AI-ML GROUP 5

TEAM MEMBERS:

Raj Shah

Preeti Suvarna

Payal Thorat

Pradeep Patwa

Problem Statement: To build a Cryptocurrency Trading Bot which will advise which cryptocurrency to trade on as well as trade for you.

Introduction: There are many varieties of cryptocurrency bots. Investors can subscribe to the bot programs to aid in their cryptocurrency trading. Bots can be incredibly helpful for complete beginners who are new to cryptocurrency and looking to invest and get advice on what to invest at that possible time.

Advantages of Cryptocurrency bots:

1. Saves Time: Monitoring cryptocurrency trends can be really time consuming and stressful especially when you have a lot of money riding on it. Bots will reduce this time and stress considerably.

2. Bots lower the barrier to entry: While crypto trading bots do not guarantee long-term success either, they certainly lower the barrier to entry for newcomers. By utilizing a trading bot to ease their way into crypto trading, newcomers can copy the actions of some of the more successful traders.

Work done in Final Week :

1. In this week we have implemented all the information which we have gathered from last 6 weeks.
2. We learned about basics of blockchain, trading bot, other crypto bot which are available in market, cryptocurrency with its working, and also how to implement those things in codes.
3. We wrote the following python program with the help of Google-Colab for implementation of our code.
4. Here, we have used Coinbase API.
5. The simulation of code is as follows:

Output:

In [ ]:
```
!pip install cbpro
```

```
Collecting cbpro
  Downloading cbpro-1.1.4-py2.py3-none-any.whl (35 kB)
Collecting websocket-client==0.40.0
  Downloading websocket_client-0.40.0.tar.gz (196 kB)
     |████████████████████████████████| 196 kB 11.4 MB/s
Collecting six==1.10.0
  Downloading six-1.10.0-py2.py3-none-any.whl (10 kB)
Collecting requests==2.13.0
  Downloading requests-2.13.0-py2.py3-none-any.whl (584 kB)
     |████████████████████████████████| 584 kB 50.0 MB/s
Requirement already satisfied: sortedcontainers>=1.5.9 in /usr/local/lib/pyth
on3.7/dist-packages (from cbpro) (2.4.0)
Collecting pymongo==3.5.1
  Downloading pymongo-3.5.1.tar.gz (1.3 MB)
     |████████████████████████████████| 1.3 MB 53.7 MB/s
Building wheels for collected packages: pymongo, websocket-client
  Building wheel for pymongo (setup.py) ... done
  Created wheel for pymongo: filename=pymongo-3.5.1-cp37-cp37m-linux_x86_64.w
hl size=362226 sha256=419f13d62e1281e0e48e0208995811c48c45b5cdb04026562a9ee1b
938020252
  Stored in directory: /root/.cache/pip/wheels/96/bd/a9/81eacd9925ebaa01c560b
be29d42a0e1b678bcfb6247e9e3be
  Building wheel for websocket-client (setup.py) ... done
  Created wheel for websocket-client: filename=websocket_client-0.40.0-py2.py
3-none-any.whl size=198296 sha256=8cb2d1f0b24f609d1e911411d2f1b75dcc2a65faf52
45a646cdac2704c8f1e4a
  Stored in directory: /root/.cache/pip/wheels/e6/5d/9f/08e61891b49b66109f5e7
f4760c882a24bb710e8b7391df76d
Successfully built pymongo websocket-client
Installing collected packages: six, websocket-client, requests, pymongo, cbpr
o
  Attempting uninstall: six
    Found existing installation: six 1.15.0
    Uninstalling six-1.15.0:
      Successfully uninstalled six-1.15.0
  Attempting uninstall: requests
    Found existing installation: requests 2.23.0
    Uninstalling requests-2.23.0:
      Successfully uninstalled requests-2.23.0
  Attempting uninstall: pymongo
    Found existing installation: pymongo 3.12.0
    Uninstalling pymongo-3.12.0:
      Successfully uninstalled pymongo-3.12.0
ERROR: pip's dependency resolver does not currently take into account all the
packages that are installed. This behaviour is the source of the following de
pendency conflicts.
tensorflow 2.6.0 requires six~=1.15.0, but you have six 1.10.0 which is incom
patible.
tensorflow-datasets 4.0.1 requires requests>=2.19.0, but you have requests 2.
13.0 which is incompatible.
tensorboard 2.6.0 requires requests<3,>=2.21.0, but you have requests 2.13.0
 which is incompatible.
pandas-datareader 0.9.0 requires requests>=2.19.0, but you have requests 2.1
3.0 which is incompatible.
google-colab 1.0.0 requires requests~=2.23.0, but you have requests 2.13.0 wh
ich is incompatible.
google-colab 1.0.0 requires six~=1.15.0, but you have six 1.10.0 which is inc
```

```
ompatible.
google-api-python-client 1.12.8 requires six<2dev,>=1.13.0, but you have six
 1.10.0 which is incompatible.
google-api-core 1.26.3 requires requests<3.0.0dev,>=2.18.0, but you have requ
ests 2.13.0 which is incompatible.
google-api-core 1.26.3 requires six>=1.13.0, but you have six 1.10.0 which is
incompatible.
dm-tree 0.1.6 requires six>=1.12.0, but you have six 1.10.0 which is incompat
ible.
datascience 0.10.6 requires folium==0.2.1, but you have folium 0.8.3 which is
incompatible.
albumentations 0.1.12 requires imgaug<0.2.7,>=0.2.5, but you have imgaug 0.2.
9 which is incompatible.
Successfully installed cbpro-1.1.4 pymongo-3.5.1 requests-2.13.0 six-1.10.0 w
ebsocket-client-0.40.0
```

In [ ]:
```
import cbpro
```

In [ ]:
```
public='157fb0e81b6149e379e3f490d0df2a56'
passphrase='kui6tbg8ek'
secret='ZLvQ2viJJGKYAVwVEfETi3m7eWT+VpTlAZvG5PSLI9Yev/w4ubwNCl5W5mEkcQJ+P1PXWB
dgBRLqPyq8HYJR3Q=='
```

In [ ]:
```
url = 'https://api-public.sandbox.pro.coinbase.com'
client = cbpro.AuthenticatedClient(
    public,
    secret,
    passphrase,
    api_url=url)
```

In [ ]:
```
import time
```

In [ ]:
```
sell_price= 3000
sell_amount= 0.3
```

In [ ]:
```
buy_price= 25000
sell_amount=0.2
```

In [ ]:
```
while True:
  price = float(client.get_product_ticker(product_id="BTC-EUR")['price'])
  if price <= buy_price:
    print("Buying BTC")
    client.buy(size=buy_amount, order_type="market",product_id="BTC-EUR")
  elif price >= sell_price:
    print("Selling BTC")
    client.sell(size=sell_amount, order_type="market", product_id="BTC-EUR")
  else :
    print("Nothing price is {price:,}")
  time.sleep(10)
```

```
Selling BTC
Selling BTC

---------------------------------------------------------------------------
KeyboardInterrupt                         Traceback (most recent call last)
<ipython-input-26-a326808ff277> in <module>()
      9   else :
     10     print("Nothing price is {price:,}")
---> 11   time.sleep(10)

KeyboardInterrupt:
```

In [ ]:
```
!pip install -q pyngrok
!pip install -q streamlit
!pip install -q streamlit_ace
```

```
     |████████████████████████████████| 2.6 MB 6.8 MB/s
```

Conclusion:

With the help of Artificial Intelligence ,we tried to make a Trading Bot which will be able to trade varieties of cryptocurrency by suggesting what to buy or not and which will trade for us.

GitHub Repositories:

https://github.com/23rajshah/tradingbotforcrypt

References:

https://www.youtube.com/watch?v=a74pQbHgdXw&list=WL&index=2

https://www.nerdwallet.com/article/investing/blockchain

https://www.nerdwallet.com/article/investing/cryptocurrency-7-things-to-know

Payal Sanjay Thorat:

LinkedIn: https://www.linkedin.com/in/payal-thorat/

Github: https://github.com/payalthorat25

Raj Shah:

LinkedIn:

https://www.lin

kedin.com/in/raj

-shah-

62578b191

GitHub:

https://github.co

m/23rajshah

Pradeep Patwa:

LinkedIn : https://www.linkedin.com/in/pradeep-patwa/ Github

: https://github.com/pradeeppatwa

Preeti Suvarna:

Linkedin: https://www.linkedin.com/in/preeti-suvarna-b5b409193/

Github: https://github.com/preeti-suvarna