Neurocomputing 194 (2016) 74–86

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Collective game behavior learning with probabilistic graphical models

Zengchang Qin [a,*], Farhan Khawar [b], Tao Wan [c,*]

[a] Intelligent Computing and Machine Learning Lab, School of Automation Science and Electrical Engineering, Beihang University, Beijing, China
[b] Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong, China
[c] Department of Biological Science and Medical Engineering, Beihang University, China

ARTICLE INFO

Article history:
Received 19 August 2015
Received in revised form
25 November 2015
Accepted 25 January 2016
Communicated by Jiahu Qin
Available online 27 February 2016

Keywords:
Minority game
Probabilistic graphical model
Stock index prediction
Collective intelligence
Behavior decomposition

ABSTRACT

The *minority game* is a simple game theory model for describing the collective behavior of agents in an idealized situation where they compete for some finite resources. In this paper, we assume that collective behavior is generated by the aggregation of independent actions of agents and the action follows the minority game. A probabilistic machine learning model is proposed to model the generative process of how collective behavior emerges from individual actions. By training on collective data, we can infer the most likely parameters use the trained system to make predictions. This model can be regarded as a new learning paradigm of analyzing collective data by decomposing the generative process into independent micro-level games. To demonstrate the effectiveness of the model, we conduct experiments on an artificial data set and the real-world data. A set of selected stock indices are tested to capture their rises and falls in the market.

1. Introduction

Collective intelligence is a shared or group intelligence that emerges from the interactions (both cooperative and competitive) of individual agents and appears in consensus decision making. The collective behavior of systems of many interacting degrees of freedom has been studied intensively in physics [23,22] and different tools are applied in the study of emergence of the collective behavior from interacting agents in economics and sociology [3]. Multi-agent models are one of the most used approaches where each individual agent is self-interested and follows its own rules. Prevalence of cooperation in groups of selfish agents seems to be contradictory to the basic premise of natural selection. Evolutionary game theory provides theoretical framework to address this problem. For example, Szolnoki et al. [29] find that *wisdom of groups*[1] could promote cooperation in evolutionary social dilemmas. Nowak [21] summarizes five rules of the evolution of cooperation and even puts *natural cooperation* as the third fundamental principle of evolution beside mutation and natural selection. Most related research in evolutionary games is focused on the evolution of the system dynamics [23]. For example, the prisoner's dilemma (PD) game in particular is considered a classical paradigm of cooperation vs defection [20]. Many variants have been proposed by considering the PD game in different spatial structures including ordered networks [30], scale-free networks [16] and small-world networks [13]. A comprehensive review is available in [23]. In this research, we turn our focus on the prediction power of the agent-based system by decomposing the collective behavior into micro-level actions of agents by reconstructing the generative process of collective behavior using a probabilistic graphical model [14].

Multi-agent models are widely used to investigate how cooperative behavior emerges in different research areas including psychology [27], micro-economics [10,28], financial market modeling [11,12] and market mechanism designs [24,25]. In agent-based modeling, the involving agents are with similar capability and compete for a limited resource. Yet, the predictive power of such systems is rather limited: the mapping from the collective behavior to agents' individual actions is a one-to-many. Information is lost in the process of aggregation from individual behavior of many to one macro-level behavior. In order to overcome this problem, Qin et al. [26] assume that the agents are homogeneous with some representative behavior controlled by strategies. By tuning the strategy parameters at agent level, we can find that which strategy may yield the most likely macro-level behavior.

If we assume that every agent in the market knows the history data, the key problem is to how to decide to act based on this global information. These agents may share global information and learn from past experience. In this research, we assume that

* Corresponding authors.
E-mail addresses: zcqin@buaa.edu.cn (Z. Qin), fkhawar@ust.hk (F. Khawar), tao.wan.wan@gmail.com (T. Wan).

[1] Wisdom of groups (or wisdom of crowds) is the collective opinion of a group of individuals rather than that of a single expert.

http://dx.doi.org/10.1016/j.neucom.2016.01.075
0925-2312/

agents act based on a simple game that is referred to as the *minority game*. Proposed by Challet and Zhang [4], the minority game (MG) is a simplified version of the El-Farol Bar [1] problem: In the minority game, each odd number of players must choose one of the two choices independently at each turn, the players who end up in the minority side win the game. In the minority game, every agent is assigned with strategy which the agent uses to make choice based on perceived global information (collective behavior) [7]. Heterogeneity is important as it is unrealistic that all agents follow the same deterministic strategy [15].

In previous works [18,26], collective behavior is assumed to be generated by groups of agents with the same strategies. The Genetic Algorithm (GA) is used to optimize the agent behavior parameters in order to get the most likely guess of the collective behavior. Also, different games can be used to model the individual behavior. More accurate a model is for individual behavior, more likely collective behavior we can obtain from their aggregation [8]. In this paper, we propose a novel generative graphical model for modeling the process of generation of the collective behavior from the individual behavior. We use the new model to infer the behavior of individual agents from the available global information. The trained system can be used to predict the future collective behavior. The main contributions of this paper are the following: (1) By studying the relationship between collective behavior and its decomposed agent behavior that can be modeled by the minority game, we propose a new framework for data mining. (2) A graphical model is proposed to model the generative process of the collective data from a group of individual actions. Based on this model, we can infer the individual strategies and predict future collective behavior. (3) We apply this framework to practical time-series data mining tasks including FX rate and stock index prediction. The experimental results demonstrate the effectiveness of the new proposed model.

This paper is organized as follows: Section 2 introduces the minority game and how to use it to model agent behavior. In Section 3, we propose a graphical model for collective behavior learning. Inference and prediction of the model is given in Section 4. Section 5 gives experimental results on artificial data as well as the real market data. Finally, conclusions and future work are given in Section 6.

2. Behavior modeling using the minority game

The minority game (MG) was originated from the El Farol Bar problem and was formulated to analyze decision-making [1]. In each round of the game, there are an odd number of players and each one must independently choose one of the two choices, the players who end up on the minority side are winners. The choice of the minority players is referred to as the winning choice. There is no prior communication among players and we assume that the only global information available is the number of players of two choices from the previous rounds.

Let us first introduce the notations and terminology used in this paper:

Agent: A player of the game is referred to as an *agent* and it is the entity that makes decisions based on its *strategy*. The number of agents that participate in the game is denoted by N, which should be an odd number. An agent is indexed by an integer A: $A \in \{1, 2, \dots, N\}$.

Choice: An action made by an agent: a choice C has two possible values: $C \in \{0, 1\}$. The total number of choices equals to N in each round of the game.

Game: In a minority game, choices are represented by a vector of N elements of binary values $[C_1, C_2, \dots, C_N]$ where $C_i \in \{0, 1\}$ the ith agent's choice in this game. The total number of games is denoted by G.

Minority choice: In every game, the choice of the agents is on the minority side. Formally, let t denote the current game, $t \in \{1, 2, \dots, G\}$, then the minority choice in game t is redefined by:

$$r(t) = \begin{cases} 0 & \text{If } \sum_{n=1}^{N} C_n(t) > \dfrac{N}{2} \\ 1 & \text{Otherwise} \end{cases} \tag{1}$$

Memory and history: In the minority game, we assume that the agent's actions are governed by its strategy and minority choices of previous rounds of the game. An agent with m-bit memory means that the agent will take the information (minority choices) of the previous m rounds into account for making its current decision. The minority choices of the last m rounds at time t can be simplified to the *history*:

$$H(t) = \gamma([r(t-m), \dots, r(t-2), r(t-1)]) \tag{2}$$

where $\gamma(\cdot)$ converts a binary vector into a decimal number[2] for representational convenience. $H \in \{1, 2, 3, \dots, K\}$ and $K = 2^m$, where K is the maximum number of all possible histories.

Strategy: An agent's *strategy* can be represented by a set of rules, a lookup table or a function, which takes the histories as input and output an action of the agent (i.e., make a choice) [4–6]. A strategy, denoted by S, can be regarded as a particular set of choices on all the permutations on previous history of minority choices. Therefore, there are 2^{2^m} possible strategies in the strategy space and we assume that at one time each agent has exactly one strategy. For example, Table 1 shows the 3-bit memory, history and a sample strategy. Considering an agent employing the strategy S, when given memory is $(101)_2$ (with corresponding history 6), the agent will definitely choose the side of 0.

Probabilistic strategy: As we can see from the above example, the strategy can be represented by a lookup table and deterministic. It is not a true assumption when considering the real-world choice making. In this research, we first propose the *probabilistic strategy* (PS), which maps the history to a probability distribution over the two choices instead of one choice only. Therefore, a PS is a set of distributions (one distribution for each given history) from which the agent's choice can be sampled. We can use a Bernoulli distribution $P_B(\cdot)$ for choosing from two choices. The bottom row of Table 1 shows a sample of probabilistic strategy, where $P_B(\alpha)$ gives the probability of choosing 0, and of course, the probability of choosing 1 is $1 - P_B(\alpha)$.

The advantage of using probabilistic strategy is that we are able to incorporate uncertainty in the learning process by placing a prior over choices. Based on the observations at each round of the game, we can update the posterior in order to estimate the probabilistic strategy. In the case of the minority game where we have only two possible choices, therefore, we can use the Beta distribution[3] as the prior.

3. Graphical model for collective behavior learning

Previous works [8,15,18] show that collective behavior can be decomposed into the aggregation of individual agents' actions and each agent has its own deterministic strategy. However, in the real-world, there is always uncertainty in decision making. It is

[2] For the mathematical convenience, $\gamma(\cdot)$ actually gives the output value by adding 1 to the corresponding decimal number of the binary memory.

[3] Beta distribution can be regarded as a special case of Dirichlet distribution. If the game is with more than $N (N \geq 3)$ choices, we then need to use N-dim Dirichlet distribution as the prior.