Optimizing Resource Allocation with Intelligent Agents (Extended Abstract)

Lucas O. Souza
University of Brasilia
PO Box 4466
Brasilia, Brazil
lucasosouza@gmail.com

Celia G. Ralha
University of Brasilia
PO Box 4466
Brasilia, Brazil
ghedini@unb.br

Bruno W. P. Hoelz
University of Brasilia
PO Box 4466
Brasilia, Brazil
bhoelz@unb.br

ABSTRACT

Playing the stock market is one of the many frontiers of applied artificial intelligence. The problem of optimizing asset allocation within a portfolio to yield a return above the market is a hard problem. Current approaches include multi-agent systems in which the task of gathering data, predicting trend and choosing assets are divided between specialized cooperative agents that simulate the environment of an investment fund. This paper builds upon existing models to create ORACLIA (**O**ptimizing **R**esource **A**llocation and **C**apital **L**ucrativeness with **I**ntelligent **A**gents), a multi-agent system for portfolio optimization in the Brazilian Stock Market (Bovespa). A multi-agent architecture was defined and implemented with the use of machine learning classification models to generate buy and sell signals based on the success or failure probability of a predefined strategy. The classification models are able to predict success or failure of a strategy with over 80% accuracy for certain assets. These models successfully replaces the traditional approach that uses regression to predict stock price trends. Backtesting results, conducted in a simulated environment with actual stock market data from 2015 and 2016, show ORACLIA achieves net results up to eight times higher than single asset portfolios or general benchmarks such as Ibovespa, the main index for Bovespa.

Keywords

multi-agent systems, stock price prediction, asset allocation, supervised learning

1. INTRODUCTION

Choosing to buy assets based on a predictive upward trend is a discipline that dates back to the early days of capitalism. The introduction of artificial intelligence (AI) methods in investment portfolio management and the rise of machine learning have drawn a lot of attention to the field, which promises big rewards for those who can crack the challenge.

When a bank or a hedge fund decides which stock to buy and which to sell, many players are involved in the decision. There are those who gather the data, those who process it and try to come up with stock price trends, and those who use this information to trade. The described players' rules can be well represented by multi-agent systems that resemble real day-to-day environments.

Considering the above scenario, this paper presents ORACLIA, a multi-agent system for portfolio optimization in the Brazilian Stock Market (Bovespa). It was built upon existing approaches in which the task of gathering data, predicting trend and choosing assets are divided among specialized, cooperative agents that simulate an investment fund. Each agent in ORACLIA reflects the role that would be occupied by an employee in a hedge fund structure. The behavior of each agent is heavily based on AI methods, such as supervised learning and reinforcement learning, to add intelligence to the designed system.

2. RELATED WORK

In [4] we find a multi-agent systems for stock price prediction, in a four-layered architecture. Similar approaches are found in previous works [8, 9, 14]. While [8] is focused on predicting stock prices trend, [9] is concerned with investment decisions that optimize asset allocation. The work of [3] presents a multi-agent system, composed of coaches and advisors, that cooperate to optimize a portfolio based on the analysis of assets risk.

Significant work have also been done in the field of agents operating in a simulated stock market environment [15, 7, 1].

There is a vast literature on the topic of using non-linear regressors to predict stock prices. There seems to be a focus on neural networks, regarded as the state of the art in non-linear regressors in [11, 4, 2]. The designed systems have been applied to a variety of markets, from Romania [11] to Germany [4] and Brazil [6].

3. ORACLIA MODEL

The starting point of defining a multi-agent architecture is setting its goals. The main goal of the system in this case is to maximize the return of the portfolio, and one of the relevant sub-goals is to increase precision in predicting whether or not a predefined strategy will be successful. ORACLIA's architecture is depicted in Figure 1.

The Fund Manager agent receives input from the user of which assets to consider, how much capital to allocate and a time frame, and outputs buy and sell recommendations. In order to obtain the recommendations, the manager requests investment strategies from the Strategy Agents. Each strategy agent has access to distinct Data Provider agents that

Appears in: *Proc. of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2017), S. Das, E. Durfee, K. Larson, M. Winikoff (eds.), May 8–12, 2017, São Paulo, Brazil.*

collect relevant data available online, store and provide it when requested.

Strategy Agents also have access to a back-testing service that is used to test its strategies against historic data. Strategy agents respond to the Fund Manager with their best strategies considering the constraints provided by the user. Finally, the Fund Manager can provide one or more of strategies for the user to choose from.

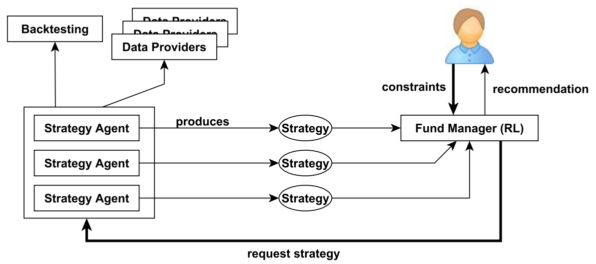

Figure 1: Architecture of the ORACLIA multi-agent system

3.1 Stock Price Prediction as a Classification Problem

The usual approach when dealing with stock price trend prediction is using non-linear regression models to predict future stock prices. Regression is a complex problem, with unsatisfactory results when dealing with stock prices. Predicting the price three days ahead (or three minutes ahead with intraday stock price) is a much harder problem than predicting one day ahead.

To avoid this pitfall, ORACLIA turns predicting stock price trend into a classification problem, contrary to the common approach of using non-linear regressors. The approach is based on a predefined strategy, which the agent holds as fixed. The strategy will tell which type of asset to buy, when, and when to short the position.

The simplest strategy, called swing trade, is buying an asset, and selling it if the price increases or decreases X% within the next N days. Parameters X and N can be optimized.

With a strategy defined, the strategy agent evaluates the entire training set, and for each observation creates a binary label, which indicates whether the strategy would have worked or not if applied on that day. The success/failure label will be the label learned by the classifier.

The goal of the classification problem is, given the variables for the current day, to define whether applying a strategy will be successful or not. There is an inherent trade-off as we are disregarding additional information that would be obtained in the regression-based approach, such as the expected price of the asset, to obtain a simpler model which can be trained to higher precision.

4. IMPLEMENTATION

ORACLIA was implemented using several open source Python packages. Scikit Learn [12] was used for machine learning algorithms and data preprocessing features. Pandas [10] and matplotlib [5], Seaborn [16] and Jupyter notebooks [13] were used for data wrangling and visualization. The agents implementation was done in the *Smart Python Agents Development Environment* (SPADE) [6], which is a FIPA-compliant multi-agent environment based on JADE. The infrastructure used was a Mac 2014 notebook, with 8 cores and 16 GB of RAM. Future implementations will be based on Amazon Elastic Search for better scalability.

4.1 Experiments

A prototype of ORACLIA was tested with the Brazilian stock market (Bovespa). Data from the five stocks with higher weight in the Ibovespa index were used in the experiments: ABEV3, BBDC4, ITUB4, PETR4, and VALE5. The training data considered the period from 2012 to 2014, while the evaluation data considered the period from Jan. 2015 to Oct. 2016.

For each stock, a strategy agent was implemented, using a swing trade strategy and a kNN classifier. In backtesting, the strategy agents achieved precision as high as 80% in predicting whether or not a strategy would be successful. The fund manager agent simulated an investment portfolio in the period of 2015-2016. The return of investment achieved in the period is eight times higher than the IBOVESPA index gains or any of the traded stocks treated separately, as shown in Figure 2.

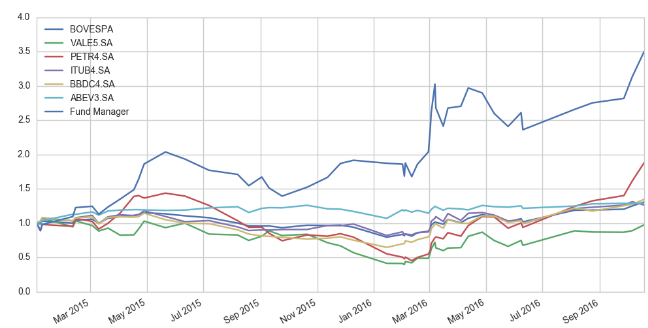

Figure 2: Performance from Jan. 2015 to Oct. 2016, total portfolio valuation of 252% compared to 32% valuation of IBOVESPA index in the same period

5. CONCLUSIONS

This paper presented ORACLIA, a multi-agent system to solve the optimal asset allocation problem, aiming to maximize return of an investment portfolio. To do so, several AI methods are employed, including a novel approach to predict stock price trends using non-parametric classifiers.

The multi-agent approach of ORACLIA, where autonomous agents can operate independently in different locations, facilitates distribution and scalability. The use of the FIPA-ACL communication protocol simplifies the introduction of agents from other FIPA-compliant platforms.

Since the agents in ORACLIA's architecture reflect roles in an investment fund structure, the proposed approach could be deployed as part of the organization. Likewise, human agents could also play the role of a Strategy Agent or even the Fund Manager in the system.

The preliminary results found in the implementation of ORACLIA are promising, and can be further improved by increasing the number of assets per portfolio and the range of strategies applied for each asset.

REFERENCES

[1] D. Bloembergen, D. Hennes, S. Parsons, and K. Tuyls. Survival of the chartist: An evolutionary agent-based analysis of stock market trading. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '15, pages 1699–1700, Richland, SC, 2015. International Foundation for Autonomous Agents and Multiagent Systems.

[2] T. Chenoweth, Z. Obradovic, and S. S. Lee. Embedding technical analysis into neural network based trading systems. *Applied Artificial Intelligence*, 10(6):523–542, 1996.

[3] P. A. L. de Castro and J. S. Sichman. Automated asset management based on partially cooperative agents for a world of risks. *Applied Intelligence*, 38(2):210–225, 2013.

[4] R. Hafezi, J. Shahrabi, and E. Hadavandi. A bat-neural network multi-agent system (bnnmas) for stock price prediction: Case study of dax stock price. *Applied Soft Computing*, 29:196–210, 2015.

[5] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing In Science & Engineering*, 9(3):90–95, 2007.

[6] E. Jabbur, E. Silva, D. Castilho, A. Pereira, and H. Brandão. Design and evaluation of automatic agents for stock market intraday trading. In *Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)-Volume 03*, pages 396–403. IEEE Computer Society, 2014.

[7] M. Kearns and L. Ortiz. The penn-lehman automated trading project. *IEEE Intelligent Systems*, 18(6):22–31, Nov 2003.

[8] J. W. Lee, J. Park, O. Jangmin, J. Lee, and E. Hong. A multiagent approach to q-learning for daily stock trading. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 37(6):864–877, 2007.

[9] Y. Luo, K. Liu, and D. N. Davis. A multi-agent decision support system for stock trading. *IEEE network*, 16(1):20–27, 2002.

[10] W. McKinney. Data structures for statistical computing in python. In S. van der Walt and J. Millman, editors, *Proceedings of the 9th Python in Science Conference*, pages 51 – 56, 2010.

[11] M. D. Nemes and A. Butoi. Data mining on romanian stock market using neural networks for price prediction. *Informatica Economica*, 17(3):125, 2013.

[12] F. Pedregosa et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[13] F. Pérez and B. E. Granger. Ipython: a system for interactive scientific computing. *Computing in Science & Engineering*, 9(3):21–29, 2007.

[14] S. Rahimi, R. Tatikunta, R. Ahmad, and B. Gupta. A multi-agent framework for stock trading. *International Journal of Intelligent Information and Database Systems*, 3(2):203–227, 2009.

[15] A. A. Sherstov and P. Stone. Three automated stock-trading agents: A comparative study. In P. Faratin and J. A. Rodríguez-Aguilar, editors, *Agent-Mediated Electronic Commerce VI. Theories for and Engineering of Distributed Mechanisms and Systems: AAMAS 2004 Workshop, AMEC 2004, New York, NY, USA, July 19, 2004, Revised Selected Papers*, pages 173–187. Springer, Berlin, Heidelberg, 2005.

[16] M. Waskom, O. Botvinnik, P. Hobson, et al. seaborn: v0.5.0 (november 2014).