23rd International Symposium INFOTEH-JAHORINA, 20-22 March 2024

On Labelling Methods for Development of Multi-Criteria Intelligent Agents in Multi-Purpose Software Product Environment

Nina Bijedic, Elmir Babovic, Drazena Gaspar, Sanja Kapetanovic, Adil Joldic
Faculty of Information Technologies, University Dzemal Bijedic of Mostar
Mostar, Bosnia and Herzegovina
(nbijedic, elmir.babovic, sanja, adil)@edu.fit.ba, drazena.gaspar@sum.ba

Migdat Hodzic
ARTI Analytics,
USA and B&H

Abstract— **Use of intelligent agents in Artificial Intelligence (AI) is steadily increasing in a variety of applications. Software products which rely on these agents are designed with an ever-increasing magnitude of implementation complexity. The agents themselves tend to use a limited number of methods, already developed and proven in AI practice. Nevertheless, the fact that multiple agents can use the same method for different purposes results in a need to label the methods in a meaningful and useful way, sufficient for such a complex set of agent relationships. In that context, this paper presents initial findings and considerations as to what is relevant in methods labelling for enabling proper and effective interactions of multiple multi-criteria intelligent agents.**

Keywords-component; Artificial Intelligence, Method Label, Intelligent Agents, Software Products

I. INTRODUCTION

Intelligent agents (IAs) seem to penetrate all aspects of AI and IT; their use has already proven to be widespread and efficient in, for example, bitcoin manipulation and recommender systems [2][3]. But nowadays it seems that IAs are given new roles, and that researchers and software developers tend to rely on them to answer the ever-more-complex demands of the software applications market. Since the IA's tend to implement already developed and proven trustworthy AI methods, it is now of crucial importance to decide on the appropriateness of the selected methods for specific applications. For the IAs to select appropriate methods, the methods themselves must be thoroughly labelled according to the attributes of each method's specific application, and agents should be able to learn about efficiency of the label attributes for the specific uses [1].

The initial research presented in this introductory paper is a continuation of the work presented in [7], in which the authors considered domain data reduction in big data environment combined with a result of the development of a multi-purpose software product in which it is foreseen that decision making in most of the processes (software product functions) is delegated to the intelligent agents. The IAs' development initiated the need to strictly define method labels, to facilitate efficient data-based inference. Even if the initial reasoning was related to the single criterium agents, it immediately turned out that to improve IAs' performance it is necessary to develop multi-criteria IAs [5]. Furthermore, it turned out that different agents will certainly deploy some of the same methods, so the complexity of the problem rapidly increased. Therefore, in this paper the authors discuss their ideas related to the structure of labels that should efficiently enable intelligent agents to perform the foreseen tasks with high accuracy, as well as reliability.

II. AGENT ENVIRONMENT

By the definition of intelligent agents, they interact with the environment by receiving input and giving feedback [4]. Therefore, it is extremely important to discuss all possible aspects of the meaning of the term "environment" to be able to decide on the relevant label attributes. In this context it is important to notice that the same output in the different environments can result in a totally different basis for decision making. One good example is the result of Factor analysis deployed for testing of a study program Curriculum consistency regarding the learning outcomes [6]. In the research presented in the paper it turned out that the roof course on software development does not match any group of learning outcomes. In a well-ordered society, which invests into the education process, that should be a problem probably resulting in blaming the responsible teacher, but in a poor society as was the case in the described research interpretation can be different. It turned out that absence of classification is a consequence of the effort that seems too much for the students at the stage when the local IT companies literally fight for them knowing the competences that they already have. In such situations a lot of students choose to work rather than to invest more effort into the finalization of their education process.

As described in above mentioned example, it is crucial to establish such an architecture in which IAs can learn from environment and use modelled knowledge to autonomously decide on appropriate method which is optimal for certain set of tasks especially in multi-purpose software product environment.

When it comes to development of a complex software product, in which various agents should be able to utilize methods from a same set, the term "environment" can stand for any of the following:

- A specific user (it might prove that at least some of the users will have specific data for reasons that software developers are not able to foresee, so agents should be able to learn about the best method-user matches)
- Environment of in which the specific user will use the product (for example, regulations and their implementation and/or interpretation directing the specific usage can vary depending on the country/region, in accordance with wealth, other socio-economic factors, etc. in a way that interpretation of output inference-based decision making must be different)
- Purpose for which the user intends to use the product (for example, the user might have data from specific industry branch, or the user wants to use only certain functions of a complex software product, etc.).

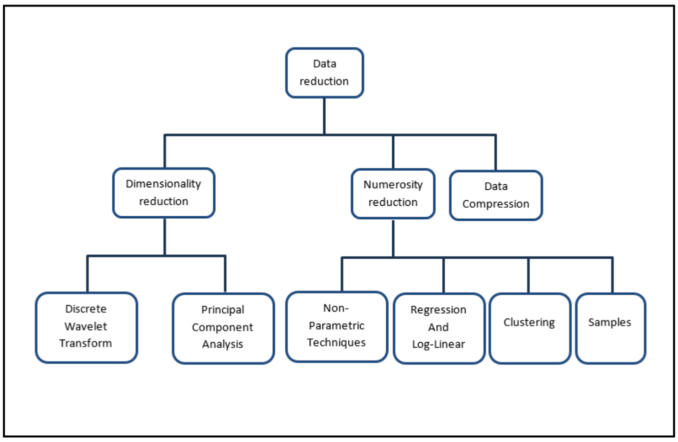
III. DOMAIN DATA REDUCTION

Before elaborating on method labeling approach, we summarize our earlier research related to domain big data reduction framework, see [7]. This step would be performed by another IA, or part of labeling IA, prior or together with the labeling and the two are closely related. In addressing these problems, organizations are faced with various challenges, including:

- Rapid data growth
- Numerous and diverse data sources
- Data heterogeneity and incompleteness
- Diverse and immature technology solutions
- Shortage of domain big data experts
- Data availability, equality, and security
- Legal issues and readiness to change.

Domain data reduction is an extensive and complex research field that includes various statistical methods, data mining and machine learning methods, but also specific steps and procedures. Even for experienced researchers in this field, it is often challenging to navigate all this. That is why researchers have tried to develop frameworks to identify the necessary steps in the domain data reduction process and make the whole process more transparent. However, most of the existing reduction frameworks are either focused on a specific problem or are incorporated into enterprise architecture for domain big data systems. The research question which the authors considered is whether it is possible to develop a general framework for domain data reduction that would serve as a basis for developing a general or specific intelligent platform and the corresponding IA for data reduction. This IA can be combined with the labeling IA, or it could be implemented separately as a part of the IA hierarchy.

Figure 1. General data reduction framework as a part of labeling IA or as a



separate IA

Method selection is dependent on the purpose of the software part as well as the environment in which the software is used.

When it comes to the environment, it can be expressed through a user; for example, if two different users are using the same functionality of a software product, each with their own data, the choice of the method can be different. Therefore, the authors suggest that the user can rate the output from the two different aspects: how well the user understands the output of a certain method and how useful the result is (does the output contributes to the increase of the user's knowledge, and if so, how much).

When considering the purposes for which the methods can be used, in the case of the data reduction process, it is necessary to bear in mind that all the methods already have known advantages and disadvantages. Furthermore, the authors suggest keeping the criterion of method purpose as general as possible, for it is not clear from the beginning is a method can be used for multiple purpose at the time it was being developed.

While methods have parent criteria, various algorithms that are developed for implementation of one method must inherit the parent criteria and add their own.

For the primary criterion in algorithm selection, the authors suggest choosing a mathematical output from a specific algorithm that is related to precision of the obtained result. This measure can be either statistical significance, or any other measure that a specific algorithm has as an output, and to make those measures comparable, they can be expressed in ranks, as a value between 0 and 1. In this case it is also necessary to limit the desired result not only with the lower, but also with an upper bound. This is because the results with near perfect fitting scores do not have to be generally applicable or informative, for example, that neural networks can be overtrained (or over adapted to the certain data set), or that the “perfect” regression is not informative. Further criteria in the algorithm choice are related to the cost, i.e. algorithm performance, or time and space that an algorithm needs to fulfill the task.

IV. METHOD LABELING

The natural order of the environment influencing the agent via input should be visible in the hierarchy of the label attributes’ organization. In that sense, the attributes related to the environment should be at the highest level. Nevertheless, it is not clear if the attributes should be solely qualitative, or if they should have an added weight. The qualitative attributes have a binary property of being accepted or rejected, but attributes with an added ponder would be able to mimic the usually fuzzy structure of the real-world environments, or, to put it into the terms of probability and statistics, to mimic all the factors as well as their interactions that software product developers can not foresee. In cases of pondered attributes, the IA should be able to learn about the threshold which will suggest if the method is appropriate for use in a specific environment or not as illustrated in Figure 2.

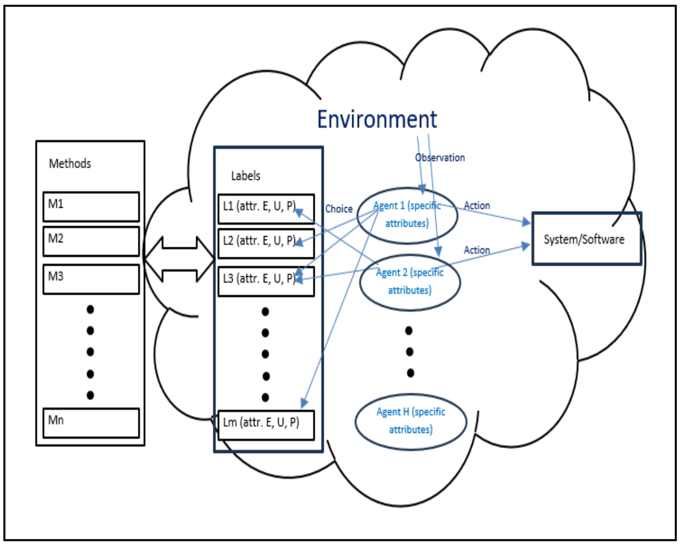


Figure 2. IAs with specific attributes and labels with E-environmental, U-user specificities and P-purpose attributes are base on which appropriate method is chosen

At the lower level of hierarchically ordered label should be the attributes that are internal to the software product in the sense that they are connected to the listed methods that IA’s can use. Those methods can also be labeled as categorical, pondered categorical, or numerical attributes. Among many possible, the main attribute here certainly seems to be some measure of method accuracy, but there are other attributes that can improve agents’ choice of method.

To summarize, criteria are:

- Purpose for which the IA intends to use the method (it is common knowledge that the same statistical method can be used for multiple purposes, and it might prove that algorithms developed and commonly used for certain purposes can prove efficient in serving other purposes than initially foreseen)
- Method accuracy according to the purpose and the attributes of higher hierarchy
- Any other classification relevant to the specific purpose

All the above discussed results in the need for the methods to have a comprehensive list of meta-data organized in the hierarchical levels. Furthermore, it follows that, as the list of potentially useful methods grows, those meta-data should be organized in a small database (or to allocate space in the product database). Consequentially, the use-case diagrams should also show intelligent agents as acters in such a product, and their interactions must be described as well. In a way, that process can affect the software architecture itself, because the agents should be able to create new links and stop using the links that turn out to be irrelevant.

V. CONCLUSIONS

In this research, authors emphasize the importance of implementation of learning and data reduction IAs, in software development projects from the perspective of autonomy of choice of appropriate methods from the set of labelled methods. There are several different ways for methods labelling such as per categorical, pondered categorical, or numerical attributes. In the situation of complex, multipurpose software development, in which the implementation of intelligent agents is unavoidable, it is crucial to prevent rapid increase of complexity by unstructured methods choice by multiple agents. Therefore, the introduction of structured labels for different methods can be seen as a possible solution to decrease complexity and improve reliability. Labels basically reflect environment and software purpose as well as individual methods accuracy and all other attributes which are relevant to implementation of certain method.

REFERENCES

[1] S. Yousaf, H. Mahfooz Ul Haque, M. Atif, M. A. Hashmi, A. Khalid, P.C. Cong Vinh, A context-aware multi-agent reasoning based intelligent assistive formalism, Internet of Things, Volume 23, 2023, 100857, ISSN 2542-6605

[2] S. Sharma and A. Ahlawat, "Architecture and Types of Intelligent Agent and Uses of Various Technologies," 2022 3rd International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), Ghaziabad, India, 2022, pp. 1-5, doi: 10.1109/ICICT55121.2022.10064524

[3] V. H. Subburaj and J. Urban, "Intelligent Agent Software Development Using AUML and the Descartes Specification Language," 2011 14th IEEE International Symposium on Object/Component/Service-Oriented

Real-Time Distributed Computing Workshops, Newport Beach, CA, USA, 2011, pp. 297-305, doi: 10.1109/ISORCW.2011.43

[4] A. Ali and M. F. Farooqui, "Interaction among Multiple Intelligent Agent Systems in web mining," 2022 3rd International Conference for Emerging Technology (INCET), Belgaum, India, 2022, pp. 1-8, doi: 10.1109/INCET54531.2022.9824344

[5] M. Ole and G. Volker, "Towards Concept Based Software Engineering for Intelligent Agents," 2019 IEEE/ACM 7th International Workshop on Realizing Artificial Intelligence Synergies in Software Engineering (RAISE), Montreal, QC, Canada, 2019, pp. 42-48, doi: 10.1109/RAISE.2019.00015

[6] N. Bijedić, D. Gašpar, A. Joldić (2023) On Machine Learning Supported Curriculum Design and Development, In EDULEARN23, 15th International Conference on Education and New Learning Technologies Proceedings, pp. 2739-2743. IATED Digital Library

[7] N. Bijedić, D. Gašpar, M. Hodzic (2022), Proc. of the International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME), 16-18 November 2022, Maldives