Artificial Intelligence In Medicine 168 (2025) 103226

Contents lists available at ScienceDirect

Artificial Intelligence In Medicine

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

Research paper

Ontology-based student testing through clinical guidelines: An AI approach

Alessio Bottrighi [a,b,*], Antonio Maconi [c], Stefano Nera [a,b], Luca Piovesan [a,b], Erica Raina [a,b], Paolo Terenziani [a,b]

[a] Computer Science Institute, DISIT, University of Eastern Piedmont, Alessandria, Italy
[b] Integrated Laboratory of AI and Medical Informatics, DAIRI, SS. Antonio e Biagio e Cesare Arrigo Hospital, Alessandria - DISIT - University of Eastern Piedmont, Italy
[c] DAIRI, SS. Antonio e Biagio e Cesare Arrigo Hospital, Alessandria, Italy

ARTICLE INFO

Keywords:
Computer interpretable clinical guidelines
Medical education
Knowledge representation and reasoning
Ontologies
Testing and explanation

ABSTRACT

On the basis of our 25-year experience with the GLARE (Guideline Acquisition, Representation and Execution) clinical decision support system, we have started to analyze the adoption of computer-interpretable clinical guidelines (CIGs) and AI techniques to ***train and test medical students about how to act on patients***. Moving from decision support to the educational task involves significant research challenges. In this paper, we propose a new facility that supports teachers in the definition of tests, by selecting and hiding to students specific parts of the CIG, and asking students how they would act on the given case study (patient) in the selected parts. Students are provided with a medical ontology to identify proper actions/decisions, and students' proposals are then automatically compared with what the CIG (considered as a "golden standard") would suggest to do to the patient through knowledge representation and reasoning techniques. Our basic explanation mechanism exploits the medical ontology to show to students the differences (if any) between their proposals and the ones of the CIG.

1. Introduction

Medicine is a complex and challenging context for education. "Traditional" medical education spans over a wide amount of knowledge, ranging from the description of diseases, of their diagnoses and treatments, to the human anatomy. However, the problem of ***how to act on a specific patient applying the best medical procedures*** is usually not considered in textbooks, and can be learned by medical students (and healthcare personnel in general) only by practicing. "*In the current healthcare system, novices … climb the learning curve working on real patients with different levels of guidance.*" [1].

In the medical literature, the evidence-based "golden standards" about how to act on patients affected by a specific disease are encoded into clinical practice guidelines (CPGs). CPGs are "*systematically developed statements to assist practitioner and patient decisions about appropriate healthcare under specific clinical circumstances*" [2]. Thousands of CPGs have already been developed all around the world. For example, the Guideline International Network (http://www.g-i-n.net) gathers 106 organizations in 54 countries, and provides a library of >6500 CPGs.

However, CPGs in the medical literature are usually encoded by large (hundreds of pages) bodies of text containing knowledge and recommendations about the treatment of a specific disease. Thus, it is difficult for a physician at the point of care to identify in such large texts the recommendations which are appropriate for the specific patient at hand. Thus, since the 90s', in the area of Artificial Intelligence in Medicine (AIM) many CPG-independent *clinical decision support systems* have been developed to manage Computer-Interpretable versions of Guidelines (CIGs). Some of them (the list is not exhaustive) are: Asbru [3], EON [4], DeGel [5], GLARE [6,7], GLIF [8], GPROVE [9], GUIDE [10], PRODIGY [11], PROforma [12,13], SAGE [14]. In 2008, in her book, Ten Teije [15] proposed an assessment of the state-of-the-art. Comparisons among some of the different approaches and surveys can be found in [16–23]. Very recently, Van Woensel et al. have presented a comparative analysis of CIG systems dealing with comorbid patients [24].

Most of such systems support the acquisition of CPGs in a computer-interpretable format, and support physicians in the application of the acquired CIGs to specific patients, through an automatic connection to the patient's electronic clinical record. Though some CIG decision support system has also been applied in medical education (consider, e.g., [25–29]), they are born for decision support. And, indeed, ***the specificity of the educational task arises new challenges*** (with respect to decision

* Corresponding author at: Computer Science Institute, DISIT, University of Eastern Piedmont, Alessandria, Italy.
E-mail addresses: alessio.bottrighi@uniupo.it (A. Bottrighi), amaconi@ospedale.al.it (A. Maconi), stefano.nera@uniupo.it (S. Nera), luca.piovesan@uniupo.it (L. Piovesan), erica.raina@uniupo.it (E. Raina), paolo.terenziani@uniupo.it (P. Terenziani).

https://doi.org/10.1016/j.artmed.2025.103226
Received 14 November 2024; Received in revised form 19 March 2025; Accepted 15 July 2025
Available online 16 July 2025

support).

In 2019 we started our research in medical education, identifying a preliminary research roadmap to build specific educational facilities on top of a CIG system [30]. Notably, for the sake of simplicity, in this paper we use the term "students" to indicate the users of our educational system. However, the approach we are going to present can be used both by medical students and by physicians, for their continuous education.

In this paper, we discuss a novel facility, the students' test facility, that we have designed during the *Personalized Training of Professional Competencies with AI* (PTPC-AI) project, started in May 2023. The goal of our testing facility is to give students a tool to check their knowledge about a given CIG (notably, our approach is CIG-independent), and to test their ability to apply CIG's recommendations to patients. Indeed, the basic idea behind our testing approach is quite simple: instead of using a CIG system to suggest how to act on a patient (i.e., as a decision support system), we use it as a golden standard to compare "what should be done" on a patient with respect to "what students suggest to do" on the patient, and as a starting point (together with ontological medical knowledge, and applying "symbolic AI" knowledge representation and reasoning techniques) to provide explanations about possible differences. In other words, our idea is to design and implement an "*ask student(s), check conformance, explain discrepancies*" process. Notably, in our approach, patients (case studies) are entirely described by the evolution of their findings, and the general research question we address is:

- How can CIG be exploited for students' testing about how to manage specific case studies?

More specifically, as we will discuss in more detail in Section 3, we focus on two main (sub) research questions:

RQ1. How to facilitate the generation and use of "local" tests, focusing on the application (to case studies) of specific subparts of a CIG?
RQ2. How to balance the level of freedom provided to students (to answer questions) and the feasibility of checking correctness and providing knowledge-based explanations?

The paper addresses such research questions, discussing the architecture, data structures and reasoning mechanisms we designed to deal with them. The main original methodological contributions we propose are:

- An innovative ***test acquisition approach***, which, given as input a case study and a CIG, uses CIG execution to find the correct subgraph, and provides a graphical interface (to facilitate man-machine interaction) to specify a query focusing on specific subparts of the sub-graph (to address **RQ1** above);
- An ***ontology-based approach***, in which navigation is used to support students' query answering, and path retrieval is used to propose knowledge-based explanations (to address **RQ2** above).

Additionally, though we realized our approach on top of GLARE, we think that a relevant advantage of our approach lies in its ***generality***, as discussed in the concluding Section.

The paper is organized as follows. In Section 2, we provide the background knowledge, sketching our GLARE approach (to make our paper self-contained), and the goals of the PTPC-AI project [31] (to settle our contribution in its general context). In Section 3, we introduce the main goals of the approach proposed in this paper. In Section 4, we propose the system architecture and (main) data structures. In sections 5 and 6, we describe how our facility supports CIG-test execution, by describing the treatment of hidden nodes (called shadow nodes; see Section 5) and hidden parts of the CIG (called holes; see Section 6). In Section 7, we present a (preliminary) experimental evaluation of our approach. Finally, Section 8 reports conclusions, related works, and future works.

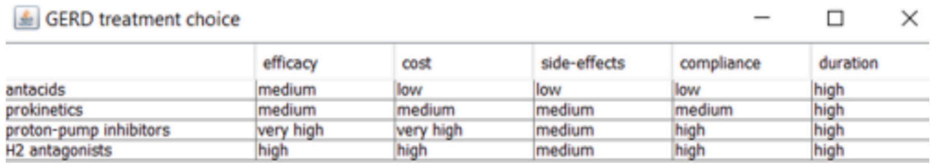
2. Background: GLARE, GLARE-Edu and the PTPC-AI project

GLARE (Guideline Acquisition, Representation and Execution) [6] constitutes our contribution to the area of ***disease-independent*** CIG decision support systems. We started to work on GLARE in 1996, cooperating with some physicians of the Hospital San Giovanni Battista in Turin (one of the major hospitals in Italy), led by Prof. Gianpaolo Molino. GLARE has already been applied to different medical domains, including bladder cancer, polytrauma, ischemic stroke, gastroesophageal reflux disease, heart failure, alcohol-related diseases, and melanoma. The kernel of GLARE (see [6]) consists of three modules: a module to ***acquire*** CPGs in computer format (i.e., into a CIG), one to ***navigate*** the acquired CIGs, and one to support the ***execution*** (application) of a CIG on a specific patient, considering the data in the clinical record of the patient.

Since 2000s', we started, taking advantage of AI techniques, to extend GLARE's kernel through the addition of new facilities, such as CIG contextualization, temporal reasoning, decision support based on decision theory, model-based verification, conformance analysis, support to the coordinated concurrent execution by multiple agents, support to the treatment of comorbid patients [32–43]. To make our paper self-contained, we briefly resume GLARE's representation formalism (subsection 2.1) and acquisition, navigation and execution mechanisms (subsection 2.2). Moreover, we discuss the context of the proposed research, by sketching the main goals and the research program of the PTPC-AI project (subsection 2.3).

2.1. GLARE's Representation formalism

GLARE, as well as many disease-independent CIG systems, represents CIGs adopting the Task-Network Model (TNM): a CIG is a hierarchical oriented graph, where nodes represent the actions and arcs represent the flow of control. GLARE's representation formalism provides both *atomic* and *composite* action nodes. Atomic actions represent "elementary" steps in a CPG (i.e., they are directly described), while composite actions are defined in terms of (the subgraph of) their components (which, in turn, may be atomic or composite). GLARE provides six different types of atomic actions. ***Work actions*** (see, e.g., circle node labelled "Incisional biopsy of skin lesion" in Fig. 2) represent atomic actions described through a set of attributes, including a link to an ontological concept (e.g. a SNOMED CT code), name, goals, cost, time, resources, and textual description. ***Pharmacological prescriptions*** model drug prescriptions, including posology. ***Query actions*** (see, e.g., rhombus node labelled "Skin pigmentation" in Fig. 2) represent data requests (including, e.g., laboratory tests). ***Decision actions*** (see, e.g., diamond node labelled "Presence of skin lesion?" in Fig. 2) model medical decisions, describing the criteria to select between alternative paths in a CIG. GLARE distinguishes two types of decisions. ***Patient-status-based decisions*** (PSB decisions for short) model cases in which alternatives have to be selected on the basis of the *clinical* status of the patient (e.g., diagnostic decisions). In the current version of GLARE, PSB decision criteria can be represented as *Decision Trees*, *Score-based formulae*, or *Boolean formulae*. For brevity, in the following we focus only on Boolean formulae, and assume they are expressed in disjunctive normal form. Notice that each atomic term in GLARE's Boolean conditions is expressed as an inequality involving a datum (expressed by a pair <data, attribute>; in the rest of this paper we term such pairs "**parameters**") and a constant value (e.g., an example of Boolean condition is "$<$*temperature, value*$>$ $\geq$ *38 AND* $<$*liver, dimension*$>$ $=$ *enlarged*"). On the other hand, GLARE's ***cost-benefit decisions*** model cases in which the choice among alternatives has to be made based on a *cost-benefit analysis* based on general parameters, like *effectiveness, cost, side-effects, compliance, and duration*. A typical case of cost-benefit decision in CIGs is the choice among clinically equivalent alternative therapies. In GLARE, a cost-benefit decision is modeled by the qualitative evaluation of the above parameters, for each one of the alternatives to be discriminated. For instance, Fig. 1


GERD treatment choice

	efficacy	cost	side-effects	compliance	duration
antacids	medium	low	low	low	high
prokinetics	medium	medium	medium	medium	high
proton-pump inhibitors	very high	very high	medium	high	high
H2 antagonists	high	high	medium	high	high

Fig. 1. Tabular representation of the cost-benefit decision for "possible GERD", as shown in GLARE acquisition module.

shows a possible way of modeling the cost-benefit decision among four different treatments in the case of "possible GERD" (Gastro-Esophageal Reflux Disease), in the tabular form which is used by GLARE's acquisition module (explained in Section 2.2 below). In such table, for each treatment, listed in rows, a different evaluation can be associated to each parameter, listed in columns (e.g., treatment "antacids" has "medium" evaluation for parameter "efficacy").

Conclusions are used to improve CIG's readability, by showing the output of decisions (see, e.g., the triangular nodes labelled "Suspected skin lesion" and "EXIT" in Fig. 2). Finally, ***composite actions*** are defined in terms of the subgraph of the actions composing them.

Notably, GLARE's node description supports the introduction of links to ontological (e.g., SNOMED CT) concepts, to enhance the CIG standardization. Such an aspect is important also to support the testing facility proposed in this paper, as shown in Sections 5 and 6 below.

Finally, GLARE's arcs describe the ***control relations*** between action-nodes. GLARE provides four different control relations: *sequence, concurrent, alternative and repetition*. ***Temporal constraints*** (e.g., "*between 2 and 3 hours after*") may be associated with arcs.

2.2. GLARE's Acquisition, navigation and execution modules

GLARE's ***acquisition module*** provides medical experts and knowledge engineers (who cooperate in the acquisition) with user-friendly

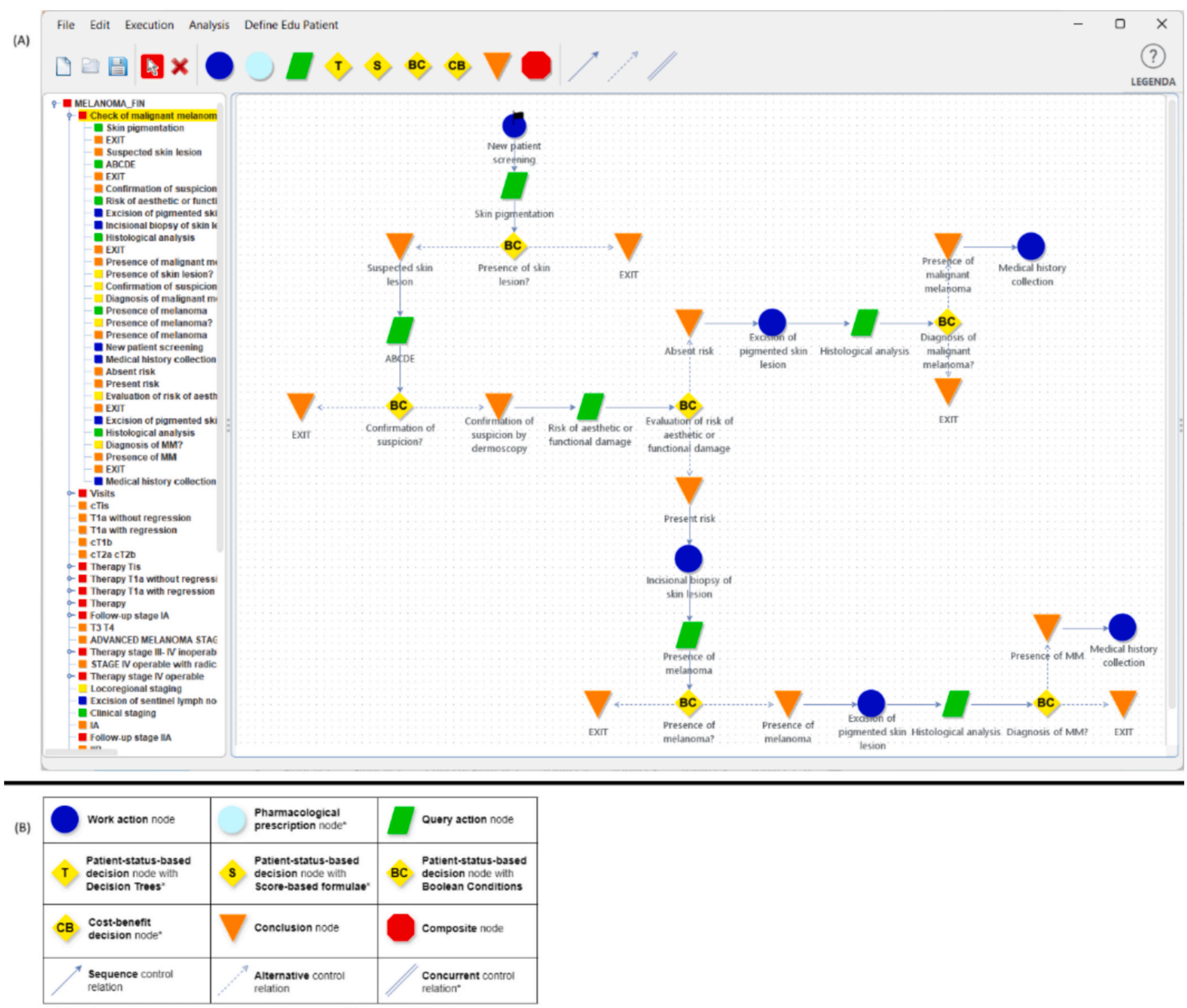


Fig. 2. A part of the CPG of melanoma: representation and graphical interface (A) and a legenda of actions and arcs (B).

support to model CPGs into GLARE's formalism. It provides a graphical interface, which allows to "draw" (in a "constrained" way) the graph representing a CPG, and provides action/arc-type-dependent windows to acquire the internal descriptions of nodes and arcs, as well as facilities for having a global hierarchical view of the graph, and for browsing it. For example, Fig. 2(A) shows GLARE's acquisition interface, considering as an example a part of the CPG for melanoma. The left part of Fig. 2(A) displays the window representing the (tree-like) hierarchy of the actions of the CPG. By clicking on the composite nodes in this window, the user triggers the window (right part of Fig. 2(A)) to acquire its subgraph. The right part of Fig. 2(A) shows the window used to acquire a graph. Different forms and colors are used to distinguish among different types of actions, and arcs model (the different types of) control relations between them (the different types of actions and arcs are listed in the legenda in the Fig. 2(B)). By clicking on the nodes of this window, the user triggers the pop-up windows to acquire their internal description (attributes and parameters).

GLARE's ***navigation module*** supports physicians and students in the navigation of a CIG. The user can navigate the multi-level graph (remember that composite nodes are, in turn, described by the graph of their components), and look at the internal description of each node and arc.

Finally, GLARE's ***execution module*** supports physicians in the application of a CIG to a specific patient. The electronic clinical record of the patient is used by the execution engine to suggest evidence-based decisions to the physician, who can follow or override GLARE's suggestions.

2.3. GLARE-Edu and the personalized training of professional competences with AI (PTPC-AI) project

We started our research in the educational area in 2019, proposing a roadmap for the definition of GLARE-Edu, a CIG system specifically designed to support medical education [30].

GLARE-Edu is developed in Java (version 8), as a stand-alone application. GLARE-Edu uses mxGraph library to store and show CIGs' graph (https://jgraph.github.io/mxgraph/). To facilitate the GLARE-Edu installation, we provided students with an executable version for Windows and MacOS, downloadable from a folder in Google Drive. CIGs and tests are loaded in Dropbox, as XML files. Tests' results are also uploaded in Dropbox automatically. Generalities about GLARE-Edu can be found on the project website (https://sites.google.com/uniupo.it/glareedu), where we also provide a demo version of our system and information about its availability (i.e., it is available under request for scientific, academic and non-profit purposes – see https://sites.google.com/uniupo.it/glareedu/materials).

Notably, GLARE-Edu takes advantage, as much as possible, of our previous work in GLARE. Basically, it adopts GLARE's formalism to represent CIGs (with few extensions). As a consequence, GLARE-Edu can exploit

- GLARE's ***acquisition*** tool for CIG acquisition,
- GLARE's ***navigation*** tool to support students in the navigation of the CIG, and
- GLARE's ***execution*** tool (with proper modifications) can be used to show students, step-by-step, how a CIG would recommend to act on a specific patient. Notably, the "educational" execution provided in this way is completely automatic (i.e., system-driven), and students just see (step by step) the results of the application of a CIG to a patient.

However, in our research activity in this area, we are aiming to provide students with additional (and more "advanced") learning facilities, such as simulation facilities in which students play an "active" role. Actually, we are carrying on our research in the educational context within the *Personalized Training of Professional Competencies with AI* (PTPC-AI) project [31], a part of a larger two-year national project, *Learning Personalization with AI and of AI* (AI-LEAP). Specifically, PTPC-AI involves also Alessandria Hospital, and aims at identifying, designing and implementing a suite of new education-oriented facilities for CIG-based medical education. While the proposed approach will be completely domain-independent, in the project we will consider melanoma as a case study. In the first part of the project, we have already acquired (using GLARE) the Italian CPG for melanoma, as well as a set of case studies (i.e., data modeling the evolution of real or invented patients affected by melanoma). We are now in the process of adding new facilities, including the testing facility proposed in this paper. An extensive experimental evaluation of our approach will be performed in the last six months of the project (see future work in Section 8).

3. Main goals and features of our educational approach

The goal of our approach to medical education is the development and testing of a suite of facilities supporting medical students in learning "how to act" on patients based on the evidence-based "best practices" provided by CPGs.

Specifically, together with the physicians of the Alessandria Hospital, we have identified three main focuses for our educational approach. We aim at teaching students (given a CIG dealing with a specific disease):

(A1) What actions have to be performed on a specific patient, and in which temporal ordering
(A2) Which data and laboratory examinations are needed to deal with the patient
(A3) How to make the correct (diagnostic and therapeutic) decisions about a specific patient

The ***navigation*** and ***execution*** tools mentioned so far aim at supporting in the study of a CPG (navigation) and its application to specific case studies (execution), with specific focus on aspects (A1 – A3) above.

However, after some students' study, it is also important to test students' competences. In this paper, we focus on the educational facility that we have recently designed for ***students' testing***.

The basic idea is simple: we use computerized CPGs (i.e., CIGs) as a "golden standard" about how to act on patients affected by a given disease. Given the description of the status of the patient (notably, since we are not interested to show students how to "physically" act on patients, we do not need virtual reality techniques to model patients), which includes the current findings of the patient and, possibly, the part of CIG that has been already enacted on the patient, a part of the CIG is "hidden" to the students, and the student is asked ***how to act*** on the patient. Students have to autonomously propose *which atomic actions have to be performed*, and *which decisions have to be taken* on the patient in the "hidden" parts. The students' proposals are compared with the recommendations that would have been provided by the CIG, and explanations to students are automatically provided in the case of discrepancies (between the students' recommendations and the CIG's ones).

Different issues must be considered to make concrete the abstract idea above. First of all, we have to identify more precisely what we want to hide (i.e., ask) to students, and provide tools to make such "queries" possible. We have to specify:

A. The ***extension*** of the queries (whole guideline vs a part of it)
B. The ***modality*** of the queries (*shadow* nodes vs *hole* nodes; see below)
C. The ***focus*** of the queries. A *choice about what pieces of information have to be filled by students* must be made.
D. The range of **possible students' answers**. We have to identify what are the "***answer***" we want to offer to students (notably, such an issue deeply affects the check of conformance and the explanation process).

Concerning **issue (A)**, the wider test we can provide to students is the description of a case study (patient) affected by a given disease, and not already treated for it, asking them how they would act on the patient "from the beginning to the end".

However, such general exercise may be very long and, above all, too demanding. Therefore, we support the possibility of proposing students more specific exercises, considering the status of a patient at a given point in the CIG application, and asking about just the next action, or a pre-definite part of the CIG (thus addressing the research question **RQ1** in the introduction).

To do so, we introduce two query modalities (**issue B** above): *shadow* nodes and *hole* nodes. A **shadow node** is a node of a given action type (e.g., work action, cost-benefit decision, etc.) in the original CIG in which all pieces of information (including, e.g., the name of the action in the case of work action nodes) are hidden. The student is called to detect what is the hidden action, or, in the case of patient-status-based decisions, to say what decision s/he would take for the patient.

Shadow nodes constitute a quite elementary and easy form of test, concerning a specific node in the CIG. However, more challenging tests can be performed if, instead of focusing on a single node (whose action type is shown to students), entire connected parts of a CIG are considered. In such a context, students have to face additional challenges: they have also to understand which types of actions have to be performed, and in which order. We achieve such a goal through the introduction of **hole nodes**, which act as "placeholders" for entire parts of a CIG. Also, in the case of hole nodes, students are called to complete the CIG, adding how they would operate on the patient in the "hidden" part of the CIG. Notably, a query concerning a whole CIG can be assimilated to the case in which only a hole node -hiding the whole CIG- is shown to students.

Considering **issue (C)**, independently of whether we consider shadow or hole nodes, we have to identify ***what are the relevant aspects that we want to address in our educational approach***. Indeed, as briefly mentioned in Section 2 above, the internal description of nodes is quite complex, and requiring students to learn all details is not the goal of our approach. Our choice about what pieces of information have to be completed by students (depending on the different action-types) is discussed below, and is guided by the goal of addressing issues (A1-A3) above.

Considering **issue (D)**, we have to identify what are the "**answers**" we want to offer to students. Indeed, such an issue is central for our whole educational facility, since it deeply affects the (complexity of defining) **check of conformance** (between the student's answer and the original CIG) and the **explanation process**.

For instance, in the case of *shadow* nodes, an easy solution could be to force students to select the missing pieces of information among a predefined menu of alternatives. Such a solution would make the check of conformance trivial and explanation quite easy, since each "wrong" alternative in the menu could be paired with a textual explanation of the reason why it is not appropriate. On the other hand, such a solution limits the possibilities for students, and requires an enormous preparatory work, since the experts defining the educational tests must acquire all "wrong" alternatives and relative explanations. In this paper, we propose an intermediate approach, aiming to balance the level of freedom provided to students (to answer questions) and the feasibility of checking correctness and providing knowledge-based explanations (thus addressing the research question **RQ2** identified in the introductory section).

We let students search for the missing pieces of information in a medical **ontology**, also used to define CIG's concepts. Conformance check and explanation can thus be realized through navigation/reasoning based on the ontological knowledge, and a limited effort is required by the experts preparing the educational tests.

Specifically, we adopt the SNOMED CT ontology [44], which is a wide ontology elaborated by the medical community, widely used in the medical context. Such an ontology includes 19 domains, organized in a polyhierarchical structure, with at most 29 levels; each hierarchy contains an ordered organization of concepts, linked together through IS_A relationships, in which a node may have one or more parents (multiple inheritance). Notably, SNOMED CT can be exploited in two main modalities. First, it can be navigated through several dedicated graphical browsers, which allow users to find concepts, learn about their properties, and explore the ontology along the IS_A relationship. Second, it can be paired with software reasoners, allowing taxonomical inferences and supporting query languages. We have designed and implemented a dedicated browser to support user navigation, and adopt the Apache JENA framework to reason and query the ontology.

In the rest of the paper, we will discuss our testing facility incrementally. We first provide a description of the whole architecture supporting CIG-tests acquisition and execution (Section 4). We then move to discuss the execution of shadow nodes (Section 5), and the execution of hole nodes (Section 6).

4. Overall architecture and data structures for CIG-tests acquisition

The testing facility proposed in this paper cannot be described without discussing the overall workflow in which it is inserted, and the system architecture and data/knowledge representation supporting it.

The workflow of our approach is shown in Fig. 3. First of all, knowledge engineers (KEs) and medical experts (MEs) have to acquire a CPG in computer format (i.e., a CIG). Then, given a CIG, KEs and Teachers (TEs) (or MEs) have to acquire case studies for it. At that point, given a CIG and a case study for it, TEs can elaborate tests. Finally, tests are executed by students.

The part of GLARE-Edu's architecture which is relevant for the testing facility is shown in Fig. 4, and the (relationship between) data/knowledge structures are shown in Fig. 5.

GLARE-Edu adopts GLARE's **CIG acquisition module** to acquire CPGs in computer format (e.g., CIGs). Notably, such a module interacts (through a dedicated browser) with SNOMED CT ontology, to support term standardization (see the top part of Fig. 4). Thus, the main entities in the acquired CIGs refer to the corresponding concept in SNOMED CT. For instance, the top part of Fig. 5 shows that each parameter required in a query action is linked to a SNOMED CT concept, as well as CIG work actions.

The **GLARE-Edu case studies acquisition module** supports experts plus knowledge engineers in the acquisition of case studies, each one representing the evolution of a patient along her/his treatment, adopting the proper CIG (see the bottom part of Fig. 4).[1] In synthesis, each step in the case study is an instantiation of an action-type in the CIG (see the links from concepts in the case study level and the CIG level in Fig. 5). Notably, the module supports the acquisition of a value for each one of the parameters required in query actions, thus acquiring the evolution of the clinical status of the patient [45].

The red parts of Fig. 4 show the new contributions proposed in this paper. Given a CIG and a case study (for such a CIG), a test is a request about what should be done on such a patient at given point(s) of

[1] Notice that CIG clinical decision support systems retrieve automatically patients' data from the electronic clinical records, which are managed by an autonomous DBMS in which data are inserted (ideally) as soon as they become available. On the other hand, in the educational task, each case study (in the form of the evolution of the clinical data of a patient) has to be defined and acquired entirely "a priori", to provide students a library of "exercises".

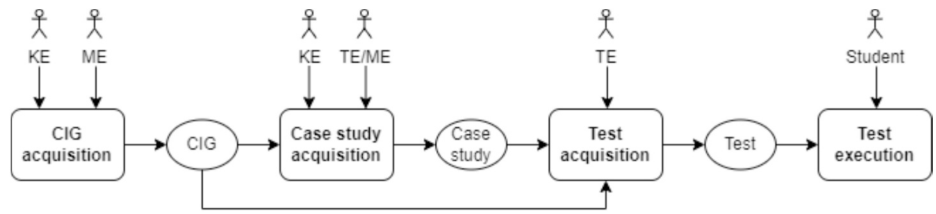


Fig. 3. Workflow of educational activities in GLARE-Edu. In the workflow, rounded rectangles represent tasks, and ovals represent data/knowledge structures.

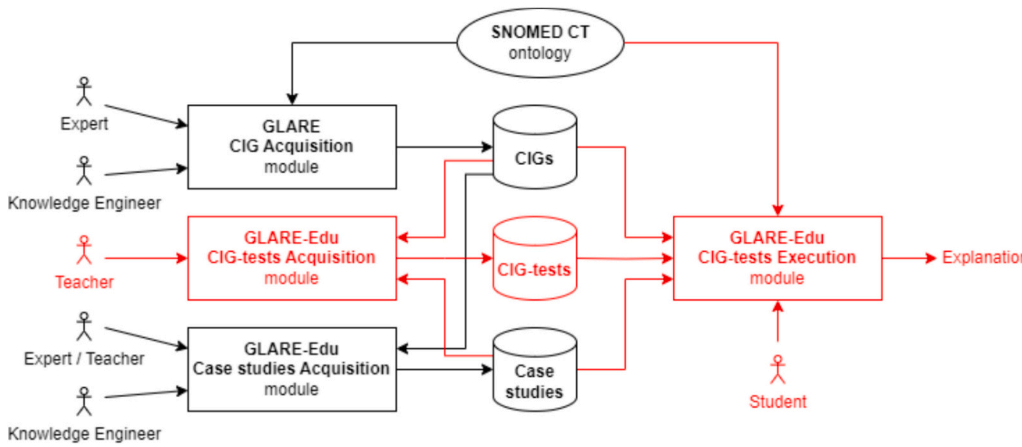


Fig. 4. Part of the GLARE-Edu architecture relevant for the testing facility. Red parts highlight the new contributions described in this paper. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

her/his diagnoses and treatment. As we will see in the following, we propose to show tests to students as a ***CIG-test*** in which specific nodes (shadow nodes, shown as grey nodes in the bottom part of Fig. 5), or connected parts of the CIG (hole nodes, not shown in Fig. 5[2]) are "hidden", and asking students how they would treat the patient (case study) at that point(s). **GLARE-Edu CIG-tests acquisition module** (see the central part of Fig. 4) supports teachers in the acquisition of such tests. Such a module is described in Section 4.1 below.

Once tests are defined, **GLARE-Edu CIG-test execution module** supports the student execution of a CIG-test. The input of such a module is a CIG, a CIG-test, and a case study. Broadly speaking, the module asks to the student how s/he would act on the patient at the "hidden" point(s) in the CIG-test, acquiring her/his answer (that must be provided using terms in the SNOMED CT ontology), comparing the student's answer to the action suggested by the CIG, and, in the case of discrepancies, showing them. For the sake of readability, we present the CIG-test execution module incrementally, by first discussing how to deal with CIG-tests concerning a specific action (i.e., a shadow node; see Section 5), and then generalizing to whole parts of the CIG (i.e., hole node(s); see Section 6).

In the rest of this section, we briefly discuss GLARE-Edu CIG-tests acquisition module.

[2] The goal of Figure 5 is to show the correspondence between concepts at the different levels of our representation. While there is a direct correspondence between shadow nodes and nodes at the other levels, in the case of "hole" nodes, the correspondence is not direct (i.e., a hole node contains a set of hides nodes, each one having -analogously to shadow nodes- a correspondent at the other levels; see also Figure 6 in the following). In some sense, one could also see the grey "shadow" nodes at the CIG query level in Figure 5 as the nodes constituting a unique hole in the CIG.

4.1. GLARE-Edu CIG-tests acquisition module

As shown in the architecture in Fig. 4 above, the CIG-tests acquisition module takes as input a CIG, and a patient (case study), intended as the evolution of the status of one patient treated using the chosen CIG. Both the CIG and the patient are selected by the teacher(s) in GLARE-Edu libraries. Given such an input, our module supports the acquisition of a CIG-test by showing her/him the instantiation of the CIG corresponding to the selected patient. Specifically, a graphical interface shows the teacher the whole CIG, in which the path followed to treat the given patient is highlighted (see Fig. 6(a) below).

At this point, the teacher has the option to specify a *shadow* node or a *hole* node.[3] Notably, such an option is repeatedly offered to the teacher, until s/he decides that the CIG-test is complete.

- In case the teacher has chosen the "shadow" option, s/he can click on one highlighted node in the CIG (i.e., a node that is part of the CIG's path used to manage the patient). Such a node will appear as grey in the teacher's interface (Fig. 6(b)), and in the CIG-test shown to the student (Fig. 6(c)), and the student will have to manage it, as discussed in Section 5.

[3] Technically speaking, in this paper we propose two different formal languages:

- A language to model CIGs (used by both GLARE and GLARE-Edu), which provides seven action-types (work actions, pharmacological prescriptions, query actions, patient-status-based decisions, cost-benefit decisions, conclusions, and composite actions);
- A language to model tests (used by GLARE-Edu only) which includes two additional entities, "shadow" and "hole" (with the purpose of formulating "focused" queries).

  Of course, "holes" can be said to be "composite", in the sense that they represent a subgraph, but just in the language to model tests.

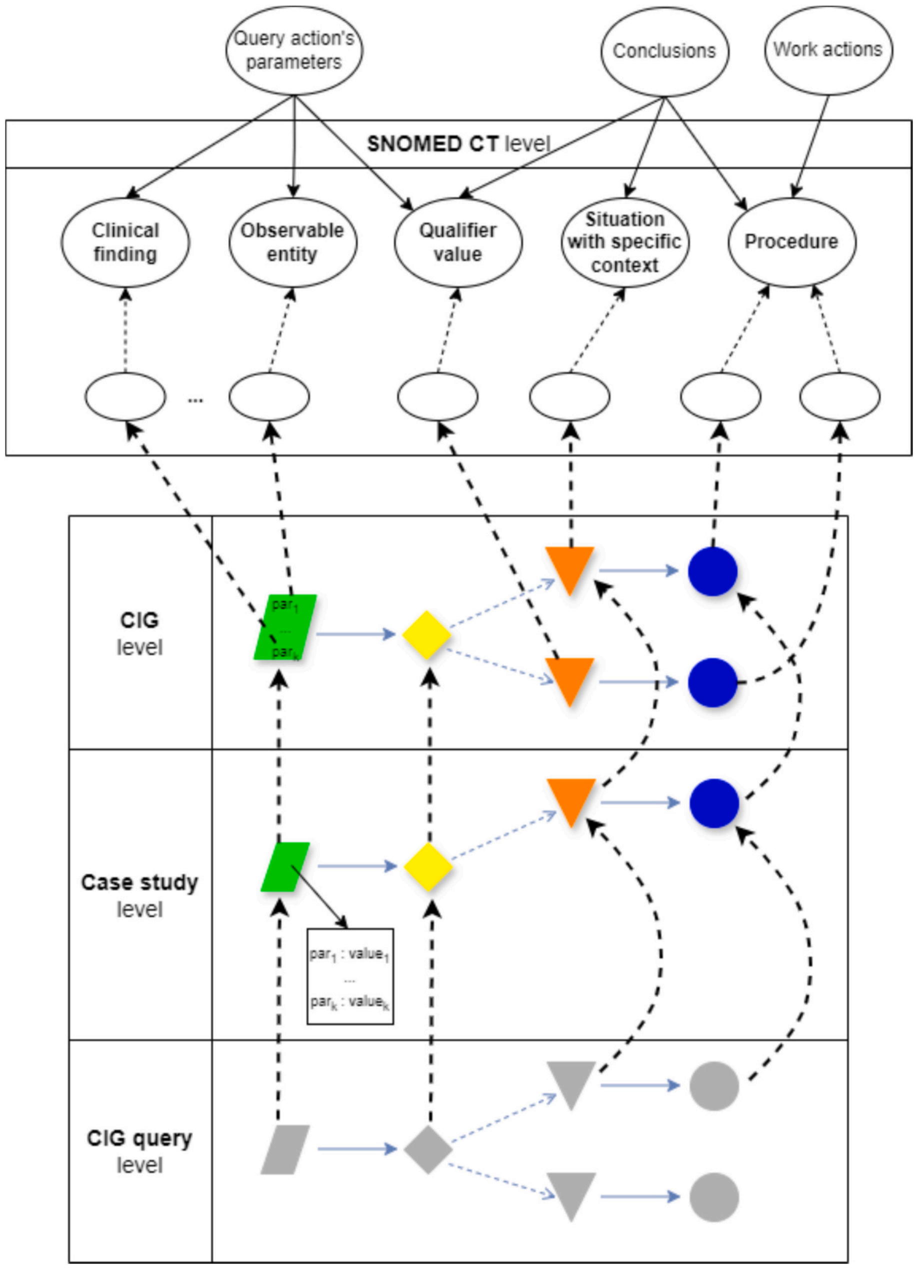


Fig. 5. Relationship between data/knowledge structures of GLARE-Edu. Legenda: at CIG, Case Study and CIG Query levels, parallelograms represent query actions, rhombuses represent decisions, triangles represent conclusions and circles represent work actions. Ovals at the SNOMED CT level represent the corresponding concepts in SNOMED CT ontology.

- In case the teacher has chosen the “hole” option, our graphical interface supports her/him in the selection of the part of the CIG which will be “hidden” by the hole node. First, the starting node of the hole must be selected by clicking on a node in the highlighted path. Then, at least an ending node (that must be chosen among highlighted nodes) must be selected by the teacher. Moreover, the teacher can select other nodes reachable from the start node, and not highlighted, to “put in the hole” also parts of the CIG that represent actions not appropriate for the given patient. All the nodes obtained by navigating the CIG from the start node to the ending nodes become parts of the hole. Notably, to make the part of the CIG not included in the hole “meaningful” for students, our facility forces the constraint that, whenever a decision node is part of a hole, also all its conclusions are automatically part of it.

For example, Fig. 6(a) shows a part of the CIG of melanoma (already shown in Fig. 2 in Section 2.2) in which a patient’s path has been highlighted, and Fig. 6(b) shows the teacher view of the CIG in case the teacher has selected “ABCDE” as shadow nodes, “Risk of aesthetic or functional damage” as starting point of a hole, and “Excision of pigmented skin lesion” and “Incisional biopsy of skin lesion” as its ending points, and Fig. 6(c) illustrates how the CIG-test is shown to students.

5. CIG-test execution: Dealing with shadow nodes

The CIG-test execution module takes as input a CIG, a patient evolution (case study), an instantiation of the CIG for that patient, and a CIG-test. It shows the student the CIG-test, and lets her/him select a shadow or hole node in it (the test is ended when all shadow and hole nodes have been managed by the student).

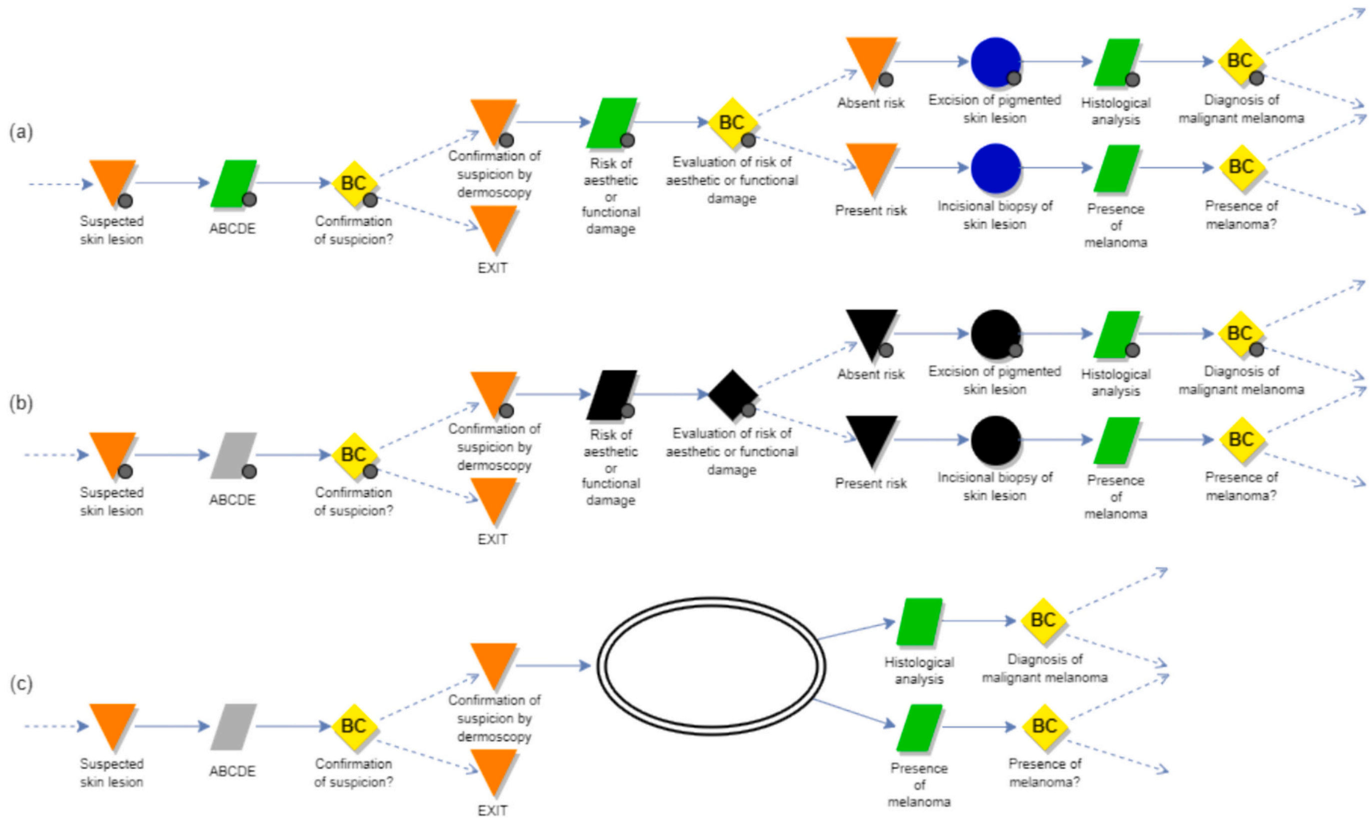


Fig. 6. Part of the melanoma CIG. In part (a) of the figure the path followed to treat the given patient is highlighted with dots in the figure; part (b) shows how the CIG is shown to the teacher after her selection of a shadow node (grey node "ABCDE") and of a hole (black nodes); part (c) shows how the CIG-test defined so far appears to students.

As soon as the student selects a shadow/hole node, the module shows the student the status of the patient before the execution of the node. Notably, such a status can be easily retrieved, since it is associated with the instance of the last Query node preceding the shadow/hole node.

Then, the student is required to "complete" the shadow or hole node, and the student's solutions are checked and explained by the CIG-test execution module, as dis-cussed below. In this Section, we focus on shadow nodes (we generalize to hole nodes in the Section 6). Different operations are performed by the CIG-test execution module, depending on the action-type of the shadow node. Notably, the action-type of the shadow node is shown to the student (so that s/he knows that s/he has to deal with a work action, or a pharmacological prescription, or a patient-status-based decision, etc.).

5.1. Action-type: Work action

When the shadow action is a *work action*, the hidden piece of information is what is the action to be performed. Our testing facility supports students' search of the action in the SNOMED CT ontology, in the hierarchy rooted by "Procedure" concept. Then, our approach automatically checks the conformance between the student's selection and the CIG "golden" work action (corresponding to the shadow node in the test). In the case of non-conformance, the testing facility provides a basic explanation of the non-conformance, in terms of SNOMED CT hierarchies. In fact, the explanation provided is the path(s) (behind the "Procedure" SNOMED CT concept) connecting the student's choice to the SNOMED CT node referenced by the CIG node, to show her/him the "ontological distance" between her/his choice and the "golden standard" in the CIG.

For example, Fig. 7 shows the case in which the shadow node in the test regards the SNOMED CT action "Incisional biopsy of skin lesion" (belonging to the melanoma CPG), and the choice of the student has been "Excision of lesion of skin". Notably, the whole path shown in Fig. 7 is provided as output as an explanation to the student, to show her/him the "ontological distance" between her/his choice and the "golden standard" in the CIG.

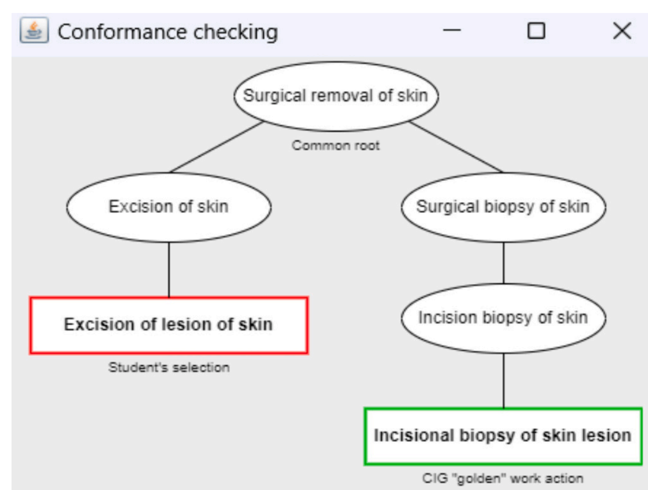


Fig. 7. Conformance checking for the SNOMED CT action "Incisional biopsy of skin lesion".

5.2. Action-type: Pharmacological prescription

When the shadow action is a *pharmacological prescription*, our testing facility hides:

i. What drug (active pharmacological ingredient) has to be administered
ii. What is the posology

To determine the drug, we support students' search in the SNOMED CT ontology, in the hierarchy rooted by the "Substance" node.

Conformance check and explanation concerning the drug are performed automatically, as described in Subsection 5.1 about concerning work actions. On the other hand, as regards posology, different components have to be considered:

1. Time pattern (e.g., twice a day)
2. Dosage form (e.g., 50 mg oral capsule)
3. Total number of repetitions (e.g., 10) and/or repetition condition (a Boolean formula; e.g., temperature value >38 AND blood_pressure value >140)

Time pattern and dosage form can be searched by the student behind the "Time patterns" and "Pharmaceutical/biologic product" concepts in the SNOMED CT ontology respectively. On the other hand, the number of repetitions and the repetition condition cannot be directly retrieved in SNOMED CT. Therefore, our facility directly asks them to the student (a specific interface is used to facilitate the acquisition of Boolean formulae acting as repetition conditions, see below in Section 5.4).

Conformance check and explanation concerning issues (1) and (2) are performed automatically, as described in subsection 5.1 about work actions. Concerning the comparison between Boolean formulae, see the discussion in Section 5.4 below.

5.3. Action-type: Query action

GLARE's query actions model requests of the ***parameter*** values needed to carry on the diagnosis/treatment of the patient. In query actions, we homogeneously model both anamnestic data, findings that can be obtained from a physician's visit of the patient, and data that need a laboratory test. In GLARE-Edu, we hide to students what are the requested parameters.

To select the parameters, we support search in the SNOMED CT ontology in the hierarchies rooted by the "Clinical finding", "Observable entity" and "Qualifier value" concepts.

Conformance check and explanation are simply performed by comparing the set of parameters proposed by the student and the one in the CIG. If there are differences, they are divided into two categories: *extra-parameters* (i.e., parameters requested by the student, but not required by the CIG), and *missing-parameters* (i.e., parameters not requested by the student, but requested by the CIG).

5.4. Action-type: Patient-status-based decision

In the case of patient-status-based decisions, two types of tests (of different complexity) are provided:

(1) Show the student the decision criteria associated with each one of the conclusions stemming from the shadow decision node, and asking her/him the conclusion s/he suggests for the given patient
(2) Do not show the student the decision criteria, and ask her/him both the conclusion s/he suggests, and the decision criteria associated with such a conclusion.

The management of case (1) above is quite easy. No search has to be done in the ontology (since the student has to choose among the conclusions shown in the CIG after the shadow node). Then, the conclusion chosen by the student is compared to the correct one (obtained by automatically applying the decision criteria to the representation of the status of the patient). If they are not the same, the correct conclusion is shown to the student.

Case (2) above is more complex. While the student's conclusion can be checked as above, there is also the need to acquire and check the condition proposed by the student.

The facility acquires the student's condition, through a graphical interface. Notably,

- The acquisition is facilitated by the fact that we only consider the *disjunctive normal form* (so that one or more sets of conjuncts are asked the student)
- The acquisition interface let the student choose only among the parameters required for the given decision (taken from the Query node preceding the decision node).

The check of conformance and explanation for the Boolean condition requires some technical discussion. In most cases, a single conjunction of atomic conditions suffices to model the condition associated with a given conclusion. However, in difficult cases conditions may involve (the disjunction of) more than one conjunction. In such cases, to check for the possible differences, we have first to determine which conjunct in the student's answer corresponds to which conjunct in the CIG's condition. We operate in two steps. First of all, we consider only the parameters (neglecting the inequalities parts of the atomic conditions), and we find the parameter correspondence that is more favorable for the student, that is, the correspondence that minimizes the number of differences. Once fixed the correspondence, for each conjunct, we first show the student the parameters that s/he should add (since they are present in the CIG but not in the student's solution), and the parameters that s/he should remove (vice versa). Then, considering the parameters present both in the student conjunction and in the corresponding one in the CIG, our tool analyses the related inequalities, and shows differences (if any) to the student.

5.5. Action-type: Cost-benefit decision

The treatment of cost-benefit decision is quite simple, since all the alternatives to be discriminated are clinically correct/equivalent. As a consequence, any choice of the student is a correct one. For such decisions, optionally, the teacher may decide to ask students to propose their own qualitative specification of the evaluation of the different criteria, for each alternative (e.g., to provide their own qualitative decision table, similar to the one shown in Fig. 1). Then, the students' table is compared with the CIG's one, and differences are shown to students.

5.6. Action-type: Conclusion

Though in GLARE formalism conclusion nodes may appear in any part of a CIG, in our current educational approach we focus only on conclusions stemming from decision nodes. For such conclusions, the student can inspect the condition, and has to search in the ontology the corresponding conclusion.

Notice that in GLARE different types of conclusions can be reached, depending on the specific decision:

a) Diagnostic conclusions (e.g., presence of melanoma)
b) Staging (e.g., cN0 for melanoma N-staging)
c) Therapeutic conclusions (e.g., surgical treatment)
d) EXIT

In the SNOMED CT ontology, (a) are rooted by the "Situation with specific context" concept, (b) by the "Qualifier value" node, and (c) by "Procedure" node. EXIT is a special type of conclusion, used to represent cases in which the CIG execution has to be ended (e.g., when the current state of the patient is out of the scope of the CIG).

As a consequence, we let the student choose concepts for conclusions from such three parts of the ontology or the EXIT node. In case the conclusion selected by the student is different from the CIG's one, two cases are possible. If the student has chosen a conclusion from a wrong sub-hierarchy (or the EXIT node, when it wasn't required by the CIG), such a fact is signaled to the student, and the CIG's conclusion is shown her/him. On the other hand, if the student has chosen the conclusion in the correct sub-hierarchy, the path(s) connecting the two conclusions in the SNOMED CT ontology is shown to the student, as described in Subsection 5.1 about work actions.

5.7. Action-type: Composite action

When the shadow node corresponds to a composite action, our verification facility operates recursively, by showing students the *shadow* template of the subgraph defining the composite actions (i.e., all the nodes in the subgraph are shadow nodes), and then operating as above.

5.8. Summary

Table 1 in the following briefly schematizes and summarizes the main issues we discussed in this section.

6. CIG-test execution: Dealing with hole nodes

Also in the case of hole nodes, our module shows the student the status of the patient when entering in the "hole part" of the CIG-test.

Table 1
A summary of GLARE-Edu management of shadow nodes (corresponding to atomic actions). The second column lists the pieces of information asked to students, the third column indicates the root concepts used for searching in SNOMED CT ontology (when applicable), and the fourth one sketches the method used to propose explanations. In the case of multiple pieces of information, indexes are added to link corresponding entities.

Action type	Hidden info (focus)	SNOMED CT concept (s)	Explanation type
Work action	Action?	"Procedure"	Path in SNOMED CT ontology
Pharmacological prescription	(1) Drug? (2) Time pattern? (3) Dosage form? (4) Repetitions?	(1) "Substance" (2) "Time Pattern" (3) "Pharmaceutical/ biologic product"	(1) Path in SNOMED CT ontology (2) Path in SNOMED CT ontology (3) Path in SNOMED CT ontology (4) Compare Boolean formulae
Query action	Parameters?	"Clinical finding", "Observable entity", "Qualifier value"	Compare sets of concepts
Patient-status-based decision (version 1)	Conclusion?		Compare concepts
Patient-status-based decision (version 2)	(1) Decision criteria? (2) Conclusion?		(1) Compare Boolean formulae (2) Compare concepts
Cost-benefit decision	Qualitative decision table?		Compare qualitative decision tables
Conclusion	Conclusions?	"Situation with specific context", "Qualifier value", "Procedure"	Path(s) in SNOMED CT ontology

Then, the CIG-test execution module navigates the part of the CIG hidden by the hole node, beginning from its first node. It asks the student, node by node, how to act on the patient. For each node:

(1) The action-type of the node (e.g., work action, pharmacological prescription, …) is asked to the student. If the student does not choose the correct action-type, a warning is shown to the student, indicating the correct type.
(2) Given the correct action-type, the CIG-test execution module operates (on the current node) as discussed in Section 5 above.

However, we adopt a slightly different approach (concerning the isolated treatment of a shadow node) when a **decision** is navigated. At that point, instead of separately applying the execution of decisions and then the execution of conclusions following the approach in Section 5, the module treats a ***decision and its conclusions together*** (notably, the CIG-tests acquisition module forces the constraints that, if a decision is in a hole, also its conclusions are in it). In such a case, the CIG-test execution module operates in three steps.

Step 1. All the possible conclusions of the decision are asked to the student. The module supports students in the SNOMED CT ontology, under the "Situation with specific context", "Qualifier value" and "Procedure" concepts, plus EXIT, and supports multiple selections. Then, conclusions selected by the student are compared to the ones in the CIG, and the student is shown:

- The conclusions s/he proposed that are not CIG's conclusions
- The CIG's conclusions not selected by the student.

Step 2. The module asks the student to select the proper conclusion for the given patient, and, in the case of patient-status-based decisions, it checks the answer, and explains discrepancies (see Section 5.4 above).

Step 3. If required by the teacher when defining the hole, it asks to the student the condition associated with the correct conclusion, and checks it (see Sections 5.4 and 5.5 above).

Then, the CIG-test execution module goes on in the navigation, starting from the correct conclusion, and neglecting the paths in the hole stemming from the wrong conclusions.

7. Experimental evaluation

In a very recent paper [46], we have described:

1) A preliminary evaluation of our overall educational approach, and
2) Our plan for an extensive experiment, to be run in the last six months of the PTPC-AI project.

For the sake of completeness, in this Section we briefly resume the main results about (1), and we sketch (2) in the concluding Section, considering future work.

The preliminary evaluation in [46] regarded GLARE-Edu in general, considering, besides a "basic" demo version of the testing facility described in this paper, also GLARE-Edu's navigation and execution facilities. In cooperation with expert physicians of the Alessandria Hospital, we acquired the Italian melanoma guideline [47] as well as several case studies. We then organized a two-hour lesson for the continuous education of 20 General Practitioners (GPs) of Alessandria territory, split into two homogeneous classes, one using GLARE-Edu and a control class (having a "standard" lesson).

At the end of the lesson, both classes spent the last 30 min completing evaluation questionnaires regarding different case studies, including different classes of questions regarding melanoma case studies. Different answers were proposed, and the physicians were asked to choose one of them. Overall, the percentage of correct answers was 44.55 % for the control group and 52.63 % for the GLARE-Edu group. Notably, as concerns questions focusing on melanoma N staging (a complex medical decision, out of the standard competences of GPs), the percentage of correct answers for GLARE-Edu group was 22,2 %, almost the double than for the control group (12,5 %).

The *p*-value (evaluated through Fisher's exact test) for the comprehensive test was 0.438. However, we believe that the low statistical significance is mostly due to (i) the limited number of physicians (and of questions) and (ii) the very short time we could devote to education.

We also asked the GLARE-Edu class some questions to ascertain the physicians' perception of the usefulness of GLARE-Edu. Notably, in a scale from 1 ("minimum") to 5 ("maximum"), the question "Would you recommend the use of GLARE-Edu?" obtained the score 4.11.

Questionnaires and results are available on the webpage Materials in the website of GLARE-Edu project (https://sites.google.com/uniupo.it/glareedu/materials), as well as the graphical representation of the melanoma CIG provided by GLARE-Edu.

8. Conclusions, related and future work

Medical education is a challenging and difficult task. CIG-based approaches can greatly contribute, especially to teach how to apply best medical procedures to specific patients. In this paper, we briefly present the background concerning GLARE, GLARE-Edu and the PTPC-AI approach to medical education, and then we focus on new facilities, supporting students' testing about how to act on a specific patient on the basis of the recommendation of a computer-interpretable clinical guideline. The main goal addressed by the approach described in this paper is to exploit CIG for students' testing about how to act on case studies, and the main original contributions are:

(i) Providing the possibility to develop tests focusing on specific subparts of CIGs, through the management of "shadow" and "hole" nodes, and
(ii) Balancing the level of freedom provided to students (to answer questions) and the feasibility of checking correctness and of providing knowledge-based explanations through the adoption of medical ontologies.

To the best of our knowledge, no other CIG-based approach in the literature provides the above contributions.

8.1. Related work

Considering related work, AI methodologies have been widely adopted for medical education. For example, already in the 1970s, a new version of MYCIN expert system was developed to teach medical diagnosis [48,49]. One general limitation of expert systems (not only in the educational context) is the difficulty of acquiring knowledge from experts. On the other hand, such a problem does not arise in the area of CPGs, due to the wide availability of thousands of good-quality CPGs, designed and certified by national and international teams of experts. Indeed, CPGs encode (mostly in textual form) the best evidence-based procedures to manage patients (affected by a specific disease), and can thus be used as valuable tools to teach students how to manage patients. Indeed, many CIG systems have been proposed by the AIM literature. While they are mostly devoted to decision support, some of them have already been used (or can be easily used) for medical education, or have been specifically extended to support explanations (consider, e.g., [50]). However, the educational task involves specific challenges (with respect to decision support). If we restrict our attention to CIG systems specifically designed for education, the range of approaches becomes quite restricted. To the best of our knowledge, the first proposal of a CIG architecture explicitly supporting education dates back to Boxwala et al.'s work is reported in [25]. Boxwala et al. proposed an architecture for a guideline execution engine that utilizes guidelines encoded in GLIF [51], intended for use in a variety of applications. The engine traverses the guideline by evaluating logic conditions specified in the guideline against patient data values. The paper briefly described two applications, and mentioned that an educational application had also been developed, without proposing any additional description.

In the last ten years, there are some CIG approaches specifically devoted to education. Notably, most of them are not domain independent (like GLARE-Edu), but are geared to cover a specific disease or medical problem. One of them has been proposed by Real et al. [27], and focuses on *arterial hypertension*. The proposed approach adopts decision tables to model the diagnostic knowledge in some guidelines about the most frequent causes of *arterial hypertension*. Such tables have then been used to train residents at the Hospital Clínic de Barcelona. The residents' decisions at the different diagnostic steps were compared with the decisions provided by the decision tables, to analyze the progressive adaptation of clinicians to the recommendations in the guideline. In [28], the approach focuses on *seven types of shock*. They propose Shock-Instructor, a web-based e-learning tool that uses a knowledge base about the evidence found in the CPG, with the goal of facilitating hospital residents in learning the correct application of the CPG, and to improve their skills. A significant part of the work consists not only in the acquisition of 137 clinical rules from a CPG, but also in the definition of 51 case studies. Finally, the paper also reports a successful application of Shock-Instructor to a group of 33 residents of a tertiary hospital. Uribe-Ocampo et al. [29] focus their approach to the context of *antenatal health-care environment*. They propose a *serious game* called SIM-CIG. They model (part of) the knowledge in obstetric CPGs in rules with preconditions, goals and actions. In the videogame, they support the simulation of the treatment of virtual patients, and evaluate the proposals of the players on the basis of CIGs relative to such a knowledge, using preconditions (encoded by Boolean formulae) to evaluate conformance. Notably, like the above approaches, GLARE-Edu is CIG-based and is specifically geared towards education. However, differently from them, GLARE-Edu is general, intended as CIG independent.

[26,52] propose four guideline-driven interactive case simulation tools (ICST) for the HIV Clinical Education Initiative (CEI), to deal with: (1) insomnia screening and treatment for HIV patients, (2) mental health screening for HIV patients, (3) HIV post-exposure prophylaxis (PEP) following occupational exposure in healthcare workers, and (4) HIV PEP following sexual assault. All these tools are based on the related CPGs developed by the NYS DOH AI. The system architecture of the ICSTs is based on an enhancement of the GLIF/GLEE framework [8], and is domain-independent. These ICSTs provide several educational facilities. The most interesting of them is the "interactive decision diagram". In this facility, a clinician user can review the process of decision making on pre-defined case studies; he/she can also explore his/her own patient cases through interactions with the system checking the decision process step by step, and reviewing the different decision options. The ICST approach is the closest to GLARE-Edu in the literature, since ICSTs also support students' testing. However, contributions (i) and (ii) above are wholly original of our approach. Indeed, we are not aware of any contribution in the literature facing testing through hidden parts in the CIGs, and adopting medical ontologies to provide students with relevant "possible solutions", and to check and explain students' proposals.

Related work also includes critiquing systems. The main feature of critiquing systems is that the user of the system provides as input (i) a problem description (e.g., patient symptoms), and (ii) a specific solution (e.g., the treatment of the patient). This second input distinguishes critiquing systems from traditional expert systems, which only take a problem description as input [53]. Since the 80s', critiquing systems have a strong tradition in AI in general [53], and in AI in medicine in particular [54–56]. For instance, ESSENTIAL-ATTENDING [54] is a system designed to assist in the implementation of expert systems which critique a physician's plan for patient care, with applications in areas of medical management, patient workup, and differential diagnosis. Besides supporting (medical) experts with a knowledge-based second opinion, critiquing systems can also support education. For instance, the constructivist learning theory views learning as a process in which learners actively construct knowledge and understanding (e.g., [57]). Individualized feedback has been recognized as an important element in fostering this process [58], and critiquing systems have been used in order to provide such a feedback in an automated way (e.g., [59]). Two common approaches to critiquing are comparative critiquing and analytic critiquing. Comparative critiquing compares user's work with a presumably better solution in the system and points out differences between them. CIG represent, in computer format, the "golden standard" to manage patients affected by a specific disease. As a consequence, CIGs constitute a natural input for *comparative* critiquing systems. Two outmost examples of CIG-based critiquing systems are Asgaard [3] and the approach in [60]. In the seminal Asgaard project, authors propose a computational model for dynamic (runtime) guideline execution, plan-recognition, and critiquing tasks.

Specifically, the authors propose critiquing computational method in which the ideal actions are given by a formal model of a clinical guideline (expressed using ASBRU [61]), and where the actual actions are derived from patient data. A main feature of ASBRU is the fact that it supports the modeling of the *intentions* (represented by temporal patterns to be achieved or avoided) of the actions/plans in a guideline. The critiquing mechanism considers intentions [62]. For instance, a provider might not even follow the overall plan, but still adhere to a higher-level intention. While [61] focuses on the representation language and on the CIG execution mechanism (called *interpreter*), in the paper critiquing is mostly discussed through examples, and its possible application to education is not discussed, so that an in-depth comparison with our approach is not possible. For sure, the input of Miksch et al.'s critiquing approach is the whole treatment of a patient, while we aim at a more education-oriented approach in which (i) specific issues (shadow nodes) or parts (hole nodes) of the treatment can be considered, and students have to input the treatment step-by-step, considering concepts from a general medical ontology (SNOMED CT). On the other hand, considering also actions' intentions is an intriguing possibility, that we may consider in our future work (see below).

A general and formal approach to CIG-based critiquing has been proposed by Groot et al. [60], that show how critiquing can be cast in terms of *temporal logic*, and can be achieved by using *model checking*. Groot et al. consider clinical guidelines expressed in ASBRU language, and translate the ASBRU model to a *state transition system*. Model checking is used as the method to generate critiques. In Groot et al.'s proposal, model checking takes as input a state transition system modeling a given guideline, and a logical specification modeling the actions performed on the patient, and investigates the consistency between the formalized guideline and the treatment administered to the patient. Different forms of non-compliance (between the "ideal" and the "real" treatment) are identified. In the "direct critiquing" approach, non-compliances are checked through formulae expressed in the CTL temporal logic. An alternative critiquing approach, based on *satisfaction sets*, is also proposed. Such an approach uses a model checker to determine all the possible patient findings for which a certain property holds in the formal model of the clinical guideline. This compiled information from the guideline is then used for critiquing the patient data. Overall, the main goal of Groot et al.'s approach is to propose a strong theoretic-grounded approach demonstrating the feasibility of CIG-based critiquing based on temporal logics and model-checking. While they certainly achieved their goal, such a goal is quite far from our (educational) ones. Indeed, though theoretically appealing, we doubt that notions like *temporal logics*, *satisfaction sets* and *model checking* are the most appropriate ones to teach medical students how to apply clinical guidelines to patients.

Notably, the approach in this paper substantially extends also our previous educational work. We started our research in the educational area in 2019, proposing a roadmap for the definition of GLARE-Edu [30]. In [63] we have described the AI-LEAP project, and in [31] the PTPC-AI one. Very recently, in our JMS paper [46], we proposed a *non-technical* description of our overall educational approach. We

i. Motivated the need for educational CIG systems like GLARE-Edu (concerning existent CIG systems for decision support),
ii. Presented GLARE-Edu underlying "philosophical" principles,
iii. Discussed the main tasks of GLARE-Edu and its educational impact,
iv. Detailed our preliminary experimental evaluation of GLARE-Edu and the plan for an extensive evaluation, and
v. Proposed an extensive section of related works, settling our approach in the context of AI-based medical education (with specific emphasis on the com-parison between approaches using "symbolic" AI -like GLARE-Edu- and approaches based on Machine Learning techniques).

On the other hand, in the JMS paper, we just proposed a brief high-level description of the overall architecture of GLARE-Edu (in the figure of the architecture, we just used one box to include the overall verification/testing formalism) and of its components, in which only a subsection of <200 words was devoted to the description of the verification mechanisms, and the notions of "shadow" and "hole" nodes, as well as the interaction with medical ontologies, were not even mentioned.

In particular, the architectural extensions discussed in Section 4, and the CIG-test execution mechanism presented in Sections 5 and 6 are entirely new.

8.2. Generality of GLARE-Edu's approach

Before considering limitations and future work, it is worth analyzing the generality of our approach. At least two different dimensions can be considered: independence of the domain, and independence of GLARE representation formalism.

8.2.1. Independence of the domain

As already discussed in this paper, GLARE and GLARE-Edu are domain-independent. They can operate on any clinical guideline acquired through GLARE's acquisition module (i.e., on any CIG represented using GLARE's formalism).

8.2.2. Independence of GLARE's formalism

Indeed, most CIG systems are based on a specific representation formalism. Two relevant exceptions are Degel [5] and META-GLARE [39]. In particular, META-GLARE is parametric concerning the formalism used to represent CIGs: it takes as input a representation formalism (through a dedicated formalism acquisition tool), and provides a general tool for acquiring CIGs and one to execute them which operate on any formalism (acquired through the formalism acquisition tool).

Overall, GLARE-Edu is formalism-dependent, since it works on CIG represented in GLARE's formalism. However, there are several aspects of the approach discussed in this paper that have a high-degree of independence from GLARE's specific formalism:

- The methodology we propose to address the tasks (i) and (ii) (i.e., the introduction and management of shadow nodes and hole nodes, and the use of a medical ontology like SNOMED CT) is GLARE-independent, and could be instantiated into other CIG-based systems
- The overall architecture we propose (see Fig. 4) is GLARE-independent
- The mechanisms to navigate/query SNOMED CT ontology are GLARE-independent

On the other hand, the treatment of shadow nodes (and of hole nodes) depends on GLARE's representation formalism. Notice, however, that, as shown by several surveys, the Task Network Model (TNM) formalisms for CIGs (like e.g., GLARE, ASBRU, Proforma and many others) in the AIM literature present many common aspects [16–23]. Thus, moving from GLARE to other CIG systems (based on a TNM representation) would not be difficult from the conceptual viewpoint.

8.3. Limitations of GLARE-Edu and future work

In our next work, we aim to address some limitations of our current approach.

First of all, as already discussed in Section 7, we only have a partial and preliminary evaluation of GLARE-Edu, but we have already planned an extended experimental evaluation. We will accomplish such a goal in the last six months of the PTPC-AI project, proposing to a class of medical students of our University a six-month course about the European guideline about dyslipidemia [64]. The plan of our experiment is described in detail in [46], where we also highlight that students' engagement for a University course involves also (at least in the Italian education system) a quite complex and time-demanding processes, such as the accreditation of the course itself (specifically, our course has been already accredited as an "Attività didattiche elettive" - ADE or elective educational activities). Of course, students will be split into two homogeneous classes, and we will run a-priori, *in-itinere* and *a-posteriori* tests, comparing the two classes. Notably, the PTPC-AI project mandates that our activities during the evaluation phase are conducted under the constant supervision of the Bruno Kessler Foundation (https://www.fbk.eu/en/), serving as external evaluator.

Second, while GLARE and GLARE-Edu support the possibility of specifying (as attributes of arcs) the temporal constraints between nodes, such pieces of information are not yet managed by the testing facility discussed in this paper. Indeed, GLARE adopts a temporal reasoning module [65], and its integration in GLARE-Edu to support students' testing is one of the goals of our future work.

Third, we plan to investigate the possibility of extending our approach to consider actions' *intentions*, along the lines proposed by Shahar's et al.'s seminal work [3]. Intentions are certainly an important factor in medical reasoning, so considering them in our educational approach could lead to a significant improvement. However, to do so, a systematic association of CIG's actions and their intention is needed, and such an association is not systematically provided by clinical guidelines (at least, this is the case in our 30-year experience in the area). Even worst, a case study [66] showed that not only intentions of the clinical guidelines/protocols are often implicit but the intentions reported by experts almost always differ, which makes it hard to accommodate in a formal model (though the work in [67] achieved some progress to overcome such a problem).

Fourth, while preparing the material for the six-month course, we have realized that, to run the course and to support the personalization of the educational process, a quite large number of case studies are needed. And, more important, the acquisition of case studies requires the cooperation of a knowledge engineer and a medical expert, and is quite time consuming. As future work, we plan to automatize the generation of case studies, on the basis of the results of the checks of conformance (and their relative explanations) for CIG-tests.

CRediT authorship contribution statement

Alessio Bottrighi: Writing – original draft, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Antonio Maconi:** Resources, Project administration. **Stefano Nera:** Software, Visualization. **Luca Piovesan:** Methodology, Formal analysis, Validation, Software, Conceptualization. **Erica Raina:** Writing – original draft, Visualization, Software, Methodology, Formal analysis, Conceptualization. **Paolo Terenziani:** Writing – original draft, Validation, Supervision, Project administration, Methodology, Funding acquisition, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Paolo Terenziani reports financial support was provided by Compagnia Fondazione San Paolo, Cassa Depositi e Prestiti. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is partially supported by Compagnia di San Paolo and Fondazione Cassa Depositi e Presiti, Bando Intelligenza Artificiale 2, AI-LEAP project. Erica Raina and Stefano Nera are PhD students enrolled in the National PhD program in Artificial Intelligence, XXXIX and XL cycle, course on Health and life sciences, organized by Università Campus Bio-Medico di Roma. The authors are very grateful to Dr. Federica Grosso, for her invaluable contribution in the acquisition of the melanoma guideline, and to Yuval Shahar, for inspiring insights and discussion about CIG-based medical education.

References

[1] Gaba DM. The future vision of simulation in health care. Qual Saf Health Care 2004;13(Suppl. 1):i2–10. https://doi.org/10.1136/qhc.13.suppl_1.i2.
[2] Field MJ, Lohr KN. Clinical practice guidelines: Directions for a new program. Washington (DC): National Academies Press (US); 1990.
[3] Shahar Y, Miksch S, Johnson P. The Asgaard project: a task-specific framework for the application and critiquing of time-oriented clinical guidelines. Artif Intell Med 1998;14:29–51. https://doi.org/10.1016/s0933-3657(98)00015-3.
[4] Musen MA, Tu SW, Das AK, Shahar Y. EON: a component-based approach to automation of protocol-directed therapy. J Am Med Inform Assoc JAMIA 1996;3:367–88. https://doi.org/10.1136/jamia.1996.97084511.
[5] Shahar Y, Young O, Shalom E, Galperin M, Mayaffit A, Moskovitch R, et al. A framework for a distributed, hybrid, multiple-ontology clinical-guideline library, and automated guideline-support tools. J Biomed Inform 2004;37:325–44. https://doi.org/10.1016/j.jbi.2004.07.001.
[6] Terenziani P, Molino G, Torchio M. A modular approach for representing and executing clinical guidelines. Artif Intell Med 2001;23:249–76.
[7] Terenziani P, Montani S, Bottrighi A, Molino G, Torchio M. Applying artificial intelligence to clinical guidelines: the GLARE approach. Stud Health Technol Inform 2008;139:273–82.
[8] Peleg M, Boxwala AA, Ogunyemi O, Zeng Q, Tu S, Lacson R, et al. GLIF3: the evolution of a guideline representation format. Proc AMIA Symp 2000:645–9.
[9] Bottrighi A, Chesani F, Mello P, Montali M, Montani S, Storari S, et al. Analysis of the GLARE and GPROVE Approaches to Clinical Guidelines. In: Riaño D, ten Teije A, Miksch S, Peleg M, editors. Knowl. Represent. Health-Care Data Process. Guidel. Berlin, Heidelberg: Springer; 2010. p. 76–87. https://doi.org/10.1007/978-3-642-11808-1_7.
[10] Quaglini S, Stefanelli M, Cavallini A, Micieli G, Fassino C, Mossa C. Guideline-based careflow systems. Artif Intell Med 2000;20:5–22. https://doi.org/10.1016/s0933-3657(00)00050-6.
[11] Johnson PD, Tu S, Booth N, Sugden B, Purves IN. Using scenarios in chronic disease management guidelines for primary care. Proc AMIA Symp 2000:389–93.
[12] Fox J, Johns N, Rahmanzadeh A. Disseminating medical knowledge: the PROforma approach. Artif Intell Med 1998;14:157–81. https://doi.org/10.1016/s0933-3657(98)00021-9.
[13] Sutton DR, Fox J. The syntax and semantics of the PROforma guideline modeling language. J Am Med Inform Assoc JAMIA 2003;10:433–43. https://doi.org/10.1197/jamia.M1264.
[14] Tu SW, Campbell JR, Glasgow J, Nyman MA, McClure R, McClay J, et al. The SAGE guideline model: achievements and overview. J Am Med Inform Assoc JAMIA 2007;14:589–98. https://doi.org/10.1197/jamia.M2399.
[15] Teije A ten, Miksch S, Lucas P. Computer-based medical guidelines and protocols: A primer and current trends. Amsterdam;Washington: DC: IOS Press; 2008.
[16] Heller BR. Computer-based support for clinical guidelines and protocols: proceedings of EWGLP 2000. Amsterdam ; Washington: DC: IOS Press; 2001.
[17] Wang D, Peleg M, Tu SW, Boxwala AA, Greenes RA, Patel VL, et al. Representation primitives, process models and patient data in computer-interpretable clinical practice guidelines: a literature review of guideline representation models. Int J Med Inform 2002;68:59–70. https://doi.org/10.1016/s1386-5056(02)00065-5.
[18] Peleg M, Tu S, Bury J, Ciccarese P, Fox J, Greenes RA, et al. Comparing computer-interpretable guideline models: a case-study approach. J Am Med Inform Assoc JAMIA 2003;10:52–68. https://doi.org/10.1197/jamia.m1135.
[19] Computer-based support for clinical guidelines and protocols: proceedings of the Symposium on Computerized Guidelines and Protocols (CGP 2004). In: Kaiser K, Miksch S, Evropské Centrum pro Medicínskou Informatiku, Statistiku a Epidemiologii, editors. held at the Novotek Hotel, Prague, in the Czech Republic, as part of the International Joint Meeting EuroMISE 2004. Amsterdam: IOS Press; 2004.
[20] Isern D, Moreno A. Computer-based execution of clinical guidelines: a review. Int J Med Inform 2008;77:787–808. https://doi.org/10.1016/j.ijmedinf.2008.05.010.

[21] Bottrighi A, Giordano L, Molino G, Montani S, Terenziani P, Torchio M. Adopting model checking techniques for clinical guidelines verification. Artif Intell Med 2010;48:1–19. https://doi.org/10.1016/j.artmed.2009.09.003.
[22] Grando MA, Glasspool D, Fox J. A formal approach to the analysis of clinical computer-interpretable guideline modeling languages. Artif Intell Med 2012;54:1–13. https://doi.org/10.1016/j.artmed.2011.07.001.
[23] Peleg M. Computer-interpretable clinical guidelines: a methodological review. J Biomed Inform 2013;46:744–63. https://doi.org/10.1016/j.jbi.2013.06.009.
[24] Van Woensel W, Tu SW, Michalowski W, Sibte Raza Abidi S, Abidi S, Alonso J-R, et al. A community-of-practice-based evaluation methodology for knowledge intensive computational methods and its application to multimorbidity decision support. J Biomed Inform 2023;142:104395. https://doi.org/10.1016/j.jbi.2023.104395.
[25] Boxwala AA, Greenes RA, Deibel SR. Architecture for a multipurpose guideline execution engine. Proc AMIA Symp 1999:701–5.
[26] Le XH, Luque AE, Wang D. Development of guideline-driven mobile applications for clinical education and decision support with customization to individual patient cases. In: AMIA 2012 Am Med Inform Assoc Annu Symp 2012; 2012. Chic. Ill. USA Novemb. 3–7 2012, AMIA.
[27] Real F, Riaño D, Alonso JR. Training Residents in the Application of Clinical Guidelines for Differential Diagnosis of the Most Frequent Causes of Arterial Hypertension with Decision Tables. In: Miksch S, Riaño D, Teije A ten, editors. Knowl. Represent. Health Care - 6th Int. Workshop KR4HC 2014 Held Part Vienna Summer Log. VSL 2014 Vienna Austria July 21 2014 Revis. Sel. Pap. 8903. Springer; 2014. p. 147–59. https://doi.org/10.1007/978-3-319-13281-5_11.
[28] Riaño D, Real F, Alonso JR. Improving resident's skills in the management of circulatory shock with a knowledge-based e-learning tool. Int J Med Inform 2018;113:49–55. https://doi.org/10.1016/j.ijmedinf.2018.02.006.
[29] Uribe-Ocampo S, Torres EA, Luna IF, Florez-Arango JF, Smith JW. SIM-CIG: a serious game to practice and improve clinical guidelines adoption based on computer-interpretable guidelines. Stud Health Technol Inform 2019;264:1997–8. https://doi.org/10.3233/SHTI190753.
[30] Bottrighi A, Molino G, Piovesan L, Terenziani P. Simulating clinical guidelines for Medical Education. In: Proc. 2019 4th Int. Conf. Inf. Educ. Innov. - ICIEI 2019. Durham, United Kingdom: ACM Press; 2019. p. 66–72. https://doi.org/10.1145/3345094.3345099.
[31] Bottrighi A, Girardi E, Maconi A, Nera S, Raina E, Roveta A, et al. Clinical guidelines acquisition, representation and simulation for medical education. In: 2023 17th Int Conf Signal-Image Technol Internet-Based Syst SITIS. 354; 2023. https://doi.org/10.1109/SITIS61268.2023.00065. 60.
[32] Terenziani P, Montani S, Bottrighi A, Torchio M, Molino G. Supporting physicians in taking decisions in clinical guidelines: the GLARE "what if" facility. Proc AMIA Symp 2002:772–6.
[33] Terenziani P, Carlini C, Montani S. Towards a comprehensive treatment of temporal constraints in clinical guidelines 2002;2002:20–7. https://doi.org/10.1109/TIME.2002.1027468.
[34] Giordano L, Terenziani P, Bottrighi A, Montani S, Donzella L. Model checking for clinical guidelines: an agent-based approach. AMIA Annu Symp Proc AMIA Symp 2006;2006:289–93.
[35] Leonardi G, Bottrighi A, Galliani G, Terenziani P, Messina A, Della Corte F. Exceptions handling within GLARE clinical guideline framework. AMIA Annu Symp Proc AMIA Symp 2012;2012:512–21.
[36] Piovesan L, Molino G, Terenziani P. Supporting multi-level user-driven detection of guideline interactions 2015:413–22. https://doi.org/10.5220/0005217404130422.
[37] Piovesan L, Terenziani P. A mixed-initiative approach to the conciliation of clinical guidelines for comorbid patients. In: Lect Notes Comput Sci Subser Lect Notes Artif Intell Lect Notes Bioinforma. 9485; 2015. p. 95–108.
[38] Piovesan L, Anselma L, Terenziani P. Temporal detection of guideline interactions 2015:40–50. https://doi.org/10.5220/0005186300400050.
[39] Bottrighi A, Terenziani P. META-GLARE: a meta-system for defining your own computer interpretable guideline system-architecture and acquisition. Artif Intell Med 2016;72:22–41. https://doi.org/10.1016/j.artmed.2016.07.002.
[40] Spiotta M, Terenziani P, Dupré DT. Temporal conformance analysis and explanation of clinical guidelines execution: an answer set programming approach. IEEE Trans Knowl Data Eng 2017;29:2567–80. https://doi.org/10.1109/TKDE.2017.2734084.
[41] Piovesan L, Terenziani P, Molino G. GLARE-SSCPM: an intelligent system to support the treatment of comorbid patients. IEEE Intell Syst 2018;33:37–46. https://doi.org/10.1109/MIS.2018.2886697.
[42] Piovesan L, Terenziani P, Theseider Dupré D. Conformance analysis for comorbid patients in answer set programming. J Biomed Inform 2020;103:103377. https://doi.org/10.1016/j.jbi.2020.103377.
[43] Andolina A, Guazzone M, Piovesan L, Terenziani P. Temporal reasoning and query answering with preferences and probabilities for medical decision support. Expert Syst Appl 2022:195. https://doi.org/10.1016/j.eswa.2022.116565.
[44] International Health Terminology Standards Development Organisation. SNOMED clinical terms n.d. http://www.ihtsdo.org/snomed-ct (accessed January 23, 2024).
[45] Bottrighi A, Nera S, Piovesan L, Raina E, Roveta A, Terenziani P, et al. AI-LEAP project, PTPC-AI subproject, deliverable D1.2 n.d. https://drive.google.com/file/d/1CzTyGlshpkphnVMM_VYYBdiDQ5dbk2sH/view?usp=sharing&usp=embed_facebook (accessed April 23, 2024).
[46] Bottrighi A, Grosso F, Ghiglione M, Maconi A, Nera S, Piovesan L, et al. A symbolic AI approach to medical training. J Med Syst 2025;49. https://doi.org/10.1007/s10916-024-02139-y.
[47] Associazione Italiana Oncologia Medica. LG AIOM melanoma n.d. https://www.aiom.it/wp-content/uploads/2020/10/2020_LG_AIOM_Melanoma.pdf (accessed November 20, 2023).
[48] Clancey WJ, Shortliffe EH, Buchanan BG. Intelligent computer-aided instruction for Medical diagnosis. Proc Annu Symp Comput Appl Med Care 1979;175(83).
[49] Clancey WJ, Letsinger R. NEOMYCIN: Reconfiguring a rule-based expert system for application to teaching. In: Proc. 7th Int. Jt. Conf. Artif. Intell. vol. 2. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.; 1981. p. 829–36.
[50] Woensel WV, Scioscia F, Loseto G, Seneviratne O, Patton E, Abidi S, et al. Explainable Clinical Decision Support: Towards Patient-Facing Explanations for Education and Long-Term Behavior Change. In: Lect Notes Comput Sci Subser Lect Notes Artif Intell Lect Notes Bioinforma. 13263. LNAI; 2022. p. 57–62. https://doi.org/10.1007/978-3-031-09342-5_6.
[51] Ohno-Machado L, Gennari JH, Murphy SN, Jain NL, Tu SW, Oliver DE, et al. The guideline interchange format: a model for representing guidelines. J Am Med Inform Assoc JAMIA 1998;5:357–72. https://doi.org/10.1136/jamia.1998.0050357.
[52] Le XH, Luque AE, Wang D. Assessing the usage of a guideline-driven interactive case simulation tool for insomnia screening and treatment in an HIV clinical education program. Stud Health Technol Inform 2013;192:323–7.
[53] Silverman BG. Survey of expert critiquing systems: practical and theoretical frontiers. Commun ACM 1992;35:106–27. https://doi.org/10.1145/129852.129861.
[54] Miller PL. Building an expert critiquing system: ESSENTIAL-ATTENDING. Methods Inf Med 1986;25:71–8. https://doi.org/10.1055/s-0038-1636679.
[55] Miller PL. Expert critiquing systems: Practice-based Medical consultation by computer. New York, NY: Springer New York; 1986.
[56] Medical applications of intelligent data analysis: Research advancements. Hershey: Information science reference; 2012.
[57] Bransford JD, Goldman SR, Vye NJ. Making a difference in people's abilities to think: reflections on a decade of work and some hopes for the future. Dir Dev Influ Child 1991:147–80.
[58] Council NR. How people learn: Brain, mind, experience, and school. Expanded ed. Washington, DC: The National Academies Press; 2000. https://doi.org/10.17226/9853.
[59] Qiu L, Riesbeck C. Facilitating critiquing in education: The design and implementation of the Java Critiquer. 2003.
[60] Groot P, Hommersom A, Lucas PJF, Merk R-J, ten Teije A, van Harmelen F, et al. Using model checking for critiquing based on clinical guidelines. Artif Intell Med 2009;46:19–36. https://doi.org/10.1016/j.artmed.2008.07.007.
[61] Miksch S, Shahar Y, Johnson P. Asbru: A Task-Specific, Intention-Based, and Time-Oriented Language for Representing Skeletal Plans. 1999.
[62] Advani A, Lo K, Shahar Y. Intention-based critiquing of guideline-oriented medical care. Proc AMIA Symp 1998:483–7.
[63] Baldoni M, Baroglio C, Bottrighi A, Bucciarelli M, Capecchi S, Gandolfi E, et al. Fostering Awareness and Personalization of Learning Artificial Intelligence 2023; 3486:449–54.
[64] 2021 ESC Guidelines on cardiovascular disease prevention in clinical practice n.d. https://www.escardio.org/Guidelines/Clinical-Practice-Guidelines/CVD-Prevention-Guidelines (accessed October 29, 2024).
[65] Anselma L, Terenziani P, Montani S, Bottrighi A. Towards a comprehensive treatment of repetitions, periodicity and temporal constraints in clinical guidelines. Artif Intell Med 2006;38:171–95. https://doi.org/10.1016/j.artmed.2006.03.007.
[66] Marcos M, Berger G, van Harmelen F, ten Teije A, Roomans H, Miksch S. Using critiquing for improving medical protocols: harder than it seems. Lect Notes Comput Sci Subser Lect Notes Artif Intell Lect Notes Bioinforma 2001;2101:431–41. https://doi.org/10.1007/3-540-48229-6_59.
[67] Sips R-J, Braun L. Roos N. ECAI Workshop AI Tech Healthc Evid-Based Guidel Protoc: Applying formal medical guidelines for critiquing; 2006.