of eth.csv. there are almost 5 years data with daily price , which has exactly 1949 samples.
In the data of ltc.csv, there are almost 7 years, which has exactly 2616 samples.
In the data of bch.csv, there are almost 3 years, which has exactly 1225 samples.
In the data of eos.csv, there are almost 3 years, which has exactly 1225 samples.
For the two hour data files, they are both from 2017-07-11 06:00:00 to 2017-09-19 06:00:00. There are 1681 samples.
Finally, each data in the csv file has six features. They are opening price, high price, low price, closing price and volume.
Overall, in the final version of the data, we have 7 .csv files.

	date	open	high	low	close	volume
0	2013-09-11 16:00:00	15.0	16.0	12.0	12.0	0.35
1	2013-09-12 16:00:00	16.0	16.1	15.9	15.9	10.411
2	2013-09-13 16:00:00	15.9	16.0	15.6	16.0	361.099
3	2013-09-14 16:00:00	16.3	16.3	14.5	14.5	17.434
4	2013-09-16 16:00:00	16.0	16.0	14.9	14.9	0.134

Figure 2: In original dataset, the type of date is string and the type of other features are float. For each feature of float type, the range of its value is from zero to positive infinity.

Each one of the 5 files of the daily price of btc , eth, ltc, eos, bch is a matrix with the number of days to rows and 6 columns. For the files of hours price of them, there are 1681 rows and 6 columns in each file.

Data process

For each sample in dataset, it will be processed to a sate(see state space) in reinforcement learning. The value of each element in state will be normalized.

Data spilt

For all datasets in this project, 65% data will be used to train model, 10% data will be used to do validation and 25% will be used to test model performance.

Algorithm

In this project, because the goal is to implement a model that can make profit in trading, the model will be the agent and we will train the agent and teach it how to make profit in trading. In order to do this, we will implement some methods to help to achieve the goal: Reinforcement Learning, Deep Learning with Q-learning.
Therefore, for this project, the algorithm we use is DQN, a type of deep Q learning. The model will mainly consist of two parts: deep learning and reinforcement learning.
For the part of reinforcement learning, we use Q-learning. The state space will be defined as:

State:$(ADX_t, RSI_t, CCI_t Vol_t, Position_t, Return_t)$

ADX: Average Directional Index
RSI: Relative Strength Index
CCI: Commodity Channel Index
Vol: trading volume per day
Position: long, short, or flat which will be represented by [1,0,0], [0,1,0], [0,0,1].
Return: unrealized profit or loss before the close position
t: step time(a certain day)
Terminate: the state obtained by the last data in the dataset
The action space will be defined as:

$$\text{Action:}(Sell, Hold, Buy)$$

Q-learning:

$$Q^*(s_t, a) \rightarrow r(s_t, a_t) + max_a Q^*(s_{t+1}, a)$$

The reward structure is:

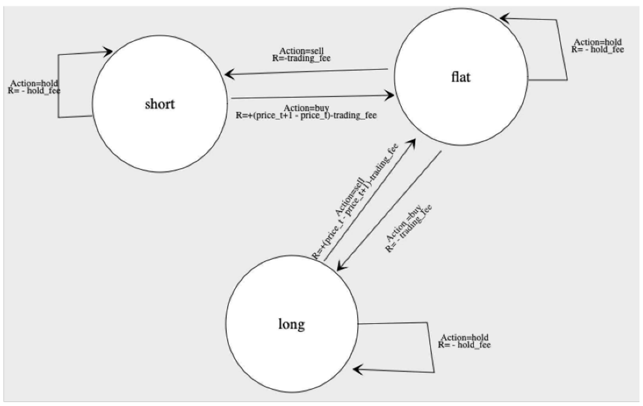

Figure 3: Reward depends on different pairs of state and action. Since the elements in each sate are independent, we can chose one of them to represent a certain state. Here we chose position.

For the part of deep learning, we build a deep neural network based on the Keras framework for Q-value function prediction. There are 8 inputs including and 3 outputs including the Q value of action hold, sell, buy. The structure of deep neural network can be shown as follows(Figure 4).

In the deep learning, the Q-learning will be updated in error fucntion as:

$$L(w) = E[(r + max'_a Q(s', a', w) - Q(s, a, w))^2]$$

The flow work of our model in this project can be shown as follows:

Outcome

Experiment setup:
Data: 65% train, 10% validation, 25% testing.
Iterations: 50
Trading fee: 0.001
Time fee: 0.001 holding fee
Gamma: 0.96
Batch size: 64

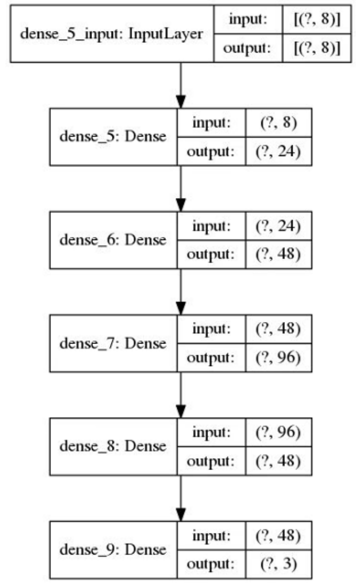

Figure 4: There are 4 hidden layers with 24, 48, 96, 48 neurons, and 1 input layer with 8 inputs and 1 output layers with 3 outputs

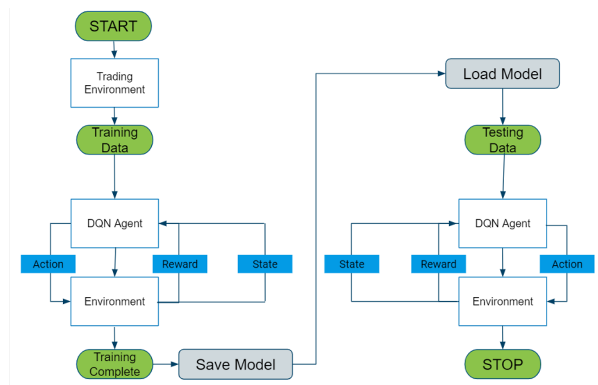

Figure 5: Work flow for this project

Learning rate: 0.001
Activation function: (relu , relu, relu, relu, linear)
Loss function: mse
Optimizer: Adam
Initial trading principal: 800 digital currences and 100000$
Profit = (final trading principal-initial trading principal) / initial trading principal X 100%
Objection: the price history digital currencies
The basic goal of the experiment will be that we can make a right trend judgment in a large cycle. The better expected out is that we can achieve good profits in short-term high-frequency trading. However, the results how that model does work on the long term trading but it can't make profit in a short term high frequency trading.

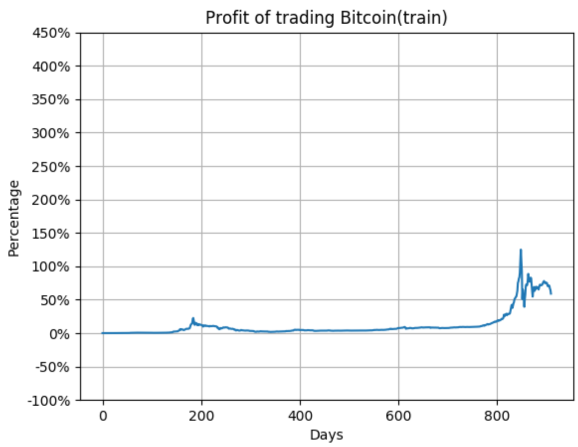

Figure 6: Long term trading for Bitcoin. In this results, we test our trained model by using the first 800 days price of Bitcoin(train data). As we can see, agent keep making profit

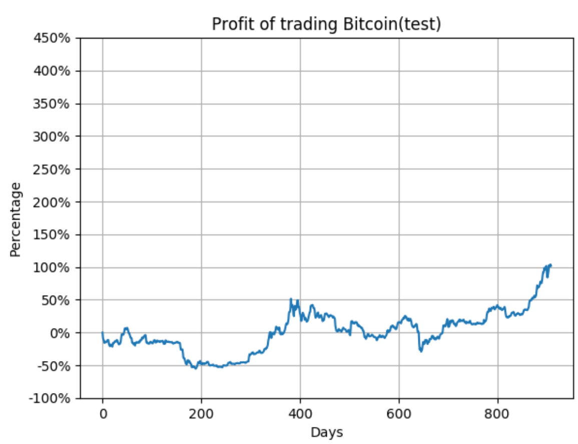

Figure 7: Long term trading for bitcoin. In this results, we test our model by using the most recent 800 days price of bitcoin. As we can see, even agent get money lost at initial, it finally make profit and almost make total value double

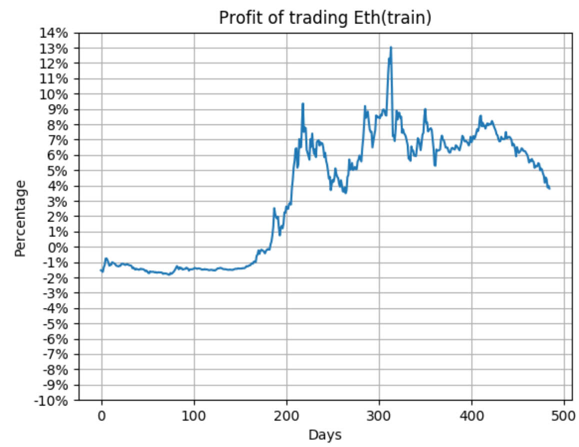

Figure 8: Long term trading for eth. In this results, we test our trained model by using the first 500 days price of eth(train data). As we can see, agent keep making profit but not very good profit

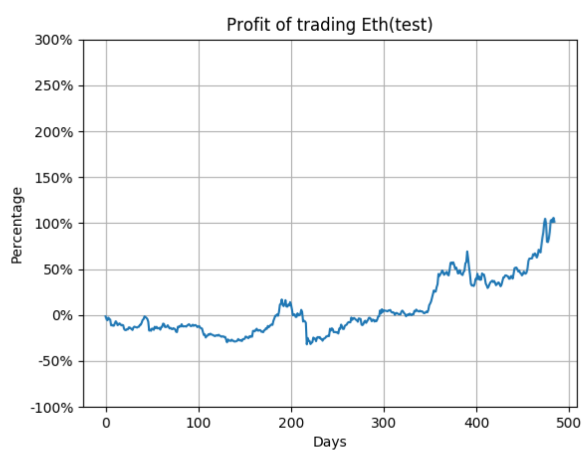

Figure 9: Long term trading for eth. In this results, we test our model by using the most recent 500 days price of eth. As we can see, even agent get money lost at initial, it finally make profit and almost make total value double

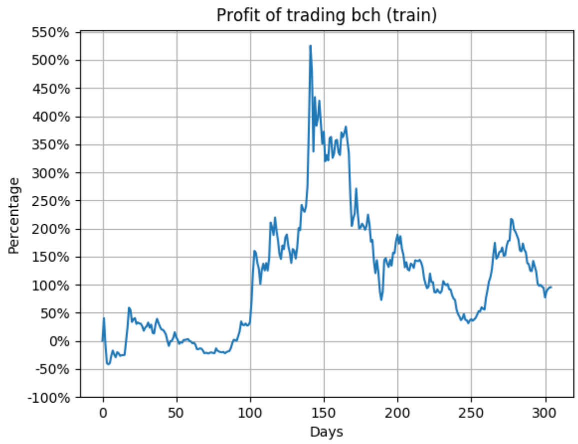

Figure 10: Long term trading for bch. In this results, we test our trained model by using the first 300 days price of bch(train data). As we can see, agent still can make profit but also with higt lost

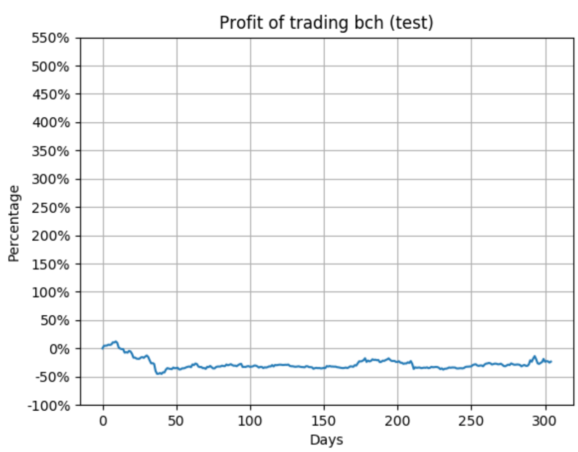

Figure 11: Long term trading for bch. In this results, we test our model by using the most recent 300 days price of bch. However, agent are keep losing in this case

Figure 12: Long term trading for eos. In this results, we test our trained model by using the recent 800 days price of eos. As we can see, agent keep making profit but not very good profit

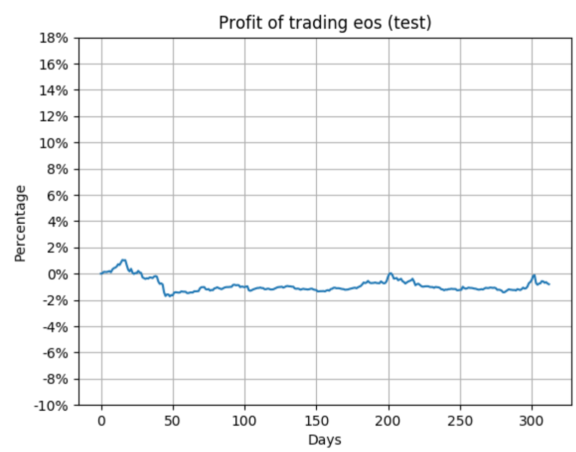

Figure 13: Long term trading for eos. In this results, we test our model by using the most recent 300 days price of eos. However, agent are keep losing in this case.

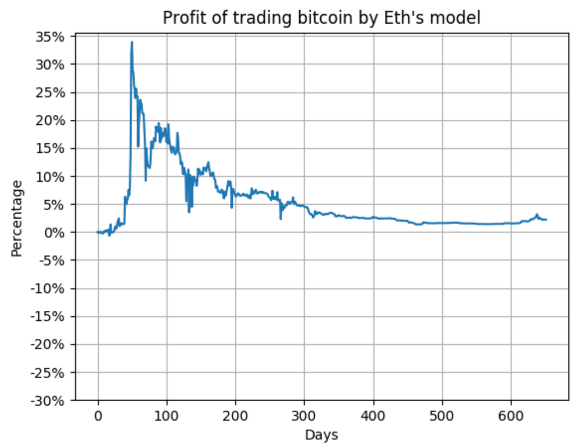

Figure 14: Long term trading for ltc. In this results, we test our trained model by using the first 600 days price of ltc(train data). As we can see, agent keep making profit but almost lost all profit eventually

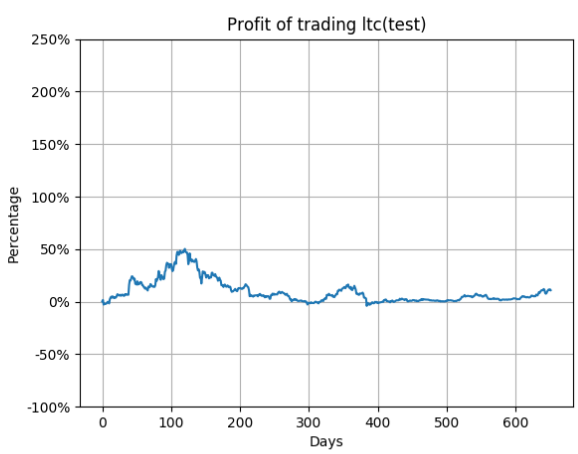

Figure 15: Long term trading for ltc. In this results, we test our trained model by using the recent 600 days price of ltc. As we can see, agent keep making profit but not very good profit

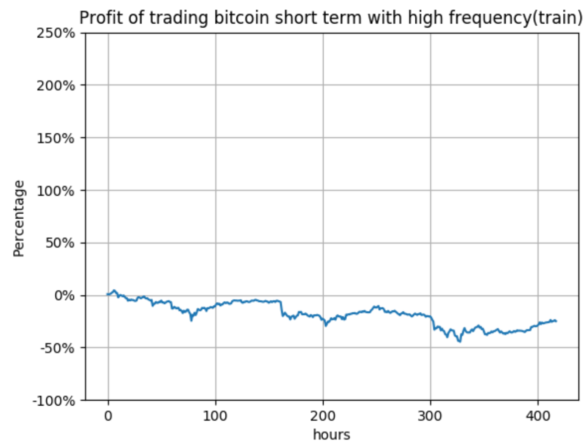

Figure 16: Short term and high frequency trading for bitcoin. In this results, we test our trained model by using the recent 400 hours price of bitcoin. As we can see, agent doesn't make any profit and it keeps losing.

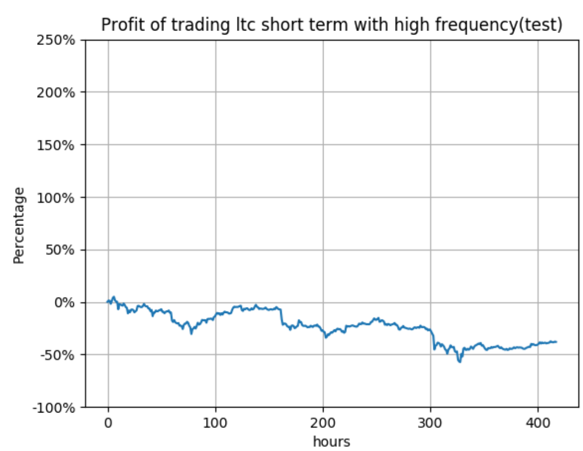

Figure 17: Short term and high frequency trading for ltc. In this results, we test our trained model by using the recent 400 hours price of ltc. As we can see, agent doesn't make any profit and it keeps losing.

Primary Issues

1. Data are hard to be obtained. There are not too much data about some digital currencies that are not popular. Even for the most five popular digital currencies, we can get few data for high frequency trading(hour price history).
2. If the deep neural network structure is complex, it will cost a lot time to do a parameter tuning and train it. Model will be hard to converge.
3. It is difficult to establish a trading environment as possible as closer to the real trading, we have to simplify the trading environment.

Future Improvement

Even if the model can create benefits in long-term transactions, there are still a lot of works to improve the performance of the model:
1.If we have enough time and computation, we can add some more complex layers into deep neural network like CNN and LSTM to strong our model.
2.We can redefine our state space by using more efficient and independent trading indicators or indexes.
3.We can also redefine our reward structure as we did not consider some extreme cases. For example, we use flat to represent all flat states but in reality, a flat state like wide range should be viewed as long-short conversion while here it just be viewed as flat.

Timelines

09/24/2020: Proposal
10/24/2020: Collect and preprocess datasets
11/24/2020: Coding and training Model
12/01/2020: Evaluation and results Analysis
12/06/2020: Final report
12/07/2020: Final presentation rehearse

Team Contributions

In this section, the contribution written here is through out the entire projects. Each of us contributes some part of the whole work.

Team Member #1: Bohan Li

Writing the first draft of the report, putting forward the core ideas, coding and training the model.

Team Member #2: Zhekai Dong

Writing the first draft of the report, reviewing report fixing coding errors and data processing.

Team Member #3: Hyeseong Choi

Code reviewing, report reviewing

References

@onlinewikipedia, author = ”wikipedia”, title = ”Average directional movement index”, ”https://en.wikipedia.org/wiki/Average_directional_movement_index”,addendum ”(accessed: 12.07.2020)”, keywords = ”ADX”
@onlinewikipedia, author = ”wikipedia”, title = ”Relative strength index”, ”https://en.wikipedia.org/wiki/Relative_strength_index”,addendum ”(accessed: 12.07.2020)”, keywords = ”ADX”
@onlinewikipedia, author = ”wikipedia”, title = ”Commodity channel index”, ”https://en.wikipedia.org/wiki/Commodity_channel_index”,addendum ”(accessed: 12.07.2020)”, keywords = ”ADX”