German Medical Data Sciences 2023 — Science. Close to People.
R. Röhrig et al. (Eds.)

doi:10.3233/SHTI230709

Comparison and Incorporation of Reasoning and Learning Approaches for Cancer Therapy Research

Andre THEVAPALAN[a,*], Daan APELDOORN[b,1,*], Gabriele KERN-ISBERNER[a], Ralf G. MEYER[c], Mathias NIETZKE[c], and Torsten PANHOLZER[b]
[a] *TU Dortmund University*
[b] *IMBEI Medical Informatics, University Medical Center of the Johannes Gutenberg University Mainz*
[c] *St.-Johannes-Hospital, Dortmund*

Abstract. Representing knowledge in a comprehensible and maintainable way and transparently providing inferences thereof are important issues, especially in the context of applications related to artificial intelligence in medicine. This becomes even more obvious if the knowledge is dynamically growing and changing and when machine learning techniques are being involved. In this paper, we present an approach for representing knowledge about cancer therapies collected over two decades at St.-Johannes-Hospital in Dortmund, Germany. The presented approach makes use of InteKRator, a toolbox that combines knowledge representation and machine learning techniques, including the possibility of explaining inferences. An extended use of InteKRator's reasoning system will be introduced for being able to provide the required inferences. The presented approach is general enough to be transferred to other data, as well as to other domains. The approach will be evaluated, e. g., regarding comprehensibility, accuracy and reasoning efficiency.

Keywords. Answer set programming, cancer therapy recommendation, decision support system, expert system, knowledge representation, machine learning

1. Introduction

1.1. Background

At Dortmund-based St.-Johannes-Hospital, knowledge about experiences with cancer therapies has been collected for over two decades. First approaches of rendering the collected knowledge accessible and maintainable (especially for medical doctors) [12] comprised (1) a semi-formal mind map approach and (2) a formal approach based on *answer set programming* (ASP) [5]. While the first approach appeared to be practical by allowing for collecting knowledge easily through the creation of nodes in the mind map, the steadily growing and changing mind map has become more and more complex and thereby harder to maintain over time. For the second approach, the knowledge contained in the mind map was modeled manually in the form of formal rules, allowing for

[1] Corresponding author, daan.apeldoorn@uni-mainz.de
[*] Equivalently contributing authors (shared first authorship)

automated inference of therapy recommendations. This approach already overcame some issues of the original mind map approach. However, despite its straightforward syntax, the approach of ASP generally requires that knowledge experts (in this case medical professionals) have prior technical knowledge regarding logic programming to fully comprehend a large logic program and in particular its semantics.

In this paper, an approach is presented that can enrich the aforementioned approaches for overcoming the described issues. It is built on the InteKRator toolbox [3], [8] which allows for learning a human-readable representation of the knowledge. An extended use of InteKRator's reasoning system is described to be able to cover the semantics induced by the prior approaches. The resulting system is evaluated, e. g., regarding comprehensibility, accuracy and reasoning efficiency, and is fit for a potential multi-paradigm approach to be used in conjunction with [12].

1.2. Objectives and Requirements

The overall objective of the approach presented here is to learn comprehensible representations of cancer therapy knowledge collected for over two decades at St.-Johannes-Hospital in Dortmund, Germany (for preliminary work see [12]). Such an approach will be useful to enrich and support preliminary approaches.

A steadily growing knowledge base should be able to assist medical experts in their decision-making processes. We therefore argue that such knowledge has to be represented in a human-readable way and it should be possible to transparently infer cancer therapy recommendations from provided patient attributes. Moreover, it should be possible to explain the inferred recommendations and easily maintain the knowledge to be able to reflect the dynamics of cancer research in an adequate way. The resulting approach should be general enough to be applied in other domains as well.

In summary, the main contributions of this paper are:

- Illustrating how medical knowledge can be learned from ASP programs using the InteKRator toolbox and how its reasoning system can be used in an extended way to reflect the inferences (i. e., answer sets) provided by ASP
- Enriching inferred therapy recommendations with quantitative information
- Providing explanations for the inferred recommendations
- An evaluation of the new approach in comparison to and in conjunction with the two preliminary approaches.

2. State of the art

The state of the art is considered in two ways here: First, it will be referred to the current state and preliminary works at St.-Johannes-Hospital (Section 2.1). After that, the relevant current limitations of the InteKRator toolbox will be taken into account and it will be considered how to overcome them for this work (Section 2.2).

2.1. Current State and Preliminary Works at St.-Johannes-Hospital

Intensive cancer research yields a wide range of therapy types and pharmaceuticals for cancer therapies. Based on the patient and the properties of their cancer, different therapies are eligible. Such therapies include medicinal treatments where different combinations of drugs are applied based on temporal schemes.

During the past two decades the pharmaceutical department and the clinic of Hematology and Medical Oncology of the St.-Johannes-Hospital in Dortmund, Germany, have developed a structured collection of therapy plans for cancer patients based on clinical practice guidelines, and decision of an expert panel (tumor board), and individual experts. These therapy plans include patient requirements for their eligibility and therapy adaptations. The plans are also extended by additional information for treating physicians. Due to the rapid progress in cancer research, this collection (in the following referred to as *mind map*) is continuously growing with an average of 20,000 changes per year. The knowledge in this mind map is currently shared between over 125 hospitals and 50 medical practices in the form of a (mind map-like) web application, which assists physicians in safely and efficiently finding suitable therapies for the individual patient.

2.2. InteKRator's Reasoning System

InteKRator [8] is a practical result originating from research in the context of AI in games [1],[4].[6]. In [3], it has been proposed for using it in the context of medical applications. Originally, InteKRator had been designed to learn to infer actions from states of a game, assuming that states and actions are stemming from disjoint sets. Thus, the expressivity of InteKRator is usually limited to only one conclusion (action) for the provided input data. This paper presents a way to overcome this limitation for being able to derive multiple alternative therapy recommendations.

2.3 Related Work

To the best of our knowledge no other hospital compiles cancer therapy recommend-dations in similar, manual fashion which is suitable for the utilization in expert systems.

In the literature, systems that are similar to the proposed approach are often referred to as *clinical decision support systems (CDSSs)*. In [11], the authors discuss the general properties of CDSSs and their risks and benefits. They distinguish between knowledge-based systems and non-knowledge-based systems. In knowledge-based systems, the knowledge is created manually and maintaining such systems can become quite strenuous. In non-knowledge-based systems, the knowledge is built up using AI methods. Here, *Watson for oncology (WFO)* is one of the first implementations [10]. As in the presented approach, WFO is able to aid oncologists in finding suitable therapies for cancer patients. However, in systems like WFO, the knowledge is mainly obtained using purely machine-learning methods which allows the system to analyze clinical guidelines and update the knowledge base regularly. This results in the fact that such knowledge cannot be expressed in a human-readable and comprehensible form which prevents a direct proof that any learned knowledge is in fact medically correct.

3. Concept

The principal idea of the approach presented here is as follows: As a first step, semi-synthetic data sets are derived from the original mind map, where each row of a data set represents a *possible cancer patient*. A *possible cancer patient* is any (meaningful) combination of patient attribute values that occur in the mind map. For each possible cancer patient there exists a (possibly empty) set of therapy plans. A *therapy plan* is a sequence of different *therapies* where each therapy is of a different type (e.g., chemotherapy). One separate data set D_i is derived for each type of therapy, where i denotes the potential position of the therapy type in a therapy plan. Each data set D_i is organized in n input columns and one output column, where every input column represents a patient attribute or a therapy type previously applied to the patient. The output column represents the i-th type of therapy applied to a patient. To obtain every possible cancer patient and the corresponding recommended therapy plans in form of such data sets, we will use the aforementioned ASP encoding of the mind map. As a second step, the InteKRator toolbox [8] is used to learn knowledge bases in the form of rules with exceptions (see [3]) from these data sets. Every learned knowledge base KB_i represents the knowledge about the i-th therapy type of a planned cancer therapy. These knowledge bases are then used to infer the therapies of the respective type from the patient data. Figure **1** visualizes the workflow (for further details, see subsections).

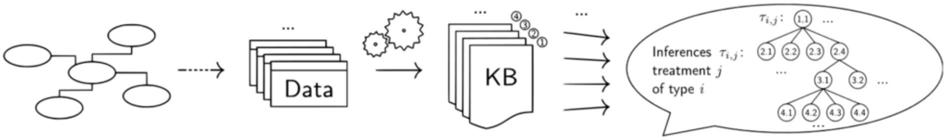

Figure 1. From Mind Map to Comprehensible Knowledge and Inferences: Multiple data sets are derived from a mind map by encoding the mind map as ASP and querying it with all combinations of possible patient input features for the i-th therapy type of a planned cancer therapy (dashed arrow). The data sets are used to learn knowledge bases KB_i representing the knowledge of the i-th therapy type of a therapy plan (gear arrow). Inferred therapies are used as additional input for inferring the therapy for the next therapy type, resulting in an inference tree where the leaves represent the answer sets of the ASP approach from [12].

3.1. Mind Map

Our presented approach extracts knowledge that is encoded in the mind map[2] by the St.-Johannes-Hospital representing medical attributes or therapy names.[3] A path from the root to a leaf represents a full *therapy plan* where the path's nodes specify which therapies are recommended given the attributes of the path. Generally, therapy plans in this mind map can contain systemic therapy types, that is, *chemotherapy, antibody therapy (pre- and post-operation), antihormone therapies* (in the following also de-noted by *CT*, *AbT(-PRE)*, and *AHT*, respectively), and *operation (OP)* as the local therapy type. Based on the patient, one or more medical therapy types can be recommended. However, the general order of the therapy types is set, i. e., a plan can start with chemotherapy, followed by an antibody therapy, an operation, a subsequent antibody therapy which can be accompanied or followed by an antihormone therapy. Typically, for each systemic therapy type a selection of drugs is available. It is on the medical experts to choose the explicit drug composition for each individual patient. For more details, we refer the reader to [12].

[2] The mind map used in this work dates back to 2015. Since then, only the amount of encoded therapy options was extended while the general structure and functioning of the graph structure remained unchanged.

[3] Note that the used graph structure differs from the classical decision tree [9] as nodes are both attributes and their possible values.

Example 1. Suppose a 45-year-old premenopausal, breast cancer patient named P_1 in an otherwise average/good general health who has a HER2-positive, ER/PR-positive, and node-positive tumor that does not need surgical intervention (which in this case means that an adjuvant therapy is desired) and who is additionally allergic to a substance group called anthracyclines. Figure **2** shows the different therapy plans that a medical expert can recommend to P_1.

In this way all possible therapy plans are stored in the mind map.

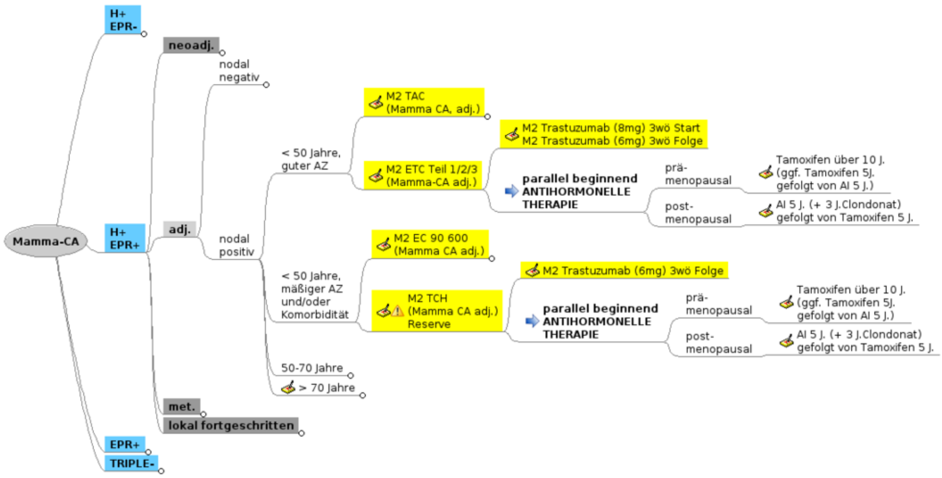

Figure 2. An Exemplary Path of the Mind Map [12]: Nodes are either patient values or therapy recommenddations. (A more detailed discussion is provided in Example 1 and 2.) Note that since this figure originates from the mind map used at the St.-Johannes-Hospital in Dortmund, all labels are in German language (please see explanations in the text).

3.2. Creation of Data Sets

In [12], the authors present an ASP-based application called *Mamma-DSCS* that encodes the mind map's knowledge for breast cancer therapies as a logic program. It allows to determine all possible *therapy plans* for a patient by adding the available patient values as instance data to the program. To show that the answer set program fully represents the therapy options and conditions of the mind map, the authors defined the instance data for each possible composition of patient values and compared the result of Mamma-DSCS with the therapy plan of the mind map.

Example 2. To obtain the therapy plans for patient P_1, we inspect the paths of the mind map that correspond to the patient's values in (Figure **2**). Each path begins with the following nodes: "*Mamma-CA" (mammary carcinoma), "H+EPR+" (HER2-positive/ER/PR-positive),* and "*nodal positiv" (node-positive).* As the patient's general health is specified as average to good, we get a first split, that is, one path includes the node *"<50 Jahre, guter AZ" (younger than 50, in good general health),* the other path includes "*<50 Jahre, mäßiger AZ und/oder Komorbidität" (younger than 50, average general health and/or comorbidities).* Each of these two paths split again, thereby providing us with four different chemotherapy options that are denoted by *TAC*, *ETC*, *EC*, and *TCH*. However, a medical expert in this field is aware that the therapies TAC and EC include anthracyclines, thereby rendering these two therapies inapplicable. This leaves two different paths each of which is followed up by a node that encodes the recommendation of an antibody therapy with *Trastuzumab*. After an ETC therapy, the patient receives a Trastuzumab therapy with a starting dose of 8mg every three weeks ("*Trastuzumab (8mg) 3wö Start"*) and a subsequent Trastuzumab therapy with 6mg every three weeks ("*Trastuzumab (6mg) 3wö Folge*"). If the patient should receive a TCH therapy as the chemotherapy, the starting dose is omitted. In both cases an antihormone therapy parallel to the antibody therapy is recommended via the node "*parallel beginnend ANTIHORMONELLE THERAPIE*". As the patient is known to be premenopausal ("*prä-menopausal"*), in both paths an antihormonal therapy with Tamoxifen for 10 years ("*Tamoxifen über 10J."*) or alternatively a Tamoxifen therapy for 5 years and a subsequent 5-year long therapy with aromatase inhibitors ("*ggf. Tamoxifen 5J. gefolgt von AI 5 J.*") is recommended.

Thus, the program that encodes the mind map's knowledge and the information which therapies contain anthracyclines outputs that the patient either receives a chemotherapy named TCH that is followed up by the antibody therapy Trastuzumab or the doctor can recommend a chemotherapy called ETC with a Trastuzumab therapy that starts with a higher starting dose. Both therapy plans include an antihormone therapy with Tamoxifen/aromatase inhibitors parallel to the antibody therapy.

We created the data sets to be processed with InteKRator by combining each instance data set of Mamma-DSCS with the respective output. As a result, we obtain a collection of semi-synthetic data sets comprising all representative instance data.

Note that there are various ways to create data sets for the usage with the InteKRator toolbox. In fact, besides the manual compilation of data set, any structured collection of knowledge (in this case about therapy recommendations) can be used as input for InteKRator as long as it can be parsed to InteKRator's straightforward syntax.

3.3. Learning Comprehensible Knowledge Bases with the InteKRator Toolbox

Since it should be possible to infer therapy plan recommendations from the learned knowledge (where each therapy plan contains multiple subsequent therapies), one separate knowledge base KB_i will be learned from each data set D_i. Thus, each knowledge base KB_i will comprise the knowledge about one type of therapy involved in the therapy plan. The learning process is similar for all knowledge bases:

(1) The knowledge about a therapy type is learned from the i-th data set D_i using InteKRator's learning module (see [2], Algorithm 1 for details), resulting in a knowledge base KB_i in the form of *rules with exceptions* (see Figure **3**). *Missing values* are not included in rule premises during the learning process.[4]

(2) A revision of KB_i is performed for every row of D_i using InteKRator's revision module: This adds some potentially missing rules on the bottom most level of KB_i that are needed for completeness to fully explain D_i.[5]

These steps are repeated until one knowledge base is learned for every data set.

[4] For this purpose, the learning algorithm of the InteKRator toolbox has been modified, which will be included in one of the next releases of the InteKRator toolbox.

[5] Note that even without revision, already ≈ 99,98% of the ASP inferences could be covered.

Topmost level: most general rule with an empty premise (lower levels comprise more specific rules with longer premises)

```
AHT_0.NONE [0.46794871794871795]

age_gt70 ^ ci_ant ^ nodal_pos ^ schedule_adj -> AHT_0.UNKNOWN [1.0]
NEG_ci_ant ^ age_lte50 ^ her_pos ^ schedule_neoAdj -> AHT_0.NONE [0.5370370370370371]
NEG_ci_ant ^ her_pos ^ nodal_pos ^ schedule_neoAdj -> AHT_0.NONE [0.6428571428571429]
AbT-PRE_0.NONE ^ AbT_0.NONE ^ her_neg ^ nodal_neg -> AHT_0.NONE [0.47115384615384615]
AbT-PRE_0.NONE ^ age_gt70 ^ her_pos ^ nodal_neg -> AHT_0.NONE [0.4122137404580153]
AbT-PRE_0.NONE ^ age_lte50 ^ her_pos ^ nodal_neg -> AHT_0.NONE [0.43609022556390975]
NEG_ci_ant ^ her_pos ^ nodal_neg ^ schedule_neoAdj -> AHT_0.NONE [0.4444444444444444]
AbT-PRE_0.NONE ^ AbT_0.NONE ^ age_lte50 ^ her_neg -> AHT_0.NONE [0.49746192893401014]
ci_ant ^ her_neg ^ nodal_pos ^ schedule_adj -> AHT_0.UNKNOWN [0.5]

AbT-PRE_0.NONE ^ OP_0.NONE ^ epr_pos ^ her_pos ^ schedule_adj -> AHT_2.AI5-TMX5_2.TMX10 [0.5149700598802395]
AbT-PRE_0.NONE ^ OP_0.NONE ^ age_lte70 ^ epr_pos ^ schedule_adj -> AHT_2.AI5-TMX5_2.TMX10 [0.7164179104477612]
NEG_ci_ant ^ age_lte70 ^ gmc_average ^ her_pos ^ nodal_pos -> AHT_0.UNKNOWN [1.0]
NEG_ci_ant ^ age_lte70 ^ her_pos ^ nodal_pos ^ schedule_adj -> AHT_0.UNKNOWN [0.3333333333333333]

...
```

Note that age_lte50 induces a subset of age_lte70 (resulting from inconsequent/incomplete categories in the mind map). For queries with patient age < 51, both age attributes need to be provided, which may result in too many therapy recommendations.

Figure 3. Excerpt from the Learned Knowledge for AHT (antihormone therapy): Rules are denoted by "`premise -> conclusion [p]`" (with ^ used for conjunction, _ used to separate prefix and alternative therapy options, and `p` being the conditional probability of the conclusion given the premise). Starting on the topmost level, the knowledge can be easily read top-down as "Usually, there will be no AHT (with `p` ≈ 0.47). Except for attributes on the next level, where it is either no AHT or it is unknown. If, e. g., no AbT-PRE (antibody therapy pre-operation), no operation, epr_pos (ER/PR-positive), her_pos (HER2-positive) and schedule_adj (adjuvant therapy) is known (cf. 1st rule of the 3rd level), then the AHT will be either AI5 (aromatase inhibitors) for at least 5 years followed by TMX5 (Tamoxifen) for at least 5 years, or TMX10 (Tamoxifen) for at least 10 years (with `p` ≈ 0.51)".

3.4. Extended use of InteKRator's Reasoning System

Unfortunately, the resulting learned knowledge bases are less expressive than the ASP approach described in [12] (see also Section 1.1). To adapt the inferences to the answer sets of the preliminary approach from [12], the following extended inference approach has been developed using InteKRator's reasoning module:

(1) The knowledge base KB_i is requested for inferring adequate therapies of a therapy plan.
(2) For each therapy inferred from KB_i, the therapy serves as an additional input for inferring the therapy of a subsequent type from KB_{i+1}.

These steps are repeated starting from $i = 1$ for all KB_i (except for the last one for which only step (1) is executed).

The described approach results in an inference tree, where tree leaves represent the answer sets of the original ASP [12] (see Figure **1**). By this means, the answer sets can be adequately reflected when inferences are derived from the learned knowledge.

4. Implementation

The extended reasoning system from Section 3.4 is implemented as a prototype in Java using the InteKRator toolbox as a library. This results in a lightweight console application which is potentially easy to adapt and to integrate into a web application.

The inferences are based on the learned knowledge bases from Section 3.3, which are provided in plain text files. In conjunction with the knowledge being represented in a hierarchical way, this renders the knowledge easy to inspect and can support the creation of user interfaces later, e. g., to outline the most important aspects or to let a medical user browse the knowledge. Moreover, the learned knowledge bases are easy to revise or can be replaced by a new version in case parts of the knowledge changes.

The source code of the prototype is available under the GNU General Public License (GPL) version 3.[6]

5. Lessons learned

5.1. Evaluation of and Comparison to the Mind Map and ASP-based solution

Compared to the mind map and the ASP approach (see Section 2.1), the approach introduced here shows advantages concerning the comprehensibility of the represented knowledge, transparency of the provided inferences and reasoning time. The comprehensibility and transparency aspects are covered by InteKRator's reasoning system, which allows for reporting the patient attributes from which the resulting inferences have been derived while adding quantitative information to the inferences (i. e., (conditional) probabilities of conclusions given attributes of possible patients).

Example 3. For P_1 from the previous examples, the ASP approach reports four therapy plan recommendations as encoded in the mind map (see Figure **2**). One of the recommended plans comprises ETC for CT, H8-6 for AbT and AI5-TMX5 for AHT. For the same patient, the InteKRator approach additionally provides explanations and adds quantitative information. E. g., for CT of the therapy plan, the rule

```
age_lte50 ^ ci_ant ^ gmc_good ^ her_pos ^ nodal_pos ^ schedule_adj -> CT_1.ETC_1.TCH [1.0]
```

is provided, meaning that either ETC or TCH can be derived through this rule from the patient attributes (which holds for 100% of the cases in the data from which the rule had been learned). Similarly, e. g., for AbT, H8-6 can be derived through the rule

```
AbT-PRE_0.NONE ^ OP_0.NONE ^ age_lte50 ^ epr_pos ^ gmc_good ^ her_pos ^ schedule_adj
                                                                  -> AbT_2.H8-6 [0.71]
```
[7].

Table **1** summarizes the comparison of the three approaches. Note that the overall number of rules is ≈ 3.5 times higher with the InteKRator approach. However, a large amount results from the knowledge bases being learned separately for each therapy type (see Figure **1**). While this might induce redundancies, it may also contribute to the comprehensibility, as the knowledge can be easily read top down for each therapy of the respective type. Moreover, usually not all rules need to be considered since the rules covering the most common cases are usually located toward the top of the knowledge bases learned with InteKRator.

---

[6] https://gitlab.com/dapel1/sequentialdecisionsupport

[7] Rule weight rounded to two digits.

Table 1. Comparison of the Preliminary Approaches and the InteKRator Approach: While the knowledge learned by the InteKRator toolbox results in a larger number of rules (separated by therapy types), it allows for much faster reasoning with a top-down view on the knowledge. It also provides quantitative information by default and allows for maintaining knowledge by (re)learning or revision. While the learned knowledge learned in principle has a lower expressivity than ASP, in this case equivalency to ASP has been achieved.

	Mind Map	**ASP**	**InteKRator**
Number of Rules	(not applicable)	≈ 250	$153_{CT} + 211_{AbT\text{-}PRE} + 43_{OP} + 200_{AbT} + 287_{AHT} = 894$
Avg. Reasoning Time per Patient	(not applicable)	≈ 3.35 sec.	$\approx 1.067 \cdot 10^{-2}$ sec.
Readability	via node expansion	formal rules	formal rules (top-down)
Quantitative Information	no	no	yes
Explanation of Inferences	manually	via extensions [7]	yes
Maintainability	manually, requires expert knowledge	manually, requires expert knowledge	by learning/revision, learning requires data
Expressivity	very high (nodes contain natural language)	high	in principle lower than ASP; but equivalent with the extensions in the presented use case

5.2. Results

While the semi-formal mind map and the formal ASP approach is manually modeled, the InteKRator approach allows for learning such knowledge from data. Due to the comprehensible top-down structure of the resulting knowledge bases, even larger amounts of such learned knowledge can be read top-down to gain an overview over the data. However, in comparison to ASP, the expressivity of the learned knowledge bases is limited. In this work, we overcome these limitations by learning one knowledge base for each therapy type and by exploiting the idea of an inference tree (cf. Figure **1**).

Apart from the possibility of automatically learning knowledge bases from data, InteKRator provides additional quantitative information both for the knowledge and the inferences thereof and offers the possibility of providing explanations of the inferences. Concerning the final accuracy, 100% of the therapy recommendations provided by the ASP approach could also be inferred with the InteKRator-based approach.

6. Conclusion

We have shown a symbiotic relationship between ASP programs and the InteKRator approach that can be exploited to facilitate the usage of expert systems in the medical field. By using semi-synthetic data sets of the ASP program from [12], InteKRator was able to learn knowledge identical to that encoded in the ASP program with respect to the inferences. The presented work, thereby, combines the positive aspects of both manual knowledge modeling and machine learning approaches to CDSSs. In this con-text, InteKRator provides several benefits for medical doctors especially in combi-nation with ASP-based expert systems: Based on the ASP-induced learned knowledge, InteKRator is innately able to provide explanations of recommended therapy plans and it adds a quantitative layer to the rules which might help medical doctors to incorporate the recommendations more easily into their decisions. Moreover, it offers the possibility of directly learning from real-world patient data while offering faster reasoning.

The presented approach is in principle also generalizable to other domains, provided that the manually modeled ASP is created anew in the context of the respective domain. The InteKRator-based learning of knowledge bases from real-world data is also applicable to other domains, provided that the data exists in an eligible format.

Future work could comprise the creation of a clinical system based on the presented (or similar) techniques, e. g., by involving multiple independent approaches, which could increase the trustworthiness of such AI-related systems in medicine.

Declarations

Contributions of the authors: The concept and major parts of this paper as well as implementation works were contributed equivalently by AT and DA (shared first authorship). AT contributed more from the ASP side while DA contributed more from the InteKRator perspective. Data preparation was jointly conducted by AT and DA with the additional help of a student assistant. Therapy data and the medical knowledge was provided by the pharmaceutical department and the clinic of Hematology and Medical Oncology of the St.-Johannes-Hospital in Dortmund (RGM and MN). Preliminary works were conducted by AT under supervision of GKI. Moreover, GKI as well TP contributed by reviewing and providing remarks. MN pointed to additional literature. RGM is heading the Clinic of Hematology and Oncology of the St.-Johannes-Hospital in Dortmund, Germany and was involved in consulting activities from the early beginning of the collaboration with the Information Engineering group (headed by GKI) at TU Dortmund University. All authors have approved the manuscript.

Conflict of Interest: This work compares/incorporates approaches by authors also contributing to this paper.

Acknowledgments

The research in this paper is partly funded by Carl Zeiss Foundation, grant number P2021-02-014 (TOPML project).

References

[1] Apeldoorn D, Dockhorn A. Exception-Tolerant Hierarchical Knowledge Bases for Forward Model Learning. IEEE Transaction on Games. 2021;13(3):249–62. DOI: 10.1109/TG.2020.3008002
[2] Apeldoorn D, Hadidi L, Panholzer T. Learning Behavioral Rules from Multi-Agent Simulations for Optimizing Hospital Processes. In: Chomphuwiset P, Kim J, Pawara P, editors. Multi-disciplinary Trends in Artificial Intelligence – 14th International Conference, MIWAI 2021, Virtual Event, July 2–3, 2021, Proceedings. Cham: Springer; 2021. p. 14–26.
[3] Apeldoorn D, Panholzer T. Automated Creation of Expert Systems with the InteKRator Toolbox. Studies in Health Technology and Informatics. 2021;283:46–55. DOI: 10.3233/SHTI210540
[4] Apeldoorn D, Kern-Isberner G. Towards an Understanding of What is Learned: Extracting Multi-Abstraction-Level Knowledge from Learning Agents. In: Rus V, Markov Z, editors. Proceedings of the Thirtieth International Florida Artificial Intelligence Research Society Conference. Palo Alto (CA): AAAI Press; 2017. p. 764–67.

[5] Brewka G, Eiter T, Truszczyński M. Answer Set Programming at a Glance. Commun. ACM. 2011;54(12):92–103.
[6] Dockhorn A, Apeldoorn D. Forward Model Approximation for General Video Game Learning. In: Browne C, Winands MHM, Liu J, Preuss M, editors. Proceedings of the 2018 IEEE Conference on Computational Intelligence and Games (CIG'18). Piscataway: IEEE; 2018. p. 425–32.
[7] Fandinno J, Schulz C. Answering the "why" in answer set programming - A survey of explanation approaches. Theory Pract. Log. Program. 2019;19(2):114-203.
[8] InteKRator Toolbox: https://www.gitlab.com/dapel1/intekrator_toolbox, accessed on Mar 30th, 2023.
[9] Quinlan JR. Induction of Decision Trees. Mach. Learn. 1986;1(1):81-106.
[10] Somashekhar S, Sepúlveda M-J, Puglielli S, Norden A, Shortliffe E, Rohit Kumar C, Rauthan A, Arun Kumar N, Patil P, Rhee K, Ramya Y. Watson for Oncology and breast cancer treatment recommendations: agreement with an expert multidisciplinary tumor board. In: Annals of Oncology, 2018;29:418-423.
[11] Sutton RT, Pincock D, Baumgart DC, Sadowski DC, Fedorak RN, Kroeker, KI. An overview of clinical decision support systems: benefits, risks, and strategies for success. In: NPJ Digital Med. 2020.
[12] Thevapalan A, Kern-Isberner G, Howey D, Beierle C, Meyer R, Nietzke M. Decision Support Core System for Cancer Therapies Using ASP-HEX. In: Brawner K, Rus V, editors. Proceedings of the Thirty-First International Florida Artificial Intelligence Research Society Conference, FLAIRS 2018, Melbourne, Florida, USA. May 21-23 2018. AAAI Press; 2018. p. 531–36.