MULTIAGENT LEARNING WITHIN A COLLABORATIVE ENVIRONMENT

CRISTINA OFELIA STANCIU[1], ADRIAN COJOCARIU[1], LJUBICA KAZI[2]

[1]"TIBISCUS" UNIVERSITY OF TIMIŞOARA, FACULTY OF ECONOMIC SCIENCE, DALIEI STR, 1/A, TIMIŞOARA, 300558, ROMANIA
[2]UNIVERSITY OF NOVI SAD, TECHNICAL FACULTY "MIHAJLO PUPIN" ZRENJANIN, ĐURE ĐAKOVIŠA BN, 23000 ZRENJANIN SERBIA
ofelia.stanciu@gmail.com, a_cojocariu@yahoo.com, ljubicakazi@ptt.rs

Abstract:
Multiagent Learning is at the intersection of multiagent systems and Machine Learning, two subdomains of artificial intelligence. Traditional Machine Learning technologies usually imply a single agent that is trying to maximize some utility functions without having any knowledge about other agents within its environment. The multiagent systems domain refers to the domains where several agents are involved and mechanisms for the independent agents' behaviors interaction have to be considered. Due to multiagent systems' complexity, there have to be found solutions for using Machine Learning technologies to manage this complexity.

Key words: Machine Learning, Multiagent Learning, Multiagent Systems

JEL classification: D80, M15

The great use of computers in any domains is well known and it is impossible not to question whether the capacity of computers to learn wouldn't increase their use. The answer is that as the learning capacity is developing, the impact it determines is spectacular. The ways of making computers to learn and possible use in different domains has been a great challenge, and there have already been developed algorithms in this matter. There have developed applications for speaking and writing recognition, for medical diagnosis that are based on such algorithms. It is considered that Machine Learning will have a major importance in the information and computational technology. As an interdisciplinary field, Machine Learning is related to artificial intelligence, statistics, philosophy, psychology, neuron-biology etc.

Some machine learning based techniques tend to eliminate the need of human intuition for analyzing the data, while other adopt a collaborative approach between man and machine. Human intuition cannot be completely removed since the system designer has to specify the mode of representing data and the mechanism used to search the data. Machine Learning is similar to an attempt of automatize the parts of a scientific procedure.

Machine Learning refers to modifications within the systems executing various tasks related to the artificial intelligence domain, tasks involving recognition, diagnostic, planning, robot control, prevision, which can only be completely defined through examples, by supplying the Input data and the expected results. It is desired for the results to be able to be deducted given that an input data series exists, but there is no well-defined input-output function, only by approximating the implicitly relations. It happens many times that the correlations and links are "hidden" into the huge amount of data, but with the help of Machine Learning technologies, these can be extracted.

There are often designed systems that do not work efficiently in the area they are used, because some particularities of the work manner could not have been well specified when they were built, so the Machine Learning methods come in handy in

these cases. Information can be various and create new knowledge flows, which could cause the reimplementation of the artificial intelligence systems, but given that this is not a practical solution, it appears that the Machine Learning technologies could handle these situations well.

Multiagent Learning is at the intersection of multiagent systems and Machine Learning, two subdomains of artificial intelligence. Multiagent Learning is considered to be learning that is done by several agents and which is only possible due to the presence of several agents.
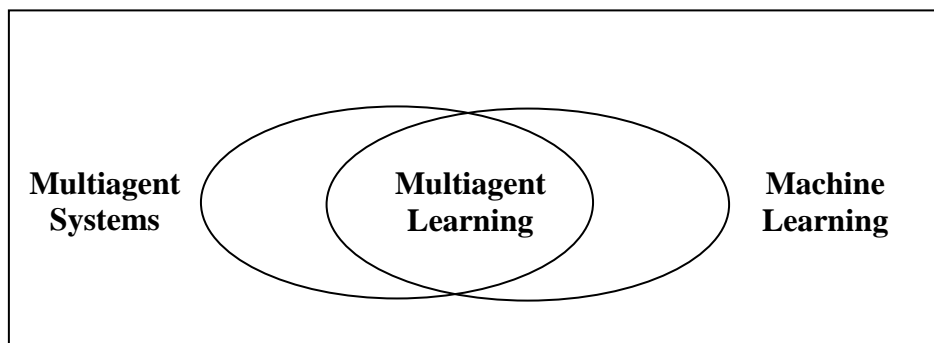
A.I.

Figure 1. *Multiagent Learning at the intersection of multiagent systems and Machine Learning*

Traditional Machine Learning technologies usually imply a single agent that is trying to maximize some utility functions without having any knowledge about other agents within its environment. The multiagent systems domain refers to the domains where several agents are involved and mechanisms for the independent agents' behaviours interaction have to be considered. The multiagent systems include any situation where an agent is learning to interact with other agents, even if the other agents have a static behavior. The main explanation for considering Multiagent Learning the situations when a single agent is actually learning is that its very learned behavior is being often used as a basis for a more complex and interactive behavior. Although a single agent is being involved in the learning process, its behavior is only obvious when other agents are present, and determines these other agents to participate to the collaborative and adversial learning situations. In the case when Multiagent Learning is happening by layering behaviors, all learning levels which imply interaction with other agents are actually contributing to and included in Multiagent Learning.

So far, most of the learning algorithms have been developed for a single agent. The learning of a single agent is focused on the way a single agent improves its own characteristics. Multiagent Learning is out of question as long as the single agent learning does not affect or is not affected by a neighbor agent. Although an agent is not very aware of the other agents' existence, it will consider them as part of its environment and their behavior will be integrated in the learned hypothesis. Coordinating a group behavior seems to be possible by the learning of a single agent.

On the other side, the learning of a single agent will not always lead to best results within the Multiagent Learning to increase the efficiency. There is a difference of consciousness of other agents, and coordination, the question arising if higher level learning will implicitly lead to higher performances.

Distributed Artificial Intelligence (DAI) is a subdomain of artificial intelligence (AI), concerning systems that consist in several independent entities which interact in a certain domain. At the beginning, DAI was divided into two subdisciplines: distributed problem solving (DPS), focused on the management of information within systems that

contain several subsystems working together for a certain goal and multiagent systems (MAS), referring to managing the behavior of a collection of independent entities or agents.

Multiagent systems (MAS) represent a subdomain of artificial intelligence which offers the principles for building complex systems that imply several agents and mechanisms for coordinating independent agents' behavior. As there is no generally accepted definition for the agent, it is considered that an agent is an entity, a sort of robot, with goals, actions and knowledge in a certain domain, located within a certain environment.

Due to multiagent systems' complexity, there have to be found solutions for using Machine Learning technologies to manage this complexity. Before studying multiagent the alternative should be taken into consideration, the alternative being centralized systems with a single agent. The centralized systems have a single agent that is making all the decisions, while others are only helping. Single agents systems must be accepted as centralized systems from a domain that also allows the multiagent approach. A single agent system can have several input data and several actors. But if each entity is transmitting perceptions towards a single core process and receives actions from it, then there is only one agent represented by the core process. Within a single agent process, it is modeling itself, modeling the environment and its interactions with it.

The multiagent systems are different from the single agent systems, and the difference consists in the existence of several agents which are modeling the goals and actions between themselves. Between the agents t here is usually a direct interaction, but although this interaction is seen as stimuli of the environment, the communication between agents is distinct from the environment. In case of these systems, the dynamic of the environment can be defined by other agents.

The building of self-managing distributed computer based systems need a decentralized bottom down approach. The autonomous decentralized systems can be modeled as collections of autonomous agents using decentralized coordination in order to generate a consensus form to be pointed out of a group of partial point of view of the agents upon the system. The consensus between the partial points of view upon the system can be used as a basis for system optimization, coordinating the execution of self-managing actions and the collective adapting of agents in an insecure and changing environment. The benefits of such an approach are an improved scalability, the possibility to set self-managed properties, self-optimization, lack of centralized points of weakness or opened to an external attack, and a possible development of the system through developing agents coordination models.

The development of self-managing distributed systems using decentralized coordination models leads to a lot of issues. Among these it is to mention the proper representation for a local point of view of an agent, the prediction of self-managed actions for agents that allow adaptation to changes within their local environment and producing of answer models that are updating the vision of agents upon the surrounding environment.

Collaborative Reinforcement Learning is a decentralized approach to the establishment and maintenance of properties from the whole system within distributed systems. Collaborative Reinforcement Learning is an extension of Reinforcement Learning for multi-agent decentralized systems, which does not use the knowledge from the whole system and each agent only interacts with its neighbor agents. Collaborative Reinforcement Learning can be used in implementation of coordinative decentralized models on the basis of cooperation and information sharing between agents using coordinative actions and different response models. The response models do not involve a negative response model which alters the local vision of an agent upon its

neighborhood and upon the model of collaborative response which allows the agents to change the effectiveness of the learned actions between them.

In a homogenous system of agents with common system optimization goals, the collaborative component allows agents to share optimal interests, increasing the probability that agents from their very neighborhood to perform similar or related actions. This process can produce a positive answer within the probability of selecting actions for a group of agents. A positive answer is a mechanism that consolidates changes in the system's structure or behavior in the same direction with the initial one. In Collaborative Reinforcement Learning the process of a positive answer continues until a negative answer appears, that is produced either by the system's limitations or by the altering model, causing an adjustment to the agent's behavior. As an effect, agents can establish relations with their neighbors in order to take optimal self-managing actions in case if a particular state of the system. Considering a certain relationship level between agents and their interests, optimization aims to lead the agent's interests to values that are producing collective behaviors.

It is thought that the system's behavioral adaptability to changes within its environment is an evaluation criterion which is important for complex distributed systems as the traditional criteria for static converging environments and stabilization for the optimal behavior of the system.

In the future many distributed systems might define autonomous component that interact and are self-organizing, self-adjusting and self-optimizing without human intervention. In spite of all this, as the dimensions and complexity of the systems are increasing, the possibility of building self-managing distributed systems by using the existing programming languages or top-down modeling techniques is reaching its limits, as there is too much global knowledge required.

BIBLIOGRAPHY

[1] Bishop, C., *Pattern Recognition and Machine Learning*, Springer Publishing, 2006
[2] Filip F.Gh., *Decizie asistată de calculator: decizii, decidenţi – metode de bază şi instrumente informatice asociate*, Editura Tehnică, Bucureşti, 2005
[3] Ganguly, A. R., Gupta A., *Data Mining Technologies and decision Support Systems for Business and Scientific Applications*, Encyclopedia of Data Warehousing and Mining, Blackwell Publishing, 2005
[4] Harmon, M., Harmon, S., *Reinforcement Learning: A Tutorial*, 1997
[5] Mitchell, T., *Machine Learning*, McGraw – Hill Publishing, 1997
[6] Quinlan, J. R. - *C4.5, Programs for Machine Learning*, Morgan Kaufmann, San Mateo, California. 1993
[7] Pop, H. F., *Sisteme inteligente în probleme de clasificare*, Mediamira Publishing Co., Cluj-Napoca, 2004