Literature Review

February 8, 2019

1 Deep Neural Network

- **Title**: Deep Learning for Spatio-Temporal Modeling- Dynamic Traffic Flows and High Frequency Trading
- **Author**: Matthew F. Dixon, Nicholas G. Polson, Vadim O. Sokolov
- **Source**: arXiv preprint arXiv:1705.09851
- **Year**: May 7, 2018

This paper focus on the application of **Recurrent Neural Network** to high frequency trading in predicting the mid-price movement. Using both the limit order and market order data from the order book, this paper have created the mid-price label (response variable) of three classes {-1, 0, 1}, in which -1 means down-tick of price, 0 means unchanged price and 1 means up-tick. They have also constructed 32 features from the order book and then trained their **RNN** model to predict the price movement. As a result, the model's performance in precision, recall and f1 score all exceeds the traditional methods such as logistic regression, elastic-net and etc, which demonstrate the ability of RNN to capture the non-linear relationship across the vast database.

This paper mainly use the Recurrent Neural network model and train the model by performing **Stochastic Gradient Descent** (**SGD**) to learn the parameters in each hidden layer in the network model. They also apply the **Drop-out** methods (**DO**) to build the model architecture by choosing the number of hidden layers and neurons in each layer and also perform variable selection. Figure 1 is an example of choosing the number of layers, where in the table are f1 scores of different model with different layers, we can see that 10 is the optimal number since after 10, the score of all the three classes decrease.

Going forward, we can learn the way this paper extract data from the order and construct features as our input. Besides we can partially follow their drop-out method to determine the architecture of our own model such as how many layers, how many neurons in each layer and how to select variables.

n_steps	$\hat{Y} = -1$	$\hat{Y} = 0$	$\hat{Y} = 1$
1	0.089	0.528	0.091
2	0.097	0.837	0.103
5	0.131	0.848	0.120
10	0.165	0.881	0.161
20	0.163	0.878	0.162
50	0.151	0.836	0.144
100	0.118	0.807	0.115

Figure 1: F1 score for different number of layers

2 Reinforcement Learning in Trading

- **Title**: Performance Functions and Reinforcement Learning for Trading Systems and Portfolios
- **Author**: John Moody, Lizhong Wu, Yuansong Liao, Matthew Saffell
- **Source**: Journal of Forecasting, Volume 17, Pages 441-470
- **Year**: 1998

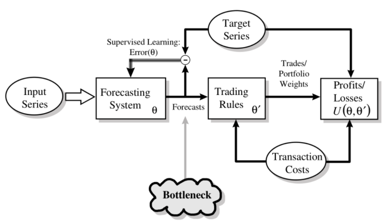
Key point of this paper:

Firstly, the author states that Reinforcement Learning (RL) based trading system should performs better than supervised learning based and labelled data based trading system. Because the latter apply optimization on intermediate quantities (loss function) rather than utility function which investors care the most. However, reinforcement learning can optimize utility function through applying gradient ascent on parameters of policy function. See figure 2.

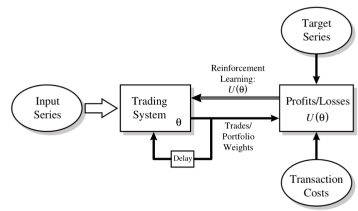
Secondly, the author gives a recursive expression of policy function $F_t(\theta; F_{t-1}, I_t)$, where θ is the set of parameters and I_t is the price information set. Then this paper gives the derivation of policy gradient of Recurrent Reinforcement Learning (RRL) by using chain rule.

Another point is a new proposed measurement of value function (utility function)–Differential Sharpe Ratio. Moody firstly uses Exponential Moving estimates to estimates the first two moments of rate of return. Then he applied Taylor expansion on EMA Sharpe Ratio to obtain the expression of Differential Sharpe Ratio. The increment of this ratio along with time can be simply computed by rate of return and some predetermined constant, which will dramatically reduce computation cost. It also emphasizes the influence of recent rate of return.

To adapt this reinforcement learning model to High Frequency Trading (HFT), we need to take market impact and transition cost into consideration. Placing limit order instead of market order is crucial in HFT. Therefore, we also need to consider how to model continuous action space. Additionally, we can improve the current policy gradient algorithm by applying policy optimization due to recent research on reinforcement learning in control field. Moreover, this paper doesn't specify the model of parameterized policy function, we can utilize current research on deep neural network to parameterize our policy function properly.



(a) Supervised learning based trading system



(b) Reinforcement learning based trading system

Figure 2: Comparison the workflow of supervised learning based trading system (a) and reinforcement learning based trading system (b). Supervised learning will apply optimization on its objective function. However, we want to optimize utility function eventually. Reinforcement learning can directly optimize utility function through parameterized policy function.

3 Modeling on High Frequency Trading

- **Title**: Machine Learning for Market Microstructure and High Frequency Trading
- **Author**: Michael Kearns, Yuriy Nevmyvaka
- **Source**: High Frequency Trading: New Realities for Traders, Markets, and Regulators
- **Year**: 2013

Machine learning approaches for high frequency trading are studied in the paper. The two major contributions of the paper are **optimized trade execution** and **predicting price movement**. **Reinforcement learning** were used to model the microstructure features. By defining the state space (v, t) and some additional features capturing order book state, the reward function of trading cost was improved. Moreover, by comparing with one-shot submission strategies, the RL perform well with lower trading cost. Visualize value function is an effective way to witness the price movement. See figure 4.There is a clear strategy transform (momentum to reversion) from short term to medium term (second to minutes). However, when the holding period extend to longer horizon, for example 30 minutes to hours, the microstructure features becomes less informative and simply learn to sell in every state. The paper adjusted the learning algorithm to reduce the long-term directional price drift by changing the evaluation from total profit to relative profit. Then the action learned go back to mean reversion.

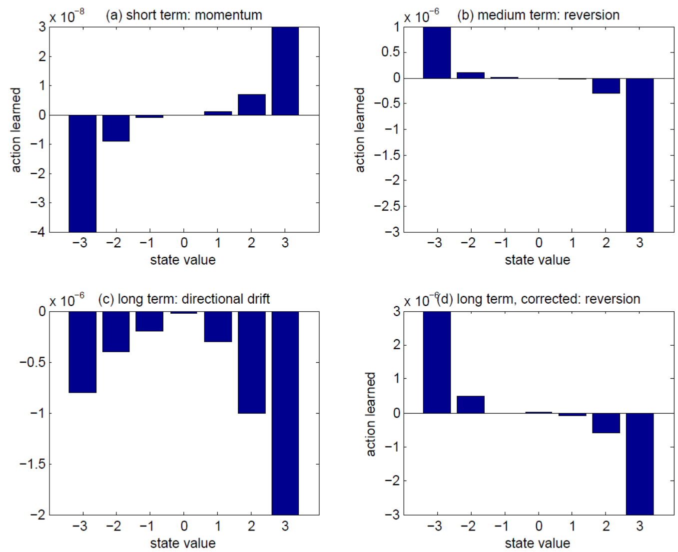


Figure 3: Learned policies depend strongly on timescale.

In conclusion, there is no easy path for machine learning to gain profitability and improve execution. Focus on **feature design and engineering** and all kinds of other fine-tuning are necessary to obtain better results.

4 Inverse Reinforcement Learning

- **Title**: Behavior Based Learning in Identifying High Frequency Trading Strategies
- **Author**: Steve Yang, Mark Paddrik, Roy Hayes, Andrew Todd, Andrei Kirilenko, PeterBeling, and William Scherer
- **Source**: Computational Intelligence for Financial Engineering Economics (CIFEr),2012 IEEE Conference on (pp. 1-8). IEEE.
- **Year**: March, 2012

Obtain reward function Optimal policy

- MDP Model for limit order book to obtain reward function
- Inverse Reinforcement Learning approached by linear programming formulation to optimal policy

Going Forward

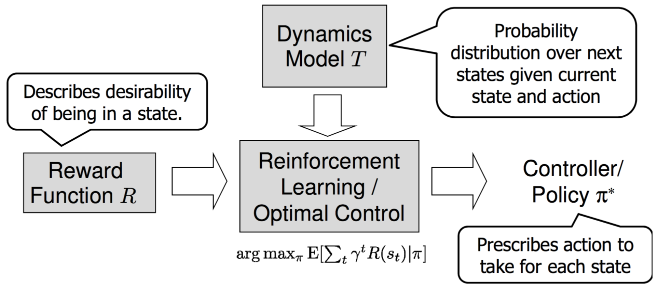


Figure 4: Working principle of IRL model

- Dealing with limit order book with MDP
- Policy Optimization by IRL Model

This paper is purposed to identify High Frequancy Trading strategies from other strategies. Two major idea in this paper is the MDP model has approached to dealing with limit order book and the IRL model is approached by linear programming formulation to working on policy optimization issue. The MDP model provided the state and action definition with several features and different action descriptions, which inspired us to consider how to use further method to deal with limit order book.

The IRL model suggested the new way to optimal the policy instead the policy gradient.