IS5006 INTELLIGENT SYSTEM DEPLOYMENT

RESEARCH ON MAS & CBR

Prepared by:

Group 7

Name	Matriculation Number
Ge Xiaomeng	A0112747H
Goh Khai Hong	A0212197H
Shen Siyuan	A0112489B
Xue Bin	A0039717X
Yong Chee Xian Matthew	A0090988J
Zou Yang	A0070179A

Semester 2, AY2020/2021

NATIONAL UNIVERSITY OF SINGAPORE

Table of Content

1. Executive Summary

The group's project is to build an algo trading Multi-Agent System (MAS) for cryptocurrency trading. In this paper, we will present our research findings on MAS with a focus of application of MAS in algo trading as well as the state-of-art models for Case-Based Reasoning (CBR). The second half of this paper is to review the system components of MAS and CBR in the group project and ensure our project incorporates the latest research findings. At the end of this paper, we will evaluate the pros and cons of the solutions and provide our view - we can adopt MAS and CBR in algo trading on the condition of proper risk management controls in place.

2. Research about MAS

2.1. Literature Review on MAS

Multi-Agent System (MAS) is 'the collection of autonomous agents situated in a certain environment, responding to their environment dynamic changes, interacting with other agents, and persisting to achieve their own goals or the global system goals' (Ahmed Abbas, 2015).

2.1.1. Agents & Environment

Simply put, MAS is a system formed by multiple agents interacting in a common environment. Agents, which have autonomy, are created to perform actions to achieve specific goals 'based on information (sensors, feedback) received from the environment'(Panait & Luke, 2005) but each agent has no full knowledge of the whole system.

2.1.2. MAS Organization

There are two main views on how a MAS should be structured - agent-centered and organization-centred. Agent-centred MAS are created with a bottom-up approach where agent behaviors and interactions are prioritized over overall system structure.Despite concerns on unpredictable results, the approach is particularly useful when designers like to observe patterns from individual agents iterating in a common environment. Organization-centred MAS is the top-down approach where designers focus more on the global structure than local agents whose actions are coordinated by rules (Ahmed Abbas, 2015).

2.1.3. Applications of MAS

MAS has a wide range of applications in both academic fields and industry such as communication, organization, deep learning, robotics, cybersecurity, transportations, production or assembly systems (Leitão & Karnouskos, 2015). In recent years, the use of blockchain technologies for MAS is also gaining focus (Calvaresi et al., 2018).

2.2. Algorithmic Trading Using Intelligent Agents

2.2.1. Algorithmic trading background

Algorithmic trading utilizes the computer algorithm to decide the characters, such as price and quantum of different order, and then use programs to automatically place trading orders. The adoption of algorithmic trading is growing extremely fast and especially in the currency market, also known as the Forex Market. This paper (Barbosa & Belo, 2008) describes a mechanism to implement autonomous Forex trading agents using an infrastructure entirely based on artificial intelligence methodologies. Unlike the use of hybrid intelligent systems or neural networks in trading that focus only on the prediction accuracy, the main objective of the system is to maximize the profit while minimize the risk (drawdown) when each agent is doing the real-time trading autonomously without requiring human intervention.

2.2.2. Proposed infrastructure

An infrastructure for implementing hybrid intelligent trading agents that can function autonomously have to be equipped with the ability to make a series of decisions. Figure 2.2.1 from the paper demonstrates the overall infrastructure which defines three interconnected modules:

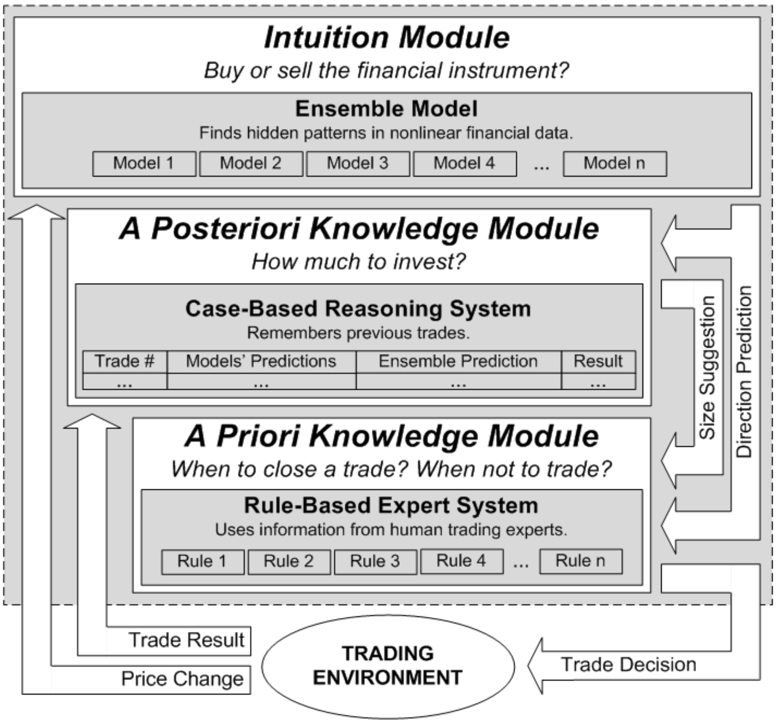

Figure 2.2.1. Infrastructure for implementing trading agents

1) Intuition Module

The module predicts the moving trend of the price for a pair of currency. This prediction is made by an Ensemble Model, which consists of
several classification and regression models that aim to improve the prediction results. This creates a weighted voting system, where the weight of each vote is based on the model's profitability, corresponding to one of two classes: "the price will go up" or "the price will go

down”. Ensemble model splits data into two sets, test set and training set, and the model will be retrained with new data so that the agents can keep learning.

2) A Posteriori Knowledge Module

This module is essentially to determine how much to buy or sell after deciding whether it is a buy signal or sell signal in the intuition module. This is done by optimizing the investing profit. One way to implement this is to double the investment when a trade is expected to be very profitable, invest a normal amount if a trade is expected to have moderate profit, and skip trades that are expected to be unprofitable. Case-based reasoning system can be used here to determine profitability.

3) A Pri Knowledge Module

This module is for making the final decision based on the predefined rules in Rule-Based Expert System, such as the settings for take-profit orders and stop-loss orders, skip trades in low liquidity days. The agents function autonomous by invoking broker APIs to implement the final trade decisions.

2.3. Application of CBR in financial forecasting

2.3.1. Conventional CBR - Composite neighbours

Case-based reasoning (CBR) allows a learning system to make increasingly useful decisions as it accumulates experiences. An important advantage of using CBR is its affinity to human learning. In relation to that affinity, is the ease of enhancing a system performance. Generally, it allows knowledge in a particular domain to be stored in formats that are conventional for the domain. Moreover, CBR can be effective even with an imperfect knowledge base.

Two key challenges typically arise from the task of cases retrieval, these are:

- Matching problem - Associating a new problem to a pertinent past case.
- Indexing Problem - Task of storing cases for effective and efficient retrieval.

CBR works by seeking the nearest neighbor to a target case. The key to the composite approach lies in determining the most effective set of weights used for constructing the neighbors. Simulated annealing is often used to obtain the optimal weights.

Step1. Begin with current case $x(t)$.

Step2. Seek the J neighboring cases $x(t_i)$ in the past which are closest to $x(t)$ according to the distance function: $d_i \equiv d[x(t_i), x(t)]$

Step3. Compute the sum of weights :

$$d_{TOT} = \sum_{i=1}^{J} d_i$$

Step4. Determine the relative weight of i^{th} neighbor : $w_i = \frac{1}{J-1}\left[1 - \frac{d_i}{d_{TOT}}\right]$

Step5. Find the successor $x(t_i + 1)$ of each case $x(t_i)$ in the set of neighbors.

Step6. Calculate the forecast for t+1 as the weighted sum of successors :

$$\hat{x}(t+1) = \sum_{i=1}^{J} w_i x(t_i + 1)$$

Figure 2.3.1. Procedure for CBR using composite neighbors

2.3.2. Random walk model vs. Implicit learning technique (NN) vs. explicit approach (CBR)

A case study conducted in 2004 (Chun & Kim, 2004) compared the return against risk output of the Polish and Korean stock market with 3 approaches.

- Random walk model - Buy Hold Strategy
- Active trading strategy - Backpropagation Neural Net (BPN)
- Active trading strategy - CBR

The resulting conclusion shows the active trading strategies outperforming the random walk model, at times by a wide margin. This supports the case that superior returns by coupling learning systems with active trading strategies is attainable.

A point to note, BPN models do suffer from protracted training periods. In order to reach a satisfactory performance, numbers runs are required. The time and effort required for training have therefore hindered their widespread application to practical domains. Whereas, CBR will offer much swifter response.

2.3.3. Dynamic Adaptive Ensemble CBR (combination of parameters)

An improvement to Conventional CBR is the notion of adding a Dynamic adaptive ensemble (DAE). DAE CBR requires finding combinations of parameters, then updating and applying an optimal CBR model to the application or domain area.

The results from the 2005 paper (Chun & Park, 2005) below shows that DAE CBR was significantly better than other models in the hit rate measure and the Mean Absolute Percentage Error (MAPE).

```
Step 1.  Perform exploratory data analysis (EDA): identify overall patterns and outliers.
Step 2.  For each variable X(t) {
         LX(t) = ln X(t)
         DLX(t) = LX(t) - LX(t-1)
         ZDLX(t) = (DLX(t) - m_x)/s_x
         }
Step 3.  Perform CBR machine for the test periods.
         a. Choose an evaluation criterion such as hit rate, MAPE, RMSE
         b. Select the learning windows and test windows.
Step 4.  Find an optimal CBR model for each test period l=h,...,t
          // t is the end of the test windows
          For i=1,..., # of distance_method
            For j=1,..., # of nearest neighbor
              For k=1,..., # of context_vector
                For l= h,...,t
                 {Forecast ZDLX(l) using Ensemble CBR parameter i,j,k:
                  ZDLX(l) = Ensemble_CBR(i,j,k)
                  Calculate the evaluation results
                  }
         Select optimal CBR parameter i_best,j_best, k_best
Step 5.  Perform a DAE CBR machine for each validation period v=t+1,...,z
        //z is the end of the test windows
         For v= t+1,...,z
             { ZDLX(v) = Ensemble_CBR(i_best,j_best, k_best)
                       //for a t+1 forecast v ← t+1
               Detransform the result {
                  DLX(v) = s_x* ZDLX(v) + m_x
                  LX(v) = DLX(v) + LX(v-1)
                  X(v) = exp[LX(v)]
                  }
              Go to Step 3.b and reset test windows
              Do Step 4 to find an optimal CBR model for a t +2 forecast}
Step 6.  Analyze the result and End DAE CBR.
```

Figure 2.3.3. The procedure for DAE CBR

2.3.4. Regression CBR (weights)

The other improvement to Conventional CBR in the notion of adding a Regression.
The regression case based reasoning (RCBR) affix different weightage to independent variables before finding similar cases. This differs in that independent variables are not assumed to be identical in their importance.

The RCBR uses a regression analysis to find relative importance of independent variables from the relationship between independent variables and a dependent variable. It then puts the relative weights using regression coefficients on independent variables. Nearest neighbors or similar cases using weighted independent variables are selected through the traditional CBR machine and dynamically updates the weightage for the next target case and performs the traditional CBR machine again.

The results from the 2006 paper (Chun & Park, 2006) below shows that the RCBR was significantly better than random walk and conventional CBR models in the hit rate measure and MAPE.

1. Perform exploratory data analysis (EDA): identify overall patterns and outliers.
2. Transform data for comparability.
   a. Convert indices (e.g. Stock Price Index) and volume by logarithmic mapping:
      $X_{ij} \rightarrow LX_{ij}$: for variable $i = 1..I$ and case $j = 1..J$
   b. Take differences to eliminate trend if appropriate
      $U_{ij} \rightarrow DU_{ij}$
   c. Standardize: eliminate effects of units (of measurement) by subtracting mean and dividing by standard deviation:
      $V_{ij} \rightarrow ZV_{ij} \equiv Z_{ij}$
3. Find the weight vector using regression analysis.
   a. Determine a learning window.
   b. Find weights to apply independent variables using OLS method.
   c. Apply new weights to independent variables.
4. Perform a CBR machine for t+1 prediction.
   a. Find the optimal CBR model considering distance functions, the number of nearest neighbors and series of context vector.
   b. Perform t+1 prediction using the final optimal model.
5. Perform a CBR machine from t+1 to t+n predictions.
   a. Do loop from 3 to 4 until a final period of prediction in the test phase.
      → Start to evaluate a new learning window.
      → Dynamically updates new weights and apply them to independent variables.
6. End a regression Case Based Reasoning
   → Calculate result from the model.
7. Detransform the variables.
   → Analyze results.

Figure 2.3.4. The procedure for RCBR

3. Algo MAS and CBR System Components

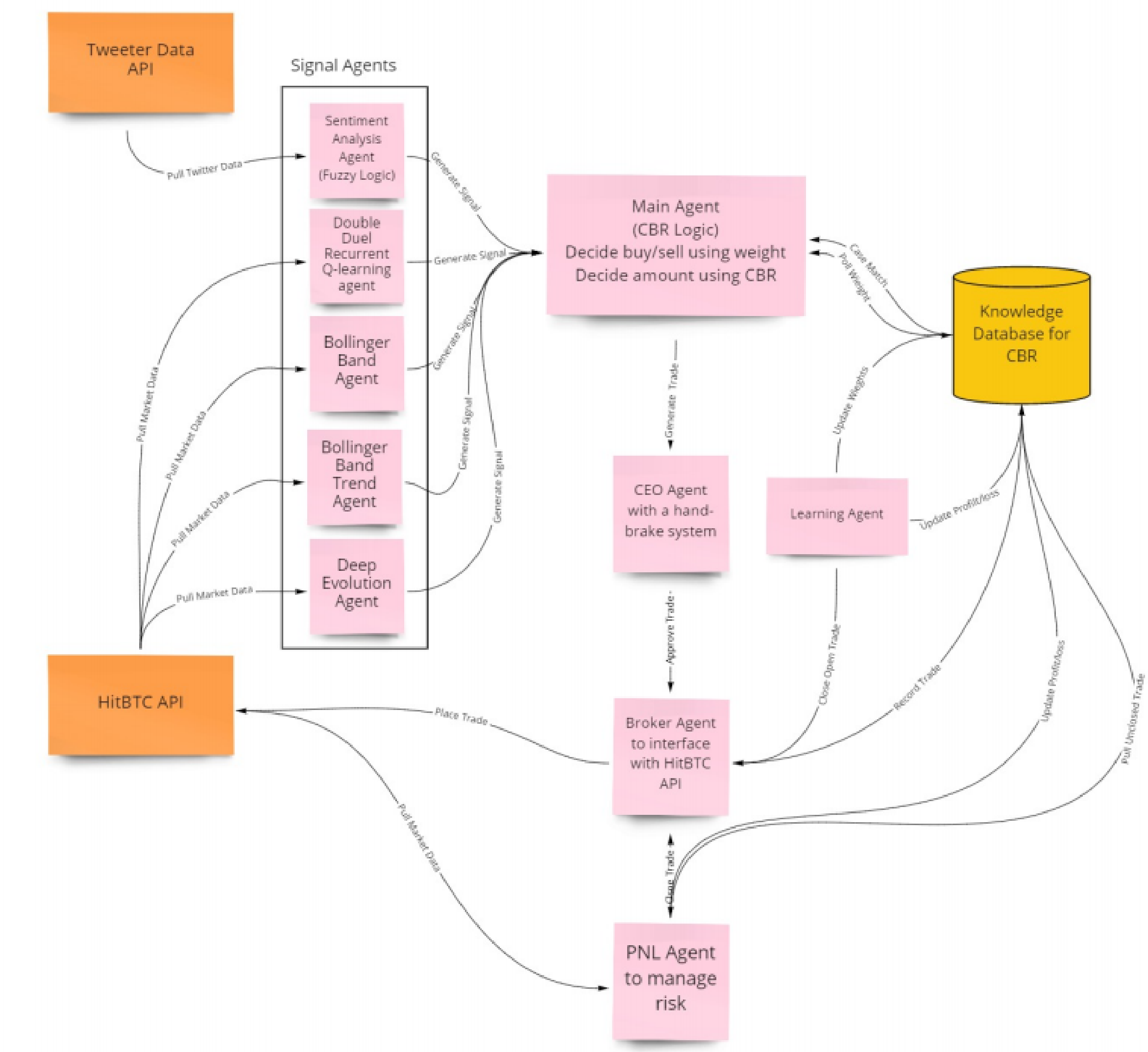

Figure 3.1 MAS Flowchart for IS5006 Group 7 Project

The MAS is incorporated into our high level architecture as depicted by the flowchart above. Each agent serves a different job category which complements the whole system:

- Signal Agents - Captures Market information
- Decider Agent - Makes decision based on signals
- Learning Agent - Update signal agents performance
- CEO Agent - Review decisions of Decider Agent and stop loss mechanism
- Profit/Loss Agent - Risk manager
- Broker Agent - Handles all interaction between system and digital exchange

Case based reasoning is fundamental to the performance of the system as it interacts with the decider agent, learning agent, broker agent and profit/loss agent to:

- Build database of cases (accumulate experiences)
- Advice on future decision making
- Prevent making same mistakes

4. Pros and cons of the solution

4.1. Advantages of the solution

In the context of algo trading, there are many advantages of using MAS and CBR as below.

4.1.1. Avoid Human Emotions

First, the use of softwares avoids human emotion in trading thus minimizing the uncertainty from the human trader's trading habits as well as human risks like fat-finger risk .

4.1.2. Better speed, accuracy and price, reduce cost

The second advantage is the improvement in speed and accuracy as trades are automatically placed based on real-time market data. This allows traders to trade at best price and also minimizes the swing of profits due to market changes while the trade is being finalized, thus reducing transactional costs.

4.1.3. Ability to backtest

Algo trading allows running with past data thus enabling traders to find flaws in trading strategy before trading in the market live.

4.2. Risks in the solution

Risks are inhabited in every stage of the solution, from data input to algo design to output decisions. Deloitte has mapped out the risks in the framework below (Krishna et al., 2017)

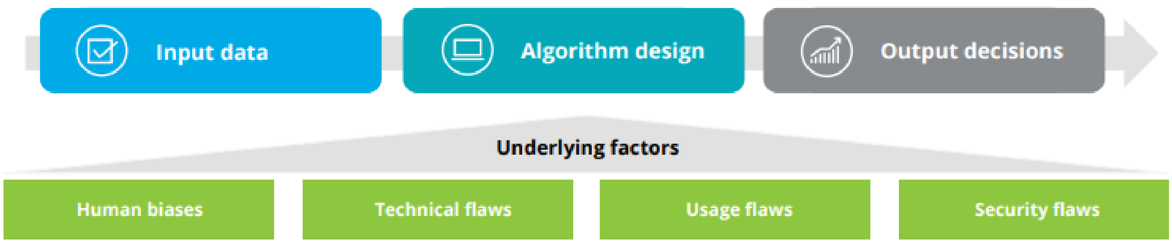

Figure 4.1. Frame for Algo Risk by Deloitte

4.2.1. Bias in data

Biased or incomplete or outdated data undermines the performance of algorithms. Incorrect data source or collection method can also result in the same. In addition to the input data to run the algorithm, data used to train the agents will also impact the performance. For example, if insufficient data is used for training, the agents may capture short-term turbulence in the market and the dominant trends thus learning incorrect patterns before going live.

4.2.2. Flaws in design

Algo trading may work against the traders if design flaws are not fixed. The flaws include incorrect assumptions made in the logic, wrong judgement logic, bugs that cause unintended outputs, misleading user guides, and lack of cybersecurity measures.

4.2.3. Interpretation issue

Output may be interpreted incorrectly if the presentation is ambiguous, or the assumptions used are misunderstood.

4.2.4. Machine Failures

Separate from the algorithm risks above, physical machinery failures can post great risk to the use of MAS and CBR in trading. For example, if a server or main machine fail, and there is no backup, trading can be halted or incorrect/incomplete trading orders may be sent to the exchanges. For example, if a machine fails and an order to sell in one exchange is failed while the order to buy in another exchange is executed, the trader may face losses from the disrupted arbitrage.

4.3. Opinion on the solution

To a large extent, the group's opinion is to use MAS and CBR while conscious effort is required at data collection, algo design and output interpretation stages such as test failover plans before going live, build monitoring mechanisms for performance, place a 'hand-brake' in the system to stop algorithm and seek human intervention if loss or swings hit a threshold.

5. Conclusion

In this paper, we have done a literature review on Multi-Agent System and zoomed into its structure in algorithm trading application which is the topic of the group's project. On top of MAS, we also studied the use of Case-Based Reasoning in financial forecasting, focusing on the state-of-art models and methodologies. We have reviewed our group project and examined the use of MAS and CBR in the project by mapping out the flowchart of agents. Finally, we evaluated the pros and cons of the solutions and reached the conclusion that we can adopt MAS and CBR in algo trading on the condition of proper risk management controls in place. In conclusion, MAS and CBR enabled users to track problems more complex than a single agent system. The AI application in trading allows traders to trade at high frequency with best prices and accuracy but may also amplify flaws in design bias or machine failures. As a result, traders should be vigilant on the risks and prepare for contingency plans and emergency brake mechanisms in advance while harvesting the benefits of algo trading.

6. References

Ahmed Abbas, H. (2015). Organization of multi-agent systems: An overview. *International Journal of Intelligent Information Systems*, *4*(3), 46.

Barbosa, R. P., & Belo, O. (2008, January). Algorithmic Trading Using Intelligent Agents.

Calvaresi, D., Dubovitskaya, A., Calbimonte, J. P., Taveter, K., & Schumacher, M. (2018). Multi-agent systems and blockchain: Results from a systematic literature review. In *Advances in Practical Applications of Agents, Multi-Agent Systems, and Complexity: The PAAMS Collection* (pp. 110–126). Springer International Publishing.

Chun, S.-H., & Kim, S. H. (2004). Data mining for financial prediction and trading: application to single and multiple markets. *Expert Systems with Applications*, *26*, 131-139.

Chun, S.-H., & Park, Y.-J. (2005). Dynamic adaptive ensemble case-based reasoning: application to stock market prediction. *Expert Systems with Applications*, *28*, 435-443.

Chun, S.-H., & Park, Y.-J. (2006). A new hybrid data mining technique using a regression case based reasoning: Application to financial forecasting. *Expert Systems with Applications*, *31*, 329-336.

Krishna, D., Albinson, N., & Chu, Y. (2017). *Managing algorithmic risks*. Deloitte. https://www2.deloitte.com/content/dam/Deloitte/lu/Documents/risk/lu-risk-algorithmic-machine-learning-risk-management.pdf

Leitão, P., & Karnouskos, S. (2015). *Industrial Agents: Emerging Applications of Software Agents in Industry*. Elsevier Science.

Panait, L., & Luke, S. (2005). Cooperative Multi-Agent Learning: The State of the Art. *Autonomous Agents and Multi-Agent Systems*, *11*(3), 387–434.