Organizing Goal-oriented Knowledge Exchange in Multiagent Systems

T. Warden* and O. Herzog

Center for Computing and Communication Technologies (TZI), University of Bremen, Germany
*E-Mail: warden@tzi.de

Summary: Software agents in complex, dynamic environments need to update, adapt, and improve their knowledge models for decision making in order to achieve adequate results. Their individual adaption often relies on machine learning from observational data. However, when data is not available in the required quantity and quality, alternative approaches are required. We propose an interaction-based approach to individual model adaption in multiagent systems, describe agent roles and discuss how a goal-oriented transfer of knowledge among agents can be integrated into an agent-based knowledge management framework.

Keywords: Artificial Intelligence, Knowledge Management, Learning, Distributed Design, Agent

1. Introduction

In recent years, multi-agent technology has been adopted for applications in the logistics domain by a growing number of research groups. Applications range from b2b coordination in the context of supply chain management (SCM) and operative transport planning to management of manufacturing and assembly processes. Typically, software agents represent either logistic service providers or individual human decision-makers such as dispatchers. In autonomous logistics, the control of logistic processes is delegated to individual logistic objects [1]. Autonomous logistic entities, represented by software agents, are to plan and supervise their own passage through logistic networks. Previous research has focussed on collaboration and coordination of individual agents. This research, by contrast, addresses collaboration in agent-oriented knowledge management (KM); namely individual adaption of prediction models that enable informed decision-making in process control. As constituents of multi-agent systems (MAS) for autonomous control of logistic processes, the individual software agents can be characterized by their respective logistic roles [2]. These may involve management of logistic resources. In transport logistics, such resources encompass means of transport or handling equipment, as well as storage facilities [3]. In production logistics, managed resources include machines or material handling equipment in assembly lines. Other roles involve the management of the subjects of the logistic functions themselves; namely, commodities [4] and work pieces [5]. Within the scope of their roles, the agents assume responsibility to execute logistic tasks on behalf of their logistic entities. The logistic roles of the agents constitute primary roles. Thus, informed decision making within these roles is their primary goal.

2. Problem Definition

A prerequisite for informed decision making is the availability of adequate information and knowledge. To acquire both, the agents assume auxiliary roles, which implement agent-oriented KM in addition to their primary logistic roles. Knowledge management is qualified here as agent-oriented for contrast with traditional use cases, since both providers and consumers of KM functions are software agents. Besides shared a-priori domain knowledge, a vital form of knowledge to be provided via dedicated KM roles, especially in dynamic environments, is a-posteriori knowledge. It comprises empirical data acquired via observation and decision support models such as classifiers that can be used as prediction models. Examples in freight haulage and SCM include prediction models for traffic densities within transport networks [3] or handling time at transshipment points. In fabrication and assembly, relevant predictors allow, e.g., estimation of processing times at machines.

The required prediction models ensue from the primary logistic roles. The requisites on dedicated auxiliary KM roles are hence described as follows: Once an agent is to assume a new primary role, models with acceptable prediction performance need to be made available in a timely fashion. We assume that these prediction models need to be constructed by each individual agent based on its own empirical data using machine learning (ML). The agents perform supervised single-agent learning. Their situation is related to that found in Closed Loop Machine Learning in that the agents gather empirical data in a goal-directed way to support further learning. However, since the agents have assigned logistic roles, they need to learn 'on the job'. Thus, the empirical data, which is required to learn a prediction model to support decision making in the context of a logistic role, is acquired from observation of the situation context while already playing the aforementioned role. This situation bears implications for the agent learning tasks: The empirical data that can be acquired is biased by primary logistic roles, specialization, and environment (e.g., due to a particular assignment of haulage or production orders). Second, since agents already need to perform their logistic roles competitively while, in the background, learning models to support decision making, prompt availability of those models is critical. Specifically, it may be infeasible for the agents to first acquire month’s worth of empirical data. When in the context of an auxiliary KM role, an agent fails to learn a prediction model with acceptable prediction quality – implying that its training set has not been representative – its only option for self-sufficient action is the acquisition of further empirical data. This approach may take considerable time. Furthermore, due to the constraint that the agent acts in order to best serve its logistic rather than its knowledge management objectives, it is hard if not infeasible to actively explore the environment to effectuate making 'helpful' observations. Consequently, there is a need to establish novel options for the individual learner to enhance the basis for its learning processes. To that end, consider that depending on the granularity of autonomous control, the primary logistic roles in particular processes may not be performed exclusively by single agents, but by groups of agents. The decentralized operative transport planning for individual trucks in a freight forwarder fleet investigated in [3] constitutes a paradigmatic example: In this scenario, several agents of a logistic service provider assume transport management roles. Thereby, each agent acts on behalf of a single truck. The decision making for such primary roles relies on a common set of dedicated prediction models. Even when distinct role specializations exist, a close relation within a role taxonomy may still imply the employment of similar prediction models.

Autonomous logistics therefore constitutes one particular instance of a system environment in a specific application domain where groups of agents within an MAS each face kindred single-agent learning tasks to construct situated prediction models in order to support identical or related primary roles that are played in overlapping operative contexts. The problem definition is then to unlock potentials for the enhancement of individual learning in system environments exhibiting aforementioned characteristics.

3. Related Work in Machine Learning

Two strands of ML with special relevance are *transfer learning* from previous learning tasks and *interactive learning* supported by other individuals. Transfer learning addresses the specific problem of an insufficient data basis to learn a model through re-use of training data originally collected for related learning tasks [6]. A broadening of the data basis is also a primary goal in active learning [7]. The approach has been suggested amongst others for classification problems where the labelling of training instances is costly or time-consuming. Beginning with a small pool of labelled training data, the learning system iteratively learns a model and assesses which additional data would provide the best chance to optimize the hitherto learned model. This data is acquired autonomously, e.g., through conduct of experiments, or from a human domain expert. Možina et al. have proposed *Argument Based Machine Learning* (ABML). This approach allows to attain improvements in the performance of a learning system by a human domain expert as an interaction counterpart [8]. The learning system is endowed with self-assessment capabilities in that it monitors its own learning progress and, specifically, identifies problem instances in the training data, that are particularly ill-covered by its learned concept description. The learner then reaches out to its pre-set human interaction partner, presenting these instances as queries. The expert uses his domain knowledge to provide a machine-readable explanation (called argumentation). These are accounted for in subsequent learning phases. ABML addresses a challenging flavour of interactive model adaption. It taps on the implicit domain expertise of an advisor to augment the existing advisee training data, thus enabling learning progress. However, the approach needs to be elevated to multiagent learning where knowledgeable peers then subsume the single human expert involved in ABML.

4. A Multiagent Approach for Interactive Adaption

Robust interactive adaption and transfer of individually learned knowledge among cooperating agents in MAS requires a suitable context. We draw on a framework for distributed KM originally proposed for intelligent agents jointly realizing control of autonomous logistic processes [2]. We extend this framework to derive necessary KM roles, means for inter-operability, intra-, and inter-agent organization of multiagent adaption.

4.1. Role-based Distributed Knowledge Management

Often, default knowledge alone is not sufficient in order to accommodate for the complexity and dynamics of an agent's task environment. It then becomes necessary to design adaptive agents, capable of individual knowledge revision and the compilation of tailored models via learning. Over time, knowledge hence becomes to an increasing degree tailored to its task context. Thus, analogously to the situation with employees within organizations, knowledge is spread rather than accumulated in a centralized knowledge repository as assumed in conventional knowledge management approaches. This situation is specifically accounted for in the knowledge management framework through encapsulation of well-differentiated KM functions as agent roles. The strength and flexibility of the role abstraction is that KM-related abilities are not restricted to specialized agents. Rather, any agent is free to assume a time-variant set of knowledge management roles as deemed appropriate in its situation context. These roles are understood as auxiliary roles, which complement the aforementioned domain-specific primary roles. The KM roles can be further categorized into internal and external roles. We adopt a notion of internal roles where these are characterized by reasoning capabilities and a deliberation pattern. Internal roles can be conducted self-sufficiently. External roles also require interaction, structured by one or more interaction protocols.

4.2. Knowledge Management Roles for Interactive Adaption

In the following we sketch the roles that are involved in interactive individual adaption. For an extended discussion of these roles, the reader is referred to [9].

Model Acquisition: This role is a specialization of the know-ledge processing role. It presupposes access to representative training data and an adequate ML scheme (e.g., a decision tree learner). Contingent on the agents' primary domain role(s), the data used for machine learning may constitute individual experience or originate from a data repository. Once a model has been learned successfully, the role also exposes its inferential capabilities for internal used by the agent.

Advisee: This role is a specialization of the knowledge consumer role. Any agent may assume the advisee role when an assessment of its decision support model has shown deficiencies in the model performance that cannot be handled by internal means alone. The agent then becomes an active learner in that it actively seeks for and eventually approaches peers that assume the learning advisory role introduced below. In the interaction associated with these roles, advisors are presented with learning problems and asked to offer advice based on their models.

Advice Integration: This role is a specialization of the know-ledge processing role. It is understood as a subsidiary task to succeed with the advisee role. One can conceive different feasible interpretations of this role. As a first option, the advice provided as input may be used to directly revise an existing model (e.g., by pruning or expanding branches in a decision tree, or revision of a rule set). A second option is to conceive the advice integration as a specialization of the model acquisition role presented above. In such a case, a new model is learned based on the initial training data and the accumulated advice as additional background knowledge to bias/focus the operation of a learning scheme that is able to handle the additional input.

Advisor: This role complements the advisee role and is played by any agent with access to a decision support model if it wants to provide a knowledge advisory service. The model, which is used as basis for the advisory service may be handcrafted, yet the probably more interesting use case involves individually learned models. To play an advisor role, it is necessary to interpret requests placed by advisees and compile tailored advice to address the communicated learning sub problem. The role abstracts from the particular type of decision support. While the adoption of an advisee role is triggered by a concrete need, the complementary advisor role may be played persistently.

Advisory Broker: This role is devised as a specialization of the knowledge broker role. An agent in this role acts as a specialized yellow pages service within the MAS. It administers meta-information about knowledge advisory services exposed by agents. The meta-information is deposited by the advisors. It specifies amongst others the respectively handled learning task (e.g., classification), any bias towards particular learning objectives (e.g., avoidance of false positive classifications) and a meta-description of the learning domain. An advisory broker also accepts requests by agents seeking advice for a learning problem and matches the request information against its advisory portfolio to point the requesting agents to suitable interaction partners.

4.3. Additional Pieces of Interactive Individual Adaption

To structure the interaction between an advisee and a number of advisees, two tiers of control need to be implemented. First, in [9], we have described suitable interaction patterns and an interaction protocol which determine the policy for advisor selection and the pre-set the course of a learning session from sending lesson contents till reception of learning advice. Since interactive adaption typically is an iterated process of multiple learning episodes, we have modelled and implemented this as a Hill-Climbing search in the model space. Thus, at each point in time, the advisee generates learning options defined by specific lesson contents, then seeks advice from peers, and for those options where it received positive feedback actually evaluates the best model revision, measured in terms of accuracy.

5. Prototype Implementation for Experimentation

Except for the advisory broker, the KM-roles outlined in Sect.4.2 that are required for interactive individual adaption of classification models have been implemented, together with the necessary meta-control and interaction protocols, for experimentation within the PlaSMA simulation environment [10] which itself builds upon JADE. The knowledge acquisition and especially knowledge integration roles build upon the ABML implementation by Možina that is included in the Orange system. As shown in Fig.1, the implementation includes additional means for the flexible setup of simulation experiments, thus far using data sets from the UCI ML repository. This allows focusing on all aspects of agent-oriented KM, and later transferring the approach to real-world logistics applications.

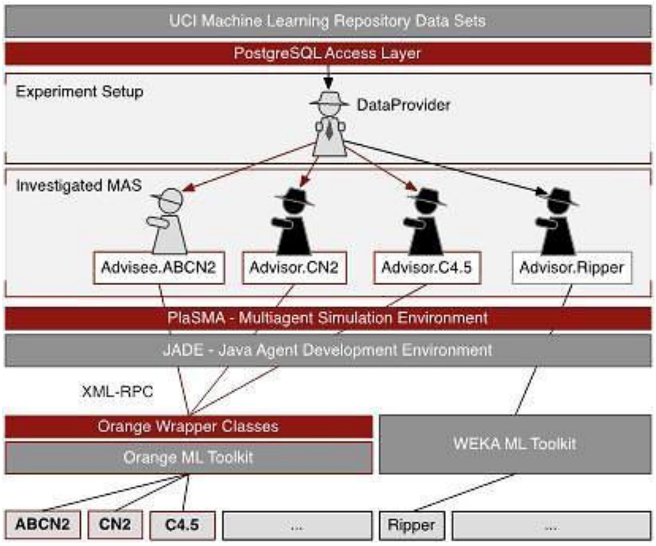

Figure 1. Conceptual overview of the implemented prototype.

6. Discussion and Future Work

In this paper we have sketched an interaction-based approach to the adaption of individual decision support models in MAS. It is desirable when software agents in complex, dynamic environments need to update, adapt, or improve their knowledge base for decision making. Sometimes, this improvement process can be based on machine learning from observational data, alone. But when available data is insufficient in quantity or quality, when data is too expensive, or when the machine learning process turns out to be too complex, alternative approaches are needed. Current work on the proposed multiagent framework comprises an assessment of the optimization potentials and computation/interaction costs, as well as support for multiple active advisors.

Acknowledgements

This research has been supported by the German Research Foundation (DFG) within the Collaborative Research Centre 637 ”Autonomous Cooperating Logistic Processes – A Paradigm Shift and its Limitations” at the University of Bremen, Germany.

References

[1] Böse, F. and Windt, K., 2007, Catalogue of Criteria for Autonomous Control, in Understanding Autonomous Cooperation and Control in Logistics, Springer:57-72.
[2] Langer, H., Gehrke, J. D., Herzog, O., 2007, Distributed Knowledge Management in Dynamic Environments, in Understanding Autonomous Cooperation and Control in Logistics, Springer:215-231.
[3] Wojtusiak, J., Warden, T., Herzog, O., 2011, Agent-based Pick-up and Delivery Planning: The Learnable Evolution Model Approach, in Proceedings of the 5th Int. Conference on Complex, Intelligent, and Software Intensive Systems.
[4] Schuldt, A. ,2011, Multiagent Coordination Enabling Autonomous Logistics, Springer.
[5] Ganji, F., Morales Kluge, E., Scholz-Reiter, B., 2010, Bringing Agents into Application: Intelligent Products in Autonomous Logistics, in AiLog – Workshop at ECAI 2010:37-42.

[6] Pan, S. J. and Yang, Q., 2010, A Survey on Transfer Learning, IEEE Transactions on Knowledge and Data Engineering, 22/10:1345-1359.
[7] Settles, B., 2009, Active Learning Literature Survey.
[8] Možina, M., Zabkar, J., Bratko, I., 2007, Argument Based Machine Learning, Artificial Intelligence, 171/10–15:922-937.
[9] Warden, T., Wagner, T., Langer, H., Herzog, O., 2012, Towards Multiagent Interactive Adaption of Individual Models for Decision Support, in 2nd Intl. Workshop on Semantic Interoperability:553-559.
[10] Warden, T., Porzel, R., Gehrke, J. D., Herzog, O., Langer, H., Malaka, R., 2010, Towards Ontology-based Multiagent Simulations: The PlaSMA Approach, in 24th European Conference on Modelling and Simulation.