社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2006－MPS－61 (4)
2006／9／14

Buy-Sell Strategy Model Construction with Hybrid XCS

Shumei Liu† and Tomoharu Nagao††

Many financial time series forecasting techniques have been so far developed for predicting stock prices. However, only with the forecasting value of the next time, it is difficult to determine the optimal buy-sell strategy and get benefit. On the other hand, because the financial time series change severely, it can hardly be identified by a single global model. One model will just be suitable for some kinds of changing patterns, but fail on other patterns. Then, in this paper, we proposed a Hybrid XCS (eXtended Classifier System) learning method by adopting multiple local models. Each local model is called Slaver-Agent and trained with XCS method. A unique Master-Agent chooses which Slaver-Agent is the most effectively for a given changing pattern. With the hybrid learning structure, multiple Slaver-Agents work alternately, and the limitation of learning by one single agent can be overcome. Their learning objective is to obtain profitable transaction decisions directly and get maximum return benefit after several transactions. Experiments have been performed on several well known securities, and the results have been compared with a single agent and some traditional Technical Analysis strategies.

1. Introduction

Recently, a number of different methods have been applied in order to predict stock market. These methods refer to Technical Analysis method, traditional statistics method, and intelligent method. In this paper, we mainly focus on the intelligent method. The general method may include Evolutionary Computation (EC), Machine Learning, as well as Neural Networks (NN).

As effective learning algorithms, Artificial Neural Networks (ANN) [1] or recurrent ANN [2] have been widely used to deal with stock market forecasting. However, several problems occurred. The local over-fitting results it only suitable for a specific period, the limits of readability of NN make it difficult to be analyzed, and the time-consuming of NN is well known to everyone as well.

Aside from the NN, GA and GP are popular algorithms used in the aspect of financial time series forecasting. LCS [3], which are based on a simple GA and RL, have been proposed to this field as well. As an expansion of LCS, in 4)∼6), three different types of agents have been proposed to learn and forecast a real stock market. However, because the vision of a single agent is limited, the aliasing position often occurs. On the other hand, even for those non-aliasing changing patterns, one agent could not successfully forecast them all.

Therefore, we propose a Hybrid XCS model, which transfers the global forecasting problem into several local ones. Each local model represents an independent agent, which is used to learn and forecast. Here, we call then salver agents. A unique Master-Agent chooses which Slaver-Agent is the most effectively for a given changing pattern.

The remainder of this report is organized as follows. Section 2 presents the related works, XCS. Section 3 describes the proposed Hybrid XCS learning architecture. Experiments are presented and analyzed in section 4. Section 5 outlines a conclusion, and discusses the future work.

2. eXtended Classifier System (XCS)

XCS was first proposed in 1995 [7]. In XCS, firstly, the agent's perception is used to build a MatchSet containing the classifiers in the population whose conditions match the perception. Secondly, based on those classifiers in MatchSet, for each possible action, the averaged prediction is computed. Then action, which own the biggest averaged prediction, has been selected the optimal one. Thridly, the selected action is performed and an immediate reward is returned to the system together with a new input configuration. Finally, the classifiers in MatchSet that propose the selected action are

† Department of Information Media and Environment, Graduate School of Environment and Information Sciences, Yokohama National University
†† Department of Social Environment and Information, Faculty of Environment and Information Sciences, Yokohama National University

Table 1 Perception of Master-Agent.

Bit	Perception	Bit	Perception
0	RSI(t,14) $\leq$ 0.4	3	DOA(t-1,13) $\geq$ 0
1	ROC(t,26) $\leq$ 1	4	Dev(t,5,10) $\geq$ 0
2	DOA(t,13) $\leq$ 0	5	RDP(t,10) $\geq$ 0.05

put in the ActionSet. The reward is used to update the parameters of the classifiers in the ActionSet. GA is applied to the ActionSet as well.

The more detail of XCS is described in 7), 8).

3. Hybrid XCS Learning Structure

3.1 Agents Settings

In this proposal, we adopt two agents in this proposal (Agent1 and Agent2), which are called Slaver-Agent here. Each agent concentrates on a local changing area. The Master-Agent(Agent0) is used to cooperate these Slaver-Agents and choose one to forecast the present changing pattern. We now define the structure of each agent.

3.1.1 Master-Agent Settings

Perception of Agent0 is reported in Table 1, each bit is built upon some Technical Analysis method. If the inequality is satisfied, the corresponding bit value is set as '1', otherwise, it is set as '0'. The definition is listed in Eq.(1) and Eq.(2). In the definition of RSI, the *aveUp* and *aveDown* mean the averaged value of all the upward or downward data respectively. With these settings, Agent0 chooses an optimal Slaver-Agent to predict the next buy-sell strategy.

Action of Agent0 labels the number of Slaver-Agent. The action is defined as 0 or 1. Action 0 means that it will select Agent1, and Action 1 will select Agent2.

$$RSI(t,n) = \frac{aveUp}{aveUp + aveDown} \quad (1)$$

$$ROC(t,n) = \frac{p(t)}{p(t-n+1)}$$

$$DOA(t,n1,n2) = SMA(t,n1) - SMA(t,n2)$$

$$Dev(t,n1,n2) = \frac{p(t) - SMA(t,n1)}{SMA(t,n1) - SMA(t,n2)}$$

$$SMA(t,n) = \frac{\sum_{i=t}^{t-n+1} p(i)}{n}$$

$$RDP(t,n) = \begin{cases} 1 & \frac{p(t)-min}{min} > 0.05 \\ 1 & \frac{max-p(t)}{max} > 0.05 \\ 0 & otherwise \end{cases} \quad (2)$$

Table 2 Perception of Agent1.

Bit	Perception
0	$MA_{t-1,20} \leq MA_{t,20}$
1	$MA_{t-2,20} \leq MA_{t-1,20}$
2	$MA_{t-3,20} \leq MA_{t-2,20}$
...	...
22	$MA_{t-23,20} \leq MA_{t-22,20}$
23	$MA_{t-24,20} \leq MA_{t-23,20}$

Table 3 Perception of Agent2.

Bit	Perception
0	$MA_{t-9,10} \leq MA_{t,10}$
1	$MA_{t-19,10} \leq MA_{t-10,10}$
...	...
5	$MA_{t-59,10} \leq MA_{t-50,10}$
6	$MA_{t-4,5} \leq MA_{t,5}$
7	$MA_{t-9,5} \leq MA_{t-5,5}$
...	...
16	$MA_{t-54,24} \leq MA_{t-50,5}$
17	$MA_{t-59,24} \leq MA_{t-55,5}$

3.1.2 Slaver-Agent Settings

The Slaver-Agent perceives a series of simple moving average values firstly. Table 2 and 3 show the perception of Agent1 and Agent2 at trading time t respectively. $MA_{t,m}$ means the average value from time t back to time *t-m*.

Secondly, in Table 2 and 3, if the inequality is satisfied, the corresponding bit value is set as '1', otherwise, it is set as '0'. Then, the agent's perception is set as bit-string successfully. Agent1 perceives 24 bits, and Agent2 perceives 18 bits.

The action of Slaver-Agent is integer value, which indicates the signal of buying, selling or holding stock at the current day. Integer value 1, 2, and 3 represent a buying signal, and the avaliable cash is 25%, 50% and 75% respectively. Integer value 0 and 4 correspond to simply holding the stock. Integer value 5, 6, 7 represent a selling signal, and the stock to be sell is 25%, 50% and 75% respectively.

3.2 Reward

Because the objective of this proposal is to get the return benefit as maximum as possible for trader, the reward value of each transaction strategy is assigned as the benefit proposed by it. However, we could not charge the benefit of this transaction immediately. It will be influenced by the changing pattern of the following stock price. Then, we use the EMA(Exponential Moving Average) value after n transaction days. The n value is set as 10 experimently, and the EMA value is defined as Eq.(3). Here, $y(t)$ is the EMA value at time t, $p(t)$ is price of stock at time t, and

$\alpha = 2/(n+1)$. The reward strategy is defined as Eq.(4), in which (t^+) means the value after transaction has been purposed.

$$y(t) = y(t-1) + \alpha * (p(t) - y(t-1)) \quad (3)$$

$$reward = benefit(t^+) - benefit(t) \quad (4)$$

$$benefit(t) = cash(t) + stock(t) * p(t)$$

$$benefit(t^+) = cash(t^+) + stock(t^+) * y(t+n)$$

3.3 Hybrid Learning Strategy

In this proposal, we construct a hierarchical learning strategy. At first, the Master-Agent learns with a XCS method to obtain which Slaver-Agent is the more effective for a given changing pattern. Then, the selected salver agent is trained by XCS method to learn and forecast the next buy-sell strategy. With the hybrid learning structure, multiple Slaver-Agents works alternately and the limitation of learning by one single agent can be overcome.

4. Experiment

4.1 Experimental Settings

In this experiment, we suppose that at most one transaction is allowed in one trading day, and only one stock is operated. We use the trader's asset to verify this proposal. Raw data of index NIKKEI, and other five stocks have been adopted. This data is taken from the daily closing price from Jul. 2001 to Jul. 2004, downloaded from Yahoo!finance. The maximum population size is 80 for Master-Agent and 1000 for Slaver-Agent. For these two kinds of Agent, the learning data is 200, generation is 40000, and tested data is 400.

4.2 Experiment Result

The asset of trader has been compared between the proposed Hybrid XCS, and other traditional methods in Table 4, where the TAs is the combination of three Technical Analysis strategies before it. The result is average of 5 experiments with the same parameters settings. From these result, we found that the proposed Hybrid XCS gets the most benefit. That means that Hybrid XCS is superior to other methods.

4.3 Analysis

4.3.1 Slaver-Agent's Determination

We now consider why the Hybrid XCS outperforms a single agent, by analyzing how the Master-Agent has classified a given changing pattern into Slaver-Agent1 or Slaver-Agent2. We adopt one experiment result on NIKKEI index for the 400 tested data. For every changing pattern recognized by Master-Agent, the frequency of each Slaver-Agent being selected has been summarized in Figs. 1. The horizontal axis is perception of Master-Agent, and the vertical axis is the frequency of Agent1 and Agent2 being chosen. This is consistent with our expectation on general. Given a changing pattern, the Master-Agent determines which Slaver-Agent is the better one. An appropriate agent will be determined for the different trend. By combining the two Slaver-Agents, the shortcoming of narrow vision for one single agent has been overcome.

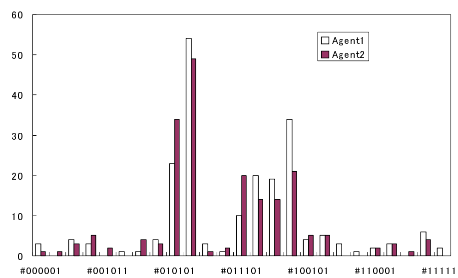

Fig. 1 Slaver-Agent determination with Hybrid XCS on NIKKEI data.

4.3.2 Buy-Sell Strategy Analysis

From the result in Sec. 4.2, we concluded that the proposal outperforms the general Technical Analysis strategy. Next, we verify that the proposal is consistent with traditional Technical Analysis methods, and is superior to it further. Here, we adopt the NIKKEI data once again.

We select 25 data from the 100 ones, and compare it with the RSI strategy as Figs. 2 and Figs. 3. In this figure, the cross-point means "buy" strategy, and the star-point means "sell" strategy. We find that, 'A1' and 'A2' both are good points, which identify that the two methods are consistent and can direct correct act at the same. On the other hand, 'B' and 'C' are good points in proposal, but could not be obtained in RSI. Furthermore, 'D1' is good point in proposal, corresponding to that a bad point 'D2' appears in RSI.

According to the analysis above, the performance of our proposal is better than RSI. For example, the buying at 'B' can get higher benefit, and the selling at 'C' will lead to less loss. Of course, there are also some bad strategies in our proposal. That will be our future interest.

5. Conclusions

These experimental results verify the effec-

Table 4 Comparison of trader's benefit with Hybrid XCS, single agent and Technical Analysis methods (ten thousand Yen).

	HybridXcs	Agent1	Agent2	Buy&Hold	TAs	Deviation	RSI	Golden Cross
NIKKEI	1068.7	996.8	1004.0	809.2	822.3	825.2	837.2	810.1
TOYOTA	350.5	320.6	338.0	300.2	315.9	319.2	348.3	300.9
HONDA	537.2	447.5	489.2	490.3	508.7	512.9	537.2	487.7
NIHONN-Tsuuun	64.4	60.6	60.7	54.2	57.2	57.9	58.4	54.9
SANNYOU	70.5	60.3	67.3	46.8	51.8	49.1	75.4	49.9
NIHONN-Oil	61.0	48.7	59.7	55.0	58.1	58.8	62.7	55.3

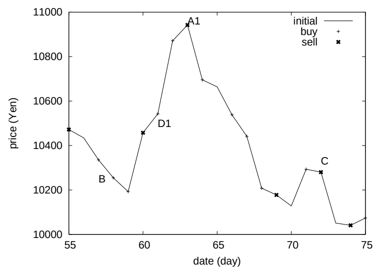

Fig. 2 Buy-Sell Strategy in detail with Hybrid XCS on NIKKEI data.

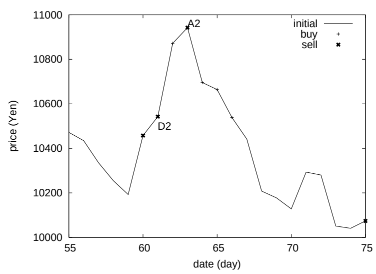

Fig. 3 Buy-Sell Strategy in detail with RSI on NIKKEI data.

tiveness of Hybrid XCS in forecasting the next buy-sell transaction strategy. It outperforms not only one single agent model, but also the traditional Technical Analysis strategy. Now we discuss some aspects on the advantage and limitations of proposed Hybrid XCS.

Firstly, in this proposal, we use two heterogeneous Slaver-Agents within a Hybrid learning structure, to overcome the shortcoming of narrow vision for one single agent. The improvement is obvious. However, the perception of present two agents is defined experimentally. At present, it is difficult to confirm their effectiveness in theory. How to define agent's perception to involve more useful information is our first concern in future.

Secondly, we just apply Technical Analysis method in Master-Agent. For the Slaver-Agents, how the Technical Analysis method can be involved is our next attention. It will refer to RSI, Golden Cross, Dead Cross, long period and short period strategy, and so on.

Finally, applying Hybrid XCS on portfolio problem is another interest.

References

1) Castiglione, F.: Forecasting price increments using an artificial neural network, *Advances in complex systems*, Vol.4, pp.45–56 (2001).

2) Kuan, C. and Liu, T.: Forecasting Exchange-Rates Using Feedforward and Recurrent Neural Networks, *Journal of Applied Econometrics*, Vol.10, pp.347–364 (1995).

3) Booker, L., Goldberg, D. and Holland, J.: Classifier systems and genetic algorithms, *Artificial intelligence*, Vol.40, pp.235–282 (1989).

4) Schulenburg, S. and Ross, P.: *An adaptive agent based economic model*, chapter Learning Classifier Systems: From Foundations to Applications, pp.265–284, LNAI 1831 (2000).

5) Schulenburg, S. and Ross, P.: Explorations in LCS models of stock trading, *In Pier Luca Lanzi, Wolfgang Stolzmann, and Stewart W. Wilson, editors, Advances in learning classifier systems, Vol.2321 LNAI*, Berlin, Springer-Verlag, pp.150–179 (2002).

6) Schulenburg, S. and Ross, P.: Strength and money: an LCS approach to increasing returns, *In Pier Luca Lanzi, Wolfgang Stolzmann, and Stewart W. Wilson, editors, Advances in learning classifier systems, Vol.1996 LNAI*, Berlin, Springer-Verlag (2001).

7) Wilson, S.: Classifier fitness based on accuracy, *Evolutionary Computation*, Vol. 3, No. 2, pp.149–175 (1995).

8) Butz, M. and Wilson, S.: An algorithmic descriptions of XCS, *Advances in learning classifier systems*, Berlin, Springer-Verlag (2000).