



Architecture for a Multimodal and Domain-Independent Clinical Decision Support System Software Development Kit

Naiara Muro, Nekane Larburu, Jordi Torres, Jon Kerexeta, Garazi Artola,
Monica Arrue, Ivan Macia, Brigitte Seroussi

► To cite this version:

Naiara Muro, Nekane Larburu, Jordi Torres, Jon Kerexeta, Garazi Artola, et al.. Architecture for a Multimodal and Domain-Independent Clinical Decision Support System Software Development Kit. 2019 41st Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Jul 2019, Berlin, Germany. pp.1399-1404, 10.1109/EMBC.2019.8856459 . hal-03648936

HAL Id: hal-03648936

<https://hal.science/hal-03648936>

Submitted on 11 Apr 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Architecture for a Multimodal and Domain-Independent Clinical Decision Support System Software Development Kit

Naiara Muro^{a,b,c}, Nekane Larburu^{b,c}, Jordi Torres^{b,c}

^aSorbonne Universités, UPMC Univ Paris 06, INSERM,
Université Paris 13, Sorbonne
Paris Cité, UMR S 1142, LIMICS, Paris, France
nmuro@vicomtech.org, nlarburu@vicomtech.org,
jtortes@vicomtech.org,

Jon Kerexeta^{b,c}, Garazi Artola^{b,c}, Mónica Arrue^{b,c},
Iván Macía^{b,c}, Brigitte Séroussi^a

^bVicomtech, Donostia-San Sebastian, Spain

^cBiodonostia, Donostia-San Sebastian, Spain

jkerexeta@vicomtech.org, gartola@vicomtech.org,
imacia@vicomtech.org, brigitte.seroussi@aphp.fr

Abstract— Digitalization of the decision-making process in healthcare has been promoted to improve clinical performance and patient outcomes. The implementation of Clinical Practice Guidelines (CPGs) using Clinical Decision Support Systems (CDSSs) is widely developed in order to achieve this purpose within clinical information systems. Nevertheless, due to several factors such as (i) incompleteness of CPG clinical knowledge, (ii) out-of-date contents, or (iii) knowledge gaps for specific clinical situations, guideline-based CDSSs may not completely satisfy clinical needs. The proposed architecture aims to cope with guideline knowledge gaps and pitfalls by harmonizing different modalities of decision support (i.e. guideline-based CDSSs, experience-based CDSSs, and data mining-based CDSSs) and information sources (i.e. CPGs and patient data) to provide the most complete, personalized, and up-to-date propositions to manage patients. We have developed a decisional event structure to retrieve all the information related to the decision-making process. This structure allows the tracking, computation, and evaluation of all the decisions made over time based on patient clinical outcomes. Finally, different user-friendly and easy-to-use authoring tools have been implemented within the proposed architecture to integrate the role of clinicians in the whole process of knowledge generation and validation. A use case based on Breast Cancer management is presented to illustrate the performance of the implemented architecture.

Key words—Clinical Decision Support System, Ontology, Clinical Practice Guidelines, Computer Interpretable Guidelines, Decisional Event, Decision Trees, Patient Outcomes, Breast Cancer

I. INTRODUCTION

Clinical practice guidelines (CPGs) are defined as “systematically developed statements to assist practitioner and patient decisions about appropriate health care for specific clinical circumstances” [1]. Hence, the implementation of CPGs is intended to decrease clinical negligence and improve patient clinical outcomes [2]. Nevertheless, the adherence to CPGs is not as strong as expected since there are many barriers to be overcome by clinicians when they try to apply them such as (i) lack of awareness or familiarity, (ii) lack of agreement, self-efficacy, or outcome expectancy, or (iii) the inertia of previous practice or behavior [3]. To overcome this deal, the implementation of guideline-based clinical decision support

systems (CDSSs) has been promoted. To insure the improvement of clinical practices, CDSSs must (i) be integrated within the clinician workflow, (ii) provide recommendations automatically, (iii) be implemented in a computerized way, and (iv) give the correct decision support in a short time [4]. Nevertheless, even with the use of CDSSs, it is estimated that about 30%–40% of patients receive treatments that do not comply with guideline recommendations [5].

In this paper, we present a multimodal CDSS that combines a guideline-based CDSS with two other different kinds of CDSSs, i.e. experience-based and data mining-based CDSS, to cope with the limitations of strictly guideline-based CDSSs. We introduce a Decisional Event (DE) structure to overcome the issue of the lack of previous practice performance data and its evaluation. DEs store all the information related to the decision-making process in a processable way. Visual analytics are also explored to present information in a user-friendly way. Finally, authoring tools for the different CDSSs are presented in order to make clinicians active in the review and update of new clinical evidence.

II. CLINICAL KNOWLEDGE FORMALIZATION

This chapter defines the three main modules that will serve as the core of our architecture: the definition of a decisional event in a processable structure, the formalization module used to translate CPGs into computer-interpretable guidelines (CIGs) for their integration into a CDSS, and the module for the semantical validation of all the formalized knowledge using an ontology. The formalization given by these three modules will be used all along the implementation of the different decision support systems.

A. Decisional Event structure

Clinical diagnostic and follow-up decisions usually tend to focus on a single patient and lose the perspective based on previous similar clinical cases and their outcomes [6]. As described in our previous work [7], the Decisional Event (DE) intends to store all the information needed and generated during the decision-making process into a computer interpretable way. This structure is composed by (i) a set of the clinical parameters characterizing a given patient, (ii) a subset of the formalized CIG rules that fit the studied patient, (iii)

the set of recommendations generated by those rules, and (iv) the final decision made by the clinician. This final decision can be one of the recommendations provided by the system, thus, in compliance with guidelines, or not. If the decision made is not compliant, (v) criteria should be given to argue this final decision non-supported by the clinical evidence formalized in the guidelines. Besides, the (vi) treatment actually performed is also stored in the DE to compute the compliance of the decision made with the treatment actually administered to the patient. Finally, (vii) a set of patient outcomes is formalized (e.g. toxicities, relapse, survival, adverse events) for the evaluation of the performance of the received treatment.

This structure is used to compute not only the compliance of the clinical decision with guideline recommendations, but also to generate the new clinical knowledge produced when a non-compliant decision is made. The DE is considered as the core of the whole architecture since it formalizes all the needed knowledge in a processable way and provides a backbone structure to the rest of the modules.

B. CIG formalization module

The translation of guideline-based clinical knowledge into a computer interpretable format requires some domain knowledge and major implementation skills. Nevertheless, it is widely known that it facilitates the decision support as it is able to analyze patient-specific clinical information using latest available evidence in the shortest time [8]. CIGs can be formalized following the “Task-Network Models” (TNM), i.e. models that represent the dependency among actions structured as hierarchical networks which, when fulfilled in a satisfactory way, provide recommendations. Several proposals have been reported in the field aiming to cope with different clinical modelling challenges, such as GLIF [9], PROforma [10], or Asbru [11].

In our approach, an object called *Condition Triplet* was defined within the DE structure to replicate the TNM in Java. This object is composed of (i) a precedent condition, (ii) a binary operator (i.e. and, or), and (iii) a consequent condition. The precedent and consequent conditions are based on a *Condition object* that was also formalized. This *Condition object* stores (i) the name of the clinical variables to be evaluated, (ii) the mathematical operator (i.e. >, >=, =, <, <=), and (iii) the value or threshold imposed by the condition to be evaluated.



Fig. 1. Our CIG formalization concept example.

This approach allows the formalization of CPGs into rule-based CIGs that can be written down in any document-based format (e.g. .drl, .xml, .json), since the knowledge is already tipped over a java-based structure. Moreover, this formalization is domain-independent,

allowing the implementation of different CPGs from different domains using the same structure.

C. Ontology-based semantical validation

Semantic Web Technologies, such as ontologies, are often proposed in the healthcare domain to represent the clinical knowledge contained in CPGs in a standard and semantically interoperable way [12]. The use of annotation properties is promoted to add labels to the ontology classes and to link each concept with its definition in validated and available standard terminologies, such as SNOMED CT¹ or NCI Thesaurus². This guaranties the interoperability of the implemented knowledge with stable and unique codes for each biomedical concept.

In our approach, we have developed a Jena API³ based tool for interacting with any ontology formalized in the Resource Description Framework (RDF) language to be used along with the CIG formalization module. Focusing on the classes and the defined properties relating them within the model, we restricted the knowledge to be used in the CIG formalization module to the one defined in the ontology. This is meant to avoid including corrupted knowledge and to assure the semantic interoperability in the whole CIG definition. As it totally relies on the relationships defined in the ontology, it is domain independent and can be used to query any ontology and to search the possible values of any class within it.

III. MULTIMODAL CDSS

The main goal of a CDSS is to give support to clinicians in the decision-making process during a clinical case evaluation by providing latest evidence-based recommendations in the shortest time [4]. Nevertheless, due to the knowledge gaps embedded within CPGs and the difficulties of the formalization and implementation of CIGs, sometimes CDSSs do not accomplish their objective successfully [13]. We propose the implementation of a multimodal CDSS relying on the conjunction of three kinds of decision supports (i.e. guideline-based, experience-based and data mining-based) to help clinicians during the decision-making process.

A. Guideline -based CDSS

The guideline-based CDSS (GL-CDSS) provides the best patient-specific evidence-based recommendations formalized in CIGs. GL-CDSSs are developed in order to improve clinical care and change physicians' behavior for better patients' clinical outcomes. Nevertheless, CPGs become out of date, especially in rapidly evolving fields, and they should be reviewed and updated to include latest evidence [3]. This requires a hard task of re-formalization

¹ <http://www.snomed.org/>

² <https://ncit.nci.nih.gov/ncitbrowser/>

³ <https://jena.apache.org/>

of the whole new guidelines and their implementation, being a very time-consuming process.

To ease the achievement of those objectives, a domain independent GL-CDSS that leans on our java-based DE structure and an authoring tool have been implemented. They are composed of a rule engine, that triggers the CIG rules in a runtime production environment and of an authoring tool to ease the CPG updating process.

1) Rule Engine

A rule engine can be defined as any system that implements rules, which when applied to data result into an outcome. In our case, rules come from the formalized CIGs and inputted patient data are evaluated to return the treatment that best fits the clinical case. Following Drools⁴, we have built up a java-based rule engine with two main components (i) an automatic rule file generator in Drools Rule Language (.drl extension) leaning on our java-based *Rule* object (contained within the DE formalization) and (ii) a standalone runtime execution environment. The first module assures that our system is domain independent and will work for any CIG formalized following our java structure. To achieve these objectives, *Rule Templates*⁵ were used.

```
dialect "java"

template "RuleTemplate"

rule "@{row.rowNumber} - @{name}"
when
    @{object}@{conditional}
then
    recom.add("@{action}");
    debug(drools);
end

end template
```

Fig. 2. Screenshot of the drl template used for CIG formalization.

These templates write the pattern of the rules, and populate them after with concrete rules content, generating a .drl file from java objects specifications. In our approach, four attributes are specified in the *Rule* java object to be mapped with the template: *name* (i.e. the name of the rule), *object* (i.e. the patient to be analyzed), *condition* (i.e. the conditional part of the rule to be accomplished by the patient clinical data) and *action* (i.e. the recommendations defined by the rule). First, formalized CIGs are retrieved from a source (e.g. xml file, database) and stored into the *Rule* java object. This *Rule* object is transformed into a Java Map object that simplifies its complexity into four key attributes and their values to be mapped to the template, where each attribute will be substituted by its value in the Java Map object. Once all rules are mapped, the final .drl file is generated and is ready to be triggered by the standalone runtime execution environment. If, in any case, the rule base needs any update or modification, the new rule will be added

following the same procedure, extending the existing .drl file with the new reported knowledge in a dynamic way.

2) Authoring tool for CPG formalization

In order to provide clinicians with a tool to easily formalize, update, and maintain CIGs, an authoring tool mapped in the backend with our java *Rule object* was built. First, the conditions that compose the rule are defined. Different combo boxes are provided to the user that query the ontology depending on the selected value. This means that if in the first combo box where the variable name should be selected the clinician clicks on "Age" the condition value will be restricted to the possible values defined for that variable in the ontology (i.e. natural numbers). The condition operator will be also filtered depending on the studied variables, since categorical and numerical values cannot be evaluated the same way. Once the condition is fulfilled, a binary operator can be selected (i.e. AND, OR) for including more conditions or end building the rule by defining the recommendation.

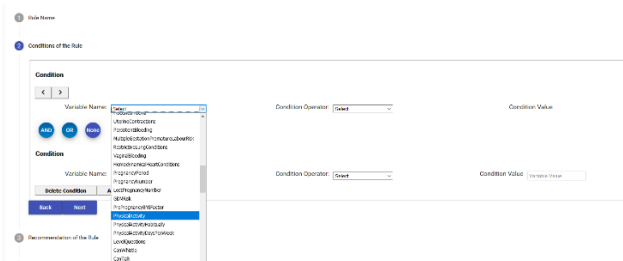


Fig. 3. Screenshot of the authoring tool GUI for the definition of new rules.

Modifications made by clinicians are automatically added into the knowledge base and used for updating the .drl to the latest version.

B. Experience-based CDSS

In some cases, the GL-based CDSS is insufficient to provide recommendations for particular clinical cases. To cope with those grey areas of CPGs, an Experience-based CDSS is proposed, which focuses on (i) the augmentation of CPGs with new experience-based rules and (ii) the assessment of the clinical evidence reported in the rules using patient-reported outcomes. Both tasks totally rely on the DE structure for the new knowledge computation.

1) Experience-based rules generation

The main objective of this module is to create new rules based on the experience of the clinicians that will cope with CPG non-compliant decisions. If the clinician considers that the proposed CPG-based recommendations are not adapted to the studied clinical case and he/she decides to not follow guidelines, the Experience-based CDSS will need the justification of this decision, i.e. non-compliance criteria. Then, the clinician decision is specified, and this information is stored. Using the triggered rules, the clinical data of the patient, the non-compliance criteria and the final decision, a new rule is built as explained in [7]. This new rule is integrated within the rule base of the GL-CDSS to be triggered in

⁴ <https://www.drools.org/>

⁵ https://docs.jboss.org/drools/release/7.6.0.Final/drools-docs/html_single/#_rule_templates

case of upcoming similar clinical cases. All the information generated in compliant and non-compliant cases is stored as new DEs to be exploited later when computing different measurements, such as the clinical evidence assessment.

2) Clinical evidence assessment

Different outcomes are analyzed to assess the impact of the treatment executed for a patient. In our previous work [15], we analyzed three different kinds of outcomes (i.e. treatment response, toxicities and adverse events, and patient-reported outcomes) for computing quality measurements, such as the usability and strength of the experience-based CDSS propositions, based on the past DE. These measurements are reflected in the rule defining that treatment as recommended. This allows not only to update the quality or strength of CPG-based recommendations but also to evaluate and validate the new experience-based propositions.

C. Data Mining-based CDSS

Purely rule-based CDSSs have some limitations since they require explicit knowledge definition of the studied clinical domain which is a very time-consuming task [16]. The data mining-based CDSS proposes to exploit the data directly to seek for new clinical knowledge that was not identified in guidelines, or to evaluate the already reported knowledge with evidence. Different machine learning techniques are proposed in the literature as supportive CDSS with good results [17]. One of the most used techniques is based on decision trees (DTs) [18], [19], since clinicians are quite used to this kind of data representation and evaluation.

1) Decision tree model generation

DTs are classification methods widely used for mining large datasets [20]. It relies on a hierarchical structure or tree that groups homogeneously data according to the variable to be predicted (e.g. predict if the patient will survive or not). The tree structure is made of nodes and depending on their position they will represent a recommendation (i.e. termination node) or a conditional part to be accomplished by the evaluated variables (i.e. non-termination nodes). The termination nodes define the stop criterion to abort the recursive partitioning into branches during the learning process of the model. Hence the model applies recursive partitioning until a stop criterion is reached. Our approach implements the most frequently used DTs (e.g. C4.5/J48, C5.0) using different toolkits for recursive partitioning from R [21], such as *partykit* [21], *rpart* [22], *C50* [23], and *R/Weka* [24]. As these models provide suitable results when supporting the reasoning process and replicate the reasoning workflow of a clinician, they have high acceptance rates among clinicians. Nevertheless, some clinicians require flexibility to modify or update the automatically generated DT [25]. To overcome this limitation, an authoring tool was built.

2) Authoring tool for DT modification

An authoring tool is meant to provide a tool allowing a user to edit or update content in an easy-to-use way. There are several software proposals for the creation and visualization of DTs, such as *Orange*⁶, *RapidMiner*⁷, *Weka*⁸ or *KNIME*⁹. These tools can generate different machine learning models, visualize them and test them. Nevertheless, they do not allow any modification once the model is generated, restricting the interaction with the end user. Other approaches give some tools for interacting with the user while creating the model [26], [27]. They visualize the dataset to be modelled to give the clinician a way to define more accurately the nodes of the tree. The main pitfall is that these visualizations are difficult to understand and hence, difficult to interact with for the end user. In our approach, the authoring tool allows editing DTs built from machine learning techniques for adding experts' knowledge. When new cases with non-modelled information are introduced into the system, our authoring tool allows the clinician to model this new knowledge and extend the model. Since it is mapped with the *Rule* java-object in the backend (Authoring tool for CPG formalization alike), the node definition of the authoring tool is equal to the condition definition. Hence, this model could easily be translated into rules and be added into the rule base, along with the CPG and experience-based rules.

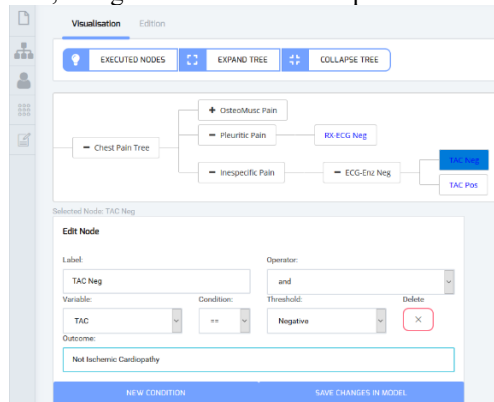


Fig. 4. Screenshot of the authoring tool GUI for the definition of a DT.

IV. VISUAL ANALYTICS

This module provides a research tool to visualize different results coming from the DE structure and clinical data. We provide two main screens: (i) a statistical dashboard with different information about administered treatments, outcomes, and compliance and (ii) parallel coordinates to show patterns among the studied clinical data for a given outcome.

The statistical dashboard focuses on the executed treatment, the outcomes, and the compliance and non-compliance

⁶ <https://orange.biolab.si/>

⁷ <https://rapidminer.com/>

⁸ <https://www.cs.waikato.ac.nz/ml/weka/>

⁹ <https://www.knime.com/>

criteria. First, a general overview of the executed treatments is given, showing the frequency of their administration. Outcome results are also analyzed, focusing on results as relapses and toxicities, where not only toxicity rates but also the kind of toxicity observed are given (e.g. toxicities can be described following the CTCAE [28] terminology, where five different grades describe the severity of the reported toxicities). Moreover, since data is retrieved from the DE stored over time, we can also explore compliance rates. We describe the compliant and non-compliant cases percentages with detailed information (i.e. the non-compliant criteria are described and shown in a graph with their usage percentage).

The parallel coordinates (see fig. 5) represents the main “n” clinical attributes as perpendicular axis, organized by their correlation level for a given outcome, having most correlated attributes next to each other [29]. Each clinical case will draw a horizontal line joining the values of the analyzed clinical attributes. Moreover, clinical cases will be grouped depending on a particular outcome (e.g. toxicity level), showing a color scale that classify them. This visualization type allows seeking for new patterns and insights among the clinical data in an easy and user-friendly way for the clinicians.

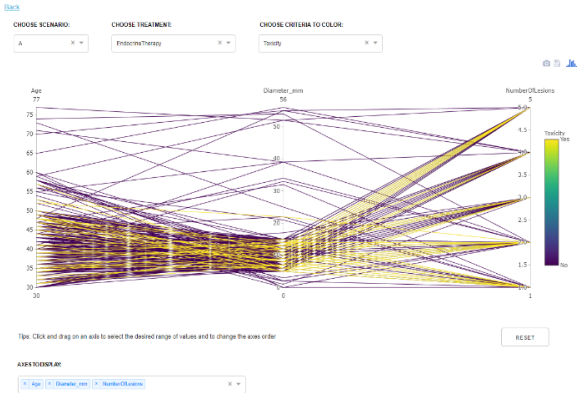


Fig. 5. Screenshot of the parallel coordinates visual analytics.

V. USE CASE: BREAST CANCER MANAGEMENT

We present a simplified use case in primary breast cancer at the diagnostic stage. We implemented the local protocol of Onkologikoa hospital as the knowledge base of the GL-CDSS. A simplified dataset of about 300 patients with 15 attributes was used for the DT generation implemented in the data mining-based CDSS, having as outcome the appearance of toxicities.

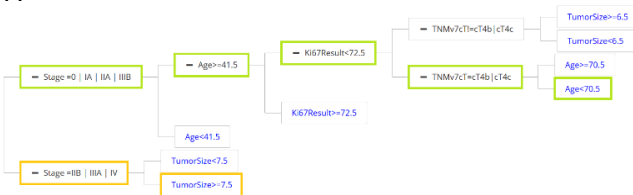


Fig. 6. The DT obtained from the primary breast cancer simplified dataset.

First, we generated the DT considering the different treatment groups available for treating primary Breast Cancer in diagnostic stage (i.e. surgery, oncology). The first node of the tree split the dataset by treatment group, predicting 0.99 of probability of having toxicities when treated with oncology versus 0.01 of probability for surgical interventions. Hence, we decided to regenerate the model ignoring this variable and we observed that patients with a tumor size larger than 7.5 mm and a Breast Cancer Stage equal to IIB, IIIA or IV had a probability of suffering any toxicity of 0.86 (orange path in fig.6). For patients with any other Breast Cancer Stage, a Ki67 lower than 72.5%, cT equal to cT4b or cT4c and age between 45 and 71 years, the score was 0.83 (green path in fig.6). We simulated a patient that matched those criteria and the guidelines gave a neoadjuvant treatment recommendation for the first case and an hormonotherapy for the second case. Thus, using the experience-based CDSS, clinicians could generate a new rule for patients that match the conditions discovered in the DT, defining a less aggressive treatment as final decision and a non-compliant criteria of high toxicity probability.

VI. DISCUSSION AND CONCLUSION

In this paper a multimodal CDSS is presented, combining guideline-based CDSS, experience-based CDSS and data mining-based CDSS. This multimodal CDSS aims to cope with the non-compliance of the guidelines by providing complementary propositions coming from alternative sources. A Decisional Event structure is presented aiming to overcome the issue of the lack of previous practice performance and its evaluation, as it stores all the information related to the decision-making process in a processable way. Some visual analytics were proposed to explore these results in a user-friendly way. Finally, authoring tools for these CDSS are presented to include the clinician actively in the review and update process of new clinical evidence.

As future work, we will extend this architecture to be fed directly from the EHR, hence, being totally integrated in the clinical workflow and we will explore its potential for epidemiological studies.

ACKNOWLEDGMENT

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 690238. Additionally, the authors acknowledge support from the four clinical partners involved in the project: Onkologikoa (San Sebastian, Spain), Eresa Grupo Médico (Valencia, Spain), Hôpital Européen Georges Pompidou (Paris, France) and Hôpital Saint-Louis (Paris, France).

REFERENCES

- [1] Institute of Medicine (US) Committee to Advise the Public Health Service on Clinical Practice Guidelines, *Clinical Practice Guidelines: Directions for a New*

- Program*. Washington (DC): National Academies Press (US), 1990.
- [2] J. M. Grimshaw y I. T. Russell, «Effect of clinical guidelines on medical practice: a systematic review of rigorous evaluations», *Lancet Lond. Engl.*, vol. 342, n.º 8883, pp. 1317-1322, nov. 1993.
- [3] M. D. Cabana *et al.*, «Why Don't Physicians Follow Clinical Practice Guidelines?: A Framework for Improvement», *JAMA*, vol. 282, n.º 15, pp. 1458-1465, oct. 1999.
- [4] K. Kawamoto, C. A. Houlihan, E. A. Balas, y D. F. Lobach, «Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success», *BMJ*, vol. 330, n.º 7494, p. 765, mar. 2005.
- [5] R. Grol, «Successes and failures in the implementation of evidence-based guidelines for clinical practice», *Med. Care*, vol. 39, n.º 8 Suppl 2.
- [6] A. Gottlieb, G. Y. Stein, E. Ruppín, R. B. Altman, y R. Sharan, «A method for inferring medical diagnoses from patient similarities», *BMC Med.*, vol. 11, n.º 1, p. 194, sep. 2013.
- [7] N. Muro *et al.*, «Augmenting Guideline Knowledge with Non-compliant Clinical Decisions: Experience-Based Decision Support», en *Innovation in Medicine and Healthcare 2017*, vol. 71, Y.-W. Chen, S. Tanaka, R. J. Howlett, y L. C. Jain, Eds. Cham: Springer International Publishing, 2018, pp. 217-226.
- [8] P. A. de Clercq, J. A. Blom, H. H. M. Korsten, y A. Hasman, «Approaches for creating computer-interpretable guidelines that facilitate decision support», *Artif. Intell. Med.*, vol. 31, n.º 1, pp. 1-27, may 2004.
- [9] A. A. Boxwala *et al.*, «GLIF3: a representation format for sharable computer-interpretable clinical practice guidelines», *J. Biomed. Inform.*, vol. 37, n.º 3, pp. 147-161, jun. 2004.
- [10] D. R. Sutton y J. Fox, «The Syntax and Semantics of the PROforma Guideline Modeling Language», *J. Am. Med. Inform. Assoc. JAMIA*, vol. 10, n.º 5, pp. 433-443, 2003.
- [11] S. Miksch, «Plan Management in the Medical Domain», *AI Commun.*, vol. 12, n.º 4, pp. 209-235, dic. 1999.
- [12] A. Galopin, J. Bouaud, S. Pereira, y B. Seroussi, «An Ontology-Based Clinical Decision Support System for the Management of Patients with Multiple Chronic Disorders», *Stud. Health Technol. Inform.*, vol. 216, pp. 275-279, 2015.
- [13] D. W. Bates *et al.*, «Ten Commandments for Effective Clinical Decision Support: Making the Practice of Evidence-based Medicine a Reality», *J. Am. Med. Inform. Assoc.*, vol. 10, n.º 6, pp. 523-530, nov. 2003.
- [16] N. Muro, N. Larburu, J. Bouaud, y B. Seroussi, «Weighting Experience-Based Decision Support on the Basis of Clinical Outcomes' Assessment», *Stud. Health Technol. Inform.*, vol. 244, pp. 33-37, 2017.
- [17] K. A. Kumar, Y. Singh, y S. Sanyal, «Hybrid approach using case-based reasoning and rule-based reasoning for domain independent clinical decision support in ICU», *Expert Syst. Appl.*, vol. 36, n.º 1, pp. 65-71, ene. 2009.
- [18] S. Safdar, S. Zafar, N. Zafar, y N. F. Khan, «Machine learning based decision support systems (DSS) for heart disease diagnosis: a review», *Artif. Intell. Rev.*, vol. 50, n.º 4, pp. 597-623, dic. 2018.
- [19] S. G. Alonso *et al.*, «Data Mining Algorithms and Techniques in Mental Health: A Systematic Review», *J. Med. Syst.*, vol. 42, n.º 9, p. 161, jul. 2018.
- [19] V. Podgorelec, P. Kokol, B. Stiglic, y I. Rozman, «Decision Trees: An Overview and Their Use in Medicine», *J. Med. Syst.*, vol. 26, n.º 5, pp. 445-463, oct. 2002.
- [20] B. Hssina, A. Merbouha, H. Ezzikouri, y M. Erritali, «A comparative study of decision tree ID3 and C4.5», *Int. J. Adv. Comput. Sci. Appl.*, vol. 4, n.º 2, 2014.
- [21] T. Hothorn, H. Seibold, y A. Zeileis, *partykit: A Toolkit for Recursive Partytioning*. 2018.
- [22] T. Therneau, B. Atkinson, B. R. (producer of the initial R. port, y maintainer 1999-2017), *rpart: Recursive Partitioning and Regression Trees*. 2018.
- [23] M. Kuhn *et al.*, *C50: C5.0 Decision Trees and Rule-Based Models*. 2018.
- [24] K. Hornik, C. Buchta, T. Hothorn, A. Karatzoglou, D. Meyer, y A. Zeileis, *RWeka: R/Weka Interface*. 2018.
- [25] E. H. Shortliffe, «Computer Programs to Support Clinical Decision Making», *JAMA*, vol. 258, n.º 1, pp. 61-66, jul. 1987.
- [26] M. Ankerst, C. Elsen, M. Ester, y H.-P. Kriegel, «Visual Classification: An Interactive Approach to Decision Tree Construction», en *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA, 1999, pp. 392-396.
- [27] S. T. Teoh y K.-L. Ma, «PaintingClass: Interactive Construction, Visualization and Exploration of Decision Trees», en *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA, 2003, pp. 667-672.
- [28] A. Trotti *et al.*, «CTCAE v3.0: development of a comprehensive grading system for the adverse effects of cancer treatment», *Semin. Radiat. Oncol.*, vol. 13, n.º 3, pp. 176-181, jul. 2003.
- [29] N. Larburu, M. Arrue, N. Muro, R. Álvarez, y J. Kerexeta, «Exploring Breast Cancer Patterns for Different Outcomes using Artificial Intelligence», en *2018 IEEE 20th International Conference on e-Health Networking, Applications and Services (Healthcom)*, 2018, pp. 1-6.