High Frequency Trading

Abou SANOU
Data Engineer / Counter of Investment / Bpifrance

Table of Contents

HFT Overview!

Some basics

- An exchange is a marketplace where securities, commodities, derivatives and other financial instruments are traded. [1]
- Market Data : The generally adopted stock market data is the sequence at regular intervals of time such as the price open, close, high, low, and volume
- How securities price is defined : **Law of supply and demand**
- A stock exchange is used to raise capital for companies seeking to grow and expand their operations. The first sale of stock by a private company to the public is referred to as an initial public offering (IPO).
- Portfolio : A portfolio is a **collection of financial investments** like stocks, bonds, commodities, cash, and cash equivalents, as well as their fund counterparts.
- **Modern Portfolio Theory(Markowitz's theory)** is a method that can be used by risk-averse investors to construct diversified portfolios that maximize their returns without unacceptable levels of risk.

[1] Aldridge, I., Krawciw, S., 2017. Real-Time Risk: What Investors Should Know About Fintech, High-Frequency Trading and Flash Crashes. Hoboken: Wiley.

What is HFT ??

- Trading of equities, options, futures at high speed in large volumes
- Earning money by exploiting the fleeting variation in stock price or demand

HFT accounts

- 70% of all trades in US Markets[1]
- And it continues to grow

HFT involves

- Using computers to place orders based on “predefined” algorithms

[1] Aldridge, I., Krawciw, S., 2017. Real-Time Risk: What Investors Should Know About Fintech, High-Frequency Trading and Flash Crashes. Hoboken: Wiley.

Main challenges of HFT

- Scalability
- Latency
  - Execute orders faster than other investors to capture fleeting variations in price and demand in the markets
- Throughput
  - Handle large volume of orders
- Flexibility
  - Adapt to changing risks and trading strategies

[1] Aldridge, I., Krawciw, S., 2017. Real-Time Risk: What Investors Should Know About Fintech, High-Frequency Trading and Flash Crashes. Hoboken: Wiley.

Recent Problems in HFT

- Knight Capital bug[2]:
  - Test script executed live trades
  - $450M loss in 45 minutes
- Nasdaq - Facebook IPO[3]
  - Order confirmations delayed
  - $62M loss in direct damages
- BATS Failure
  - Software bug in order auctions

[2] https://www.bugsnag.com/blog/bug-day-460m-loss
[3]: https://www.sec.gov/news/press-release/2013-2013-95htm

More precisely

HFT tries to :

- Buy low
- Sell High
- In high speed context
- And high scalability context

Archi Overview!

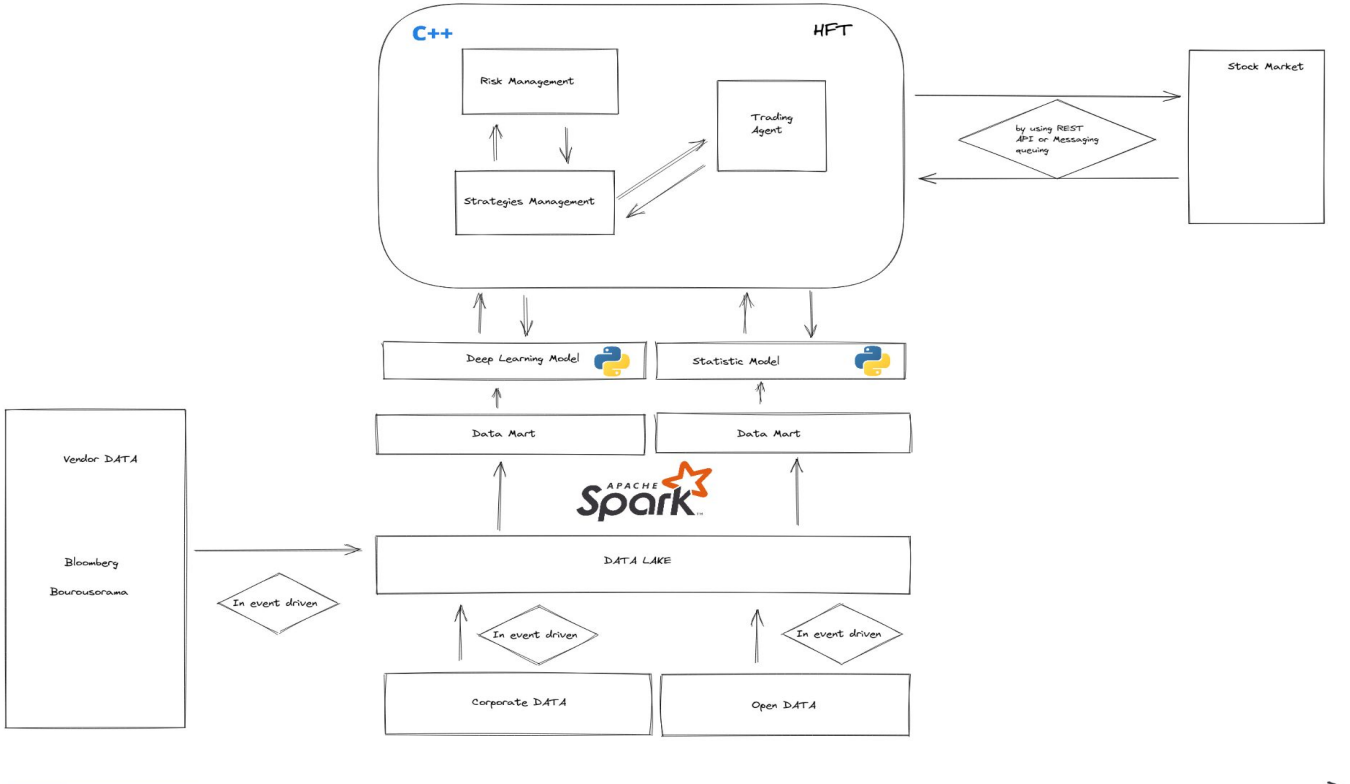
Protocol to interact with exchange

- Generally by rest api
- Also possible by pushing event in topic(event streaming)
- Some stock were created its own protocol on top of http
  Like Nasdaq : Nasdaq TotalView-ITCH[3]

[3]: http://www.nasdaqtrader.com/content/technicalsupport/specifications/dataproducts/NQTVITCHSpecification.pdf

Architecture

[3]: http://www.nasdaqtrader.com/content/technicalsupport/specifications/dataproducts/NQTVITCHSpecification.pdf



kafka

RedPanda for Streaming

[3]: http://www.nasdaqtrader.com/content/technicalsupport/specifications/dataproducts/NQTVITCHSpecification.pdf

Take care about Chinese Wall

As we can remark, A whole component for HFT, will be built in C++;

- Memory Management
- Scalability

[3]: http://www.nasdaqtrader.com/content/technicalsupport/specifications/dataproducts/NQTVITCHSpecification.pdf

Heterogeneity of data

The highest-frequency data is a collection of sequential "ticks," arrivals of the latest quote, trade, price, and volume information. Tick data usually has the following properties:

A timestamp, A financial security identification code, An indicator of what information it carries: bid price, ask price, available bid volume, available ask volume, last trade price, last trade size

Option-specific data, such as implied volatility The market value information, such as the actual numerical value of the price, available volume, or size.

[3]: http://www.nasdaqtrader.com/content/technicalsupport/specifications/dataproducts/NQTVITCHSpecification.pdf

Risk Management Component

A risk management strategy is an important tool for traders. It helps to identify and manage risk, set limits on position sizes, and set up solid trading opportunities. A good risk management strategy can help to protect capital, prevent losses, and maximise profits.

There are many modelling for the risk :

- Value at Risk
- Stochastic modelling
- Deep Risk Model[4]

[3]: http://www.nasdaqtrader.com/content/technicalsupport/specifications/dataproducts/NQTVITCHSpecification.pdf
[4]: https://arxiv.org/pdf/2107.05201.pdf

Strategy Component

- A trading Strategy is a set of rules to apply;
  a trading strategy is a fixed plan that is designed to achieve a profitable return by going long or short in markets.

- The aim of this component will be to generate according to the context of market and risk factor of market, strategies to apply in order to achieve a return.

[3]: http://www.nasdaqtrader.com/content/technicalsupport/specifications/dataproducts/NQTVITCHSpecification.pdf

Trading Agent Component

- The trading agent follow **PEAS**[5] paradigm.

PEAS For Performance Measure, Environment, Actuator, Sensor
This Paradigm is more sustainable because it can provide rational agent
This model allows you to keep the strategies and risk component separated

- We can use Reinforcement Learning for the Trading Agent.
  - More difficult to separate a unit of domain (Risk and Strategy)
  - Less use in production, problem a explaining.
  -

[5]: http://aima.cs.berkeley.edu/

Formalize the Trading World in PEAS approach

- PERFORMANCE MEASUREMENT : PnL (Profit and Loss)
- Environment : Stochastic
- Actuator : Be able to buy/ SELL or Hold
- Sensor : tools to get market parameter or knowledge from latent space

[3]: http://www.nasdaqtrader.com/content/technicalsupport/specifications/dataproducts/NQTVITCHSpecification.pdf

Formalize the Trading Agent in (Deep) Reinforcement Learning approach

- “State” are market parameters
- Actions: buy/sell/hold
- Reward : PnL
- Goal : Maximise reward

[3]: http://www.nasdaqtrader.com/content/technicalsupport/specifications/dataproducts/NQTVITCHSpecification.pdf

Some important notions in (D)RL

- Principe According to the **Bellman Equation**, long-term-reward in a **given action is equal to the reward from the current action combined with the expected reward from the future actions** taken at the following time.
- Policy function π : maps state to action, and policy optimization is to find an optimal mapping.
- Value function V: estimates *how good* it is for the agent to be in a given state.
  Other version of this function more subntable in trading context estimates how good it is to perform a given action in a given state.

In Trading Context, The Value Functions can be estimated from experience.

[3]: http://www.nasdaqtrader.com/content/technicalsupport/specifications/dataproducts/NQTVITCHSpecification.pdf

Some important notions in (D)RL

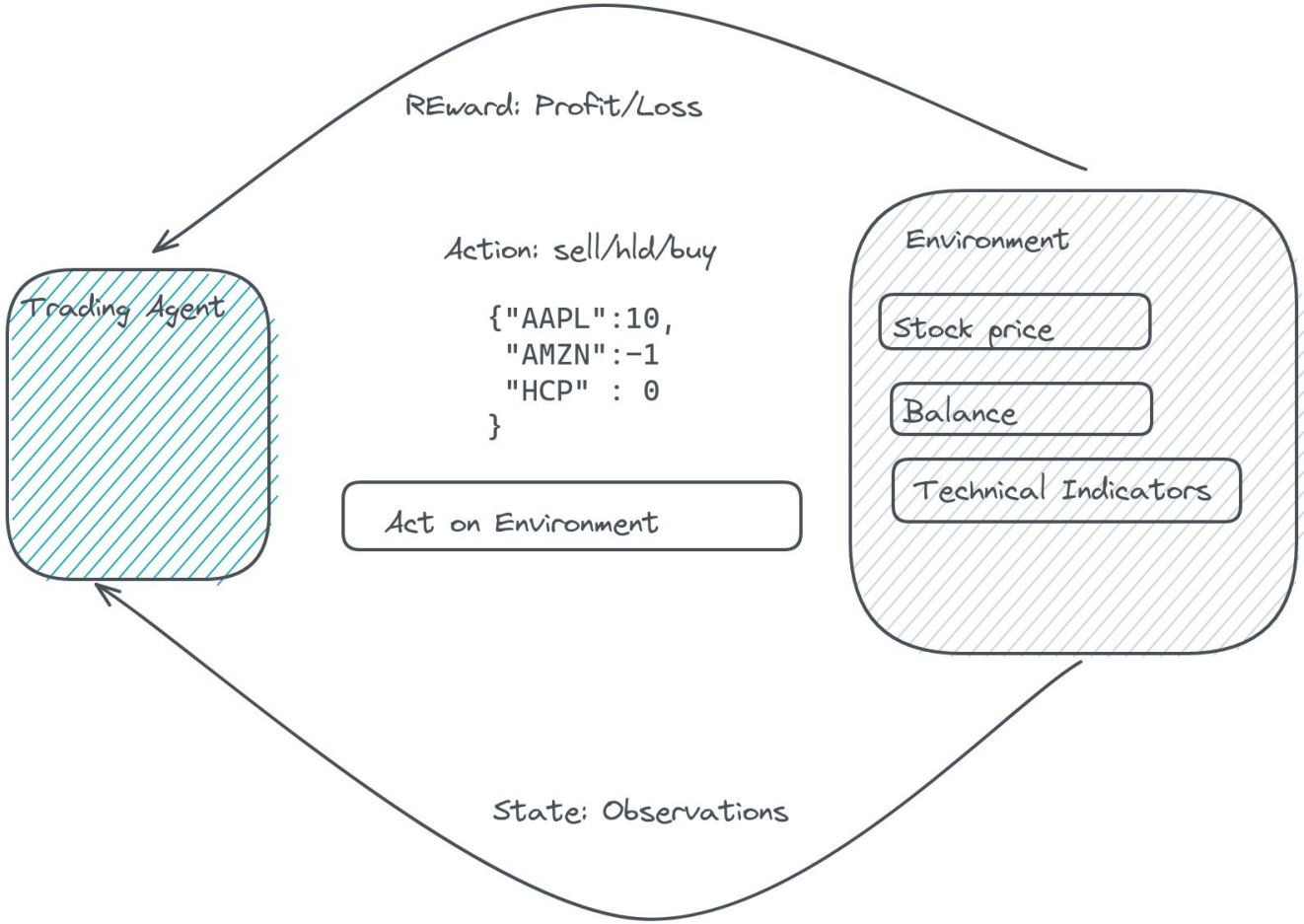
- Episode : one a sequence of states, actions and rewards, which ends with terminal state.

[3]: http://www.nasdaqtrader.com/content/technicalsupport/specifications/dataproducts/NQTVITCHSpecification.pdf

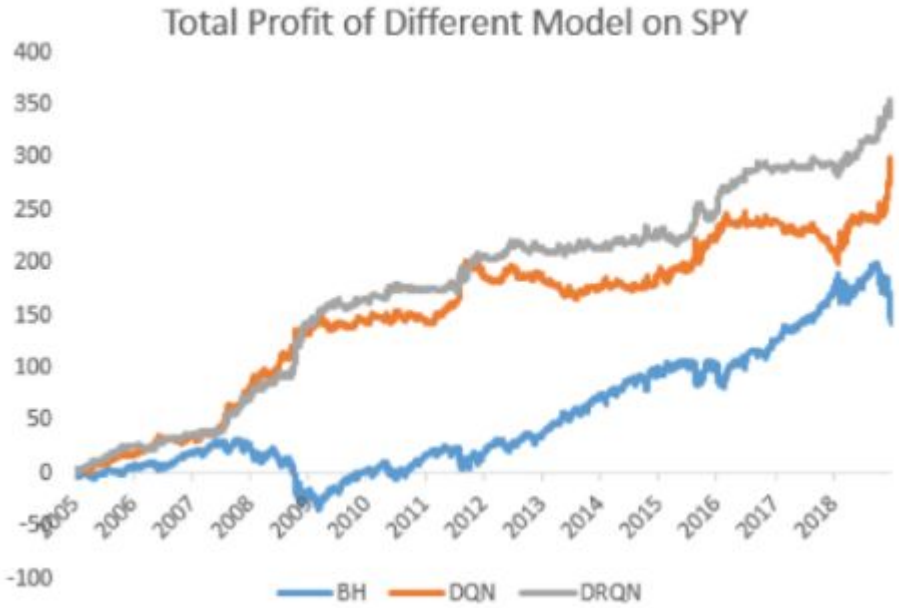
The reward function

- PnL
- Profit per tick
- Sharpe ratio
- Punishment for long hold times

[3]: http://www.nasdaqtrader.com/content/technicalsupport/specifications/dataproducts/NQTVITCHSpecification.pdf

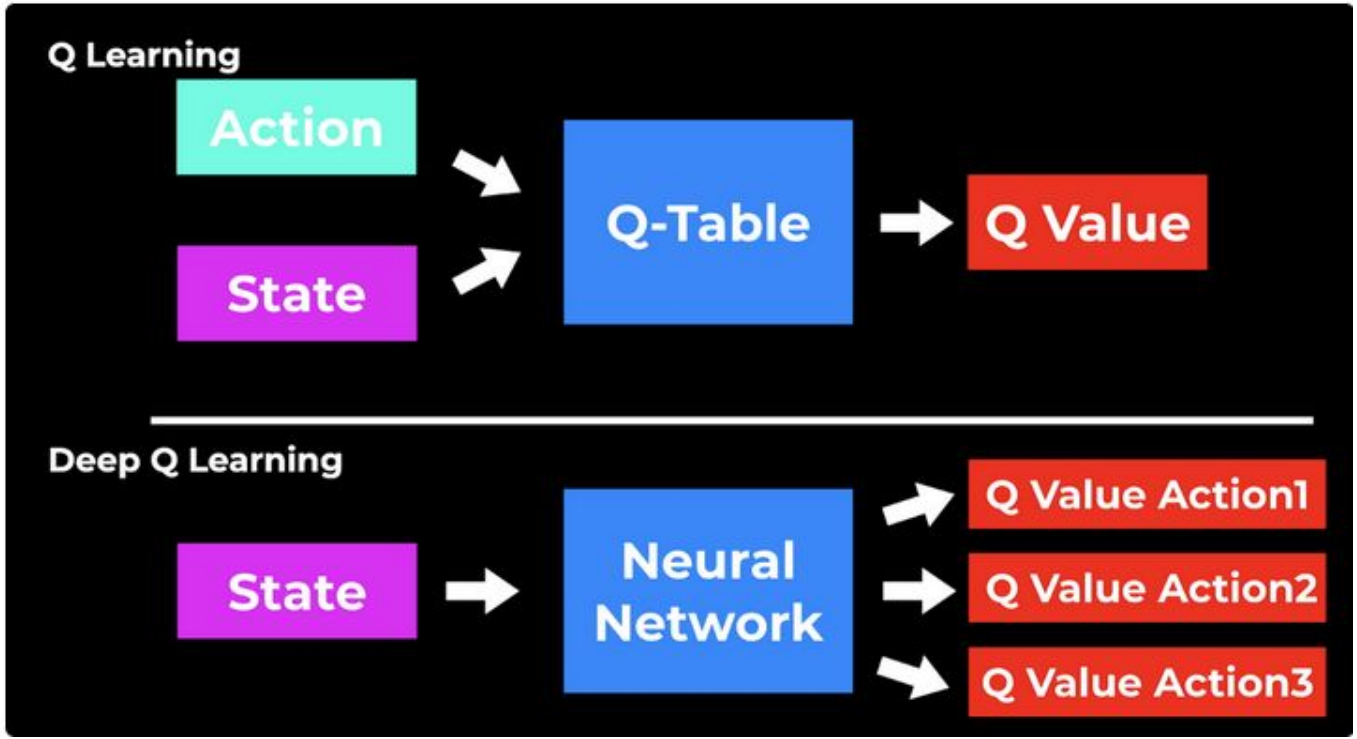


[3]: http://www.nasdaqtrader.com/content/technicalsupport/specifications/dataproducts/NQTVITCHSpecification.pdf


Total Profit of Different Model on SPY
400
350
300
250
200
150
100
50
0
-50
-100
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
BH
DQN
DRQN

More deeply in DRL

- Q-Learning[1]
  - It is a value based approach based on the so called Q-Table.
  - The **Q-Table calculates the maximum expected future reward, for each action at each state**. With this information we can then choose the action with the highest reward.
- Deep Q-Learning:
  - Deep Q Learning uses the Q-learning idea and takes it one step further. Instead of using a Q-table, **we use a Neural Network** that takes a state and approximates the Q-values for each action based on that state.
- Deep Recurrent Q-Learning
  - this is a kind of model that adds Recurrent Network layers to original DQN.
  - Indeed, DQN is a powerful model for many problem domains. However, they are limited in the sense that they learn a mapping from a limited number of past states. This is a scenario where Recurrent Network can be reasonably applied. Specifically, to implement Deep Recurrent Q-Network, we may add an LSTM layer to remember states from the past.


Q Learning
Action
State
Q-Table
Q Value
Deep Q Learning
State
Neural Network
Q Value Action1
Q Value Action2
Q Value Action3

Cons RL in HFT

- Difficulties to apply **Modern Portfolio Theory**
- RL needs a lot of realistics data and a lot of computation.

REX

Form my previous experiences :

- Companies avoid to use architectures based on unexplained models.
- Data Quality make difference.
- Don't try to follow the trend on ML World.
- Separate the system in many component as well as possible in order to improve easily each component and to affect dedicated team to each component.
- We don't need to be irrational on risk.
- The best results in trading environment came from Deep Reinforcement Learning.

References

[1]: Aldridge, I., Krawciw, S., 2017. Real-Time Risk: What Investors Should Know About Fintech, High-Frequency Trading and Flash Crashes. Hoboken: Wiley.
[3]: http://www.nasdaqtrader.com/content/technicalsupport/specifications/dataproducts/NQTVITCHSpecification.pdf
[5]: Deep Reinforcement Learning in Quantitative Algorithmic Trading: A Review, Tidor-Vlad Pricope, 2022
[6]: Sutton, Richard S., and Andrew G. Barto. *Reinforcement learning: An introduction*. MIT press, 2018.
[7]: https://github.com/yabdellah/Finance
[8]: https://lilianweng.github.io/posts/2018-04-08-policy-gradient/
[9]: https://www.gymlibrary.dev/

[3]: http://www.nasdaqtrader.com/content/technicalsupport/specifications/dataproducts/NQTVITCHSpecification.pdf

Thanks