Using openEHR's guideline definition language for representing percutaneous coronary intervention patient safety rules in a dynamic checklist system

DOI:
10.3233/SHTI190611

Document status and date:
Published: 21/08/2019

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

MEDINFO 2019: Health and Wellbeing e-Networks for All
L. Ohno-Machado and B. Séroussi (Eds.)

doi:10.3233/SHTI190611

Using openEHR's Guideline Definition Language for Representing Percutaneous Coronary Intervention Patient Safety Rules in a Dynamic Checklist System

Leixing Lu[a,b], Shan Nan[a,b,c], Sicui Zhang[a,b,c], Xudong Lu[a,b,c], Huilong Duan[a,b]

[a] *Department of Biomedical Engineering, Zhejiang University, Hangzhou, P.R. China*
[b] *Key Laboratory for Biomedical Engineering, Ministry of Education, P.R. China*
[c] *School of Industrial Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands*

Abstract

openEHR's Guideline Definition Language is designed for standardizing clinical decision support systems. In this study, we use Guideline Definition Language to represent patient safety rules in pre-operation of Percutaneous Coronary Intervention for the dynamic checklist system. After using Guideline Definition Language in this case, we had some results about its expression adaptability to requirements of patient safety rules.

Keywords:

Electronic Health Records, Patient Safety, Checklist

Introduction

The dynamic checklist system is a new type of guideline-based decision support system [1]. It can make judgments on actual data of patients with an execution engine and dynamically generates a personalized checklist accordingly. The core function of such system is based on implementation of rules. However, rules are closely associated with clinical data which makes the systems difficult to implement.

openEHR is an open standard maintained by openEHR Foundation. Its core design is to separate medical domain knowledge from specific clinical information by dividing models into two levels: archetype model and reference model. Its organizers nowadays pay attention to integration problem between clinical decision support systems (CDSS) and electronic health records (EHR). They also found that expressing and sharing computerized clinical decision support content across languages and technical platforms are a long term goal. Lack of commonly shared clinical information models and flexible support for various terminology resources have been identified as two main challenges of sharing decision logic across sites. Therefore, they proposed a new guideline definition language (GDL) and added it to the openEHR Foundation.

We carried out a case study for constructing clinical rules of the dynamic checklist system in preoperative stage of Percutaneous Coronary Intervention (PCI). We used GDL to express decision logic and build knowledge base for the dynamic checklist system. Also, we used openEHR's information models as data model to represent clinical data concepts used in this case. The data source used in this case is a clinical data platform, which is also based on openEHR foundation. Then, we combined them with an execution engine. Based on these steps, we also summarized some general methods about rule editing with GDL. Finally, we discussed advantages, limitations, and directions of its future development.

Methods

Data Models for Medical Concepts

First, we chose several archetypes based on openEHR corresponding to data requirements in the preoperative check procedure of PCI. Archetypes can be searched in the official platform named Clinical Knowledge Manager (CKM) by using keywords corresponding to domain medical concepts [2].

Edit GDL Rules

A complete GDL rule mainly consists of four parts, as shown in Figure 1:

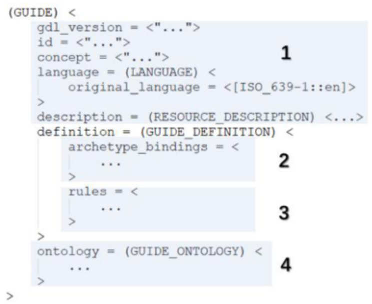

Figure 1– Basic structure of GDL

After editing a large-scale number of GDL rules, we concluded a general method for rule editing based on GDL syntax, which is mainly divided into four steps:

1. **Defining Metadata**: The metadata part is based on metadata description in the ADL, which includes mostly version information, language information, author information, keywords, rule usage and using scopes.
2. **Binding Archetypes and Elements**: We used local variables to associate with unique path of archetypes or elements.
3. **Authoring Rules**: The definition part of the rule is part of the GUIDE_definition module. Keywords WHEN and THEN are used to guide the condition and action of each rule. Priority is used for setting execution order of the rules.
4. **Defining Ontology**: The concept represented by **gt** code used in rule definition is indexed with clinical concept and can also support bindings of external term sets like SNOMED CT and ICD10.

Data Source and Engine for Rule Execution

In this study, our data source was a clinical data center platform called CDR based on openEHR. The CDR uses archetypes as the primary form for storage and access of clinical data. Data stored in CDR can be queried by the interface and returned in JSON format with mapping of path and value. The path corresponds to what we defined in the rules and the value is what we need related to the element in archetypes. By using the CDS Workbench developed by Cambio Healthcare [3], we translated the rule format from GDL to Drools. Then, we used Drools as the execution engine.

Results

The core requirement for constructing rule base is expression of a single rule. Rules were classified into six types and we made an analysis about the percentage of expressing result. The result is shown in Table 1.

Table 1– Results of rule construction

Type	Quantity	Examples	Results
Hospital procedures	5	Name, sex, age, deposit for surgery, medical insurance type.	100%
Surgical preparation	4	Medical orders and drug-related verification.	100%
General risk control	6	Blood potassium, left ventricular ejection fraction, allergies.	100%
Visual signs	8	Blood pressure, body temperature, pulse.	75%
Physical exams	6	Ultrasound, radiation.	67%
Laboratory tests	8	Blood, urine, stool, and blood coagulation routines.	50%

By executing these rules with Drools engine and running with data source from CDR platform [4], a personalized checklist was produced. Figure 2 shows one of the execution results.

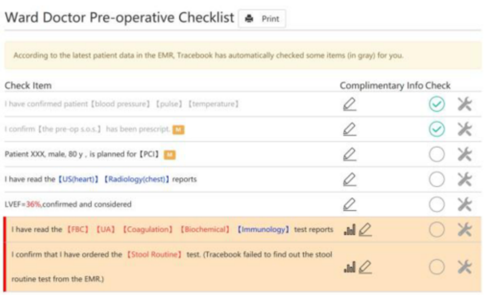

Figure 2– Execution result of the System

Discussion

According to requirements of rule expression of PCI preoperative surgery, we concluded several basic types we needed, mainly boolean expression, relational expression, logical expression, arithmetic expression, collection expression, interval expression, and data and time expression. After practicing rule construction, we found that GDL is designed to be user-friendly in expressing boolean, relation, logical, and arithmetic types; and lacking in expressing collection, interval, date and time. Collection and interval operations are important to dynamic checklist systems; keywords like foreach, where, size, contains and so on are urgently needed in cases of construction of patient safety rules. As for interval type, it can combine with data and time to express accurate date/time frame and interval without complex calculation.

The above results mainly focus on the condition part of a rule. Using GDL to fit the content of the action specification part has some shortcomings. It can hardly support our requirements totally. For example, we needed to add basic check items into a checklist template through execution of rules. As such, we needed some functions like "addTask()". Presently, we cannot find scalable functions in the action part of GDL.

Conclusions

In this study, we mainly discussed a case study about using openEHR's guideline definition language to express patient safety rules in pre-operation of PCI and its practical results. We conclude that GDL has better adaptability when constructing rules of simple expression form. At the same time, it can effectively solve the problem of data acquisition and interoperability between systems. However, the expression ability of GDL has certain limitations. Thus, it cannot completely meet requirements of dynamic checklist systems.

Future research will focus on design and implementation of a new type of guideline language to meet requirements of the dynamic checklist systems. Based on existing efforts, this new guideline language is expected to be robust, standards-based, open source tools supported, and clinically user-friendly.

Acknowledgements

Research leading to these results was funded by Chinese National Science and Technology Major Project (grant number 2016YFC0901703). The authors would like to thank their research group for their contribution to implementing the reported platforms.

References

[1] S. Nan, P.V. Gorp, X. Lu, U. Kaymak, H. Korsten, R. Vdovjak, and H. Duan, A meta-model for computer executable dynamic clinical safety checklists, *BMC Med Inform Decis Mak* **17** (2017), 170.

[2] L. Min, Q. Tian, X. Lu, and H. Duan, Modeling EHR with the openEHR approach: an exploratory study in China, *BMC Med Inform Decis Mak* **18** (2018), 75.

[3] N. Anani, M.V. Mazya, C. Rong, T.P. Moreira, O. Bill, N. Ahmed, N. Wahlgren, and S. Koch, Applying openEHR's Guideline Definition Language to the SITS international stroke treatment registry: a European retrospective observational study, *BMC Med Inform Decis Mak* **17** (2017), 7.

[4] L. Min, Q. Tian, X. Lu, J. An, and H. Duan, An openEHR based approach to improve the semantic interoperability of clinical data registry, *BMC Med Inform Decis Mak* **18** (2018), 15.

Address for correspondence

Corresponding author: Xudong Lu
Email: lvxd@zju.edu.cn
Department of Biomedical Engineering, Zhejiang University, Room 512, Zhouyiqing Building, Zhejiang University, 38 Zheda Road, Hangzhou, Zhejiang, China