Challenges of Trustable AI and Added-Value on Health
B. Séroussi et al. (Eds.)

doi:10.3233/SHTI220501

Speak-PIM, Towards a Framework for the Automatic Detection of Potentially Inappropriate Prescriptions

Abdelmalek MOUAZER[a,1], Karima SEDKI[a], Rosy TSOPRA[b,c,d] and Jean-Baptiste LAMY[a]
[a] *Université Sorbonne Paris Nord, LIMICS, INSERM, F-93000, Bobigny, France*
[b] *INSERM, Université de Paris, Sorbonne Université, Centre de Recherche des Cordeliers, F-75006 Paris, France*
[c] *HeKA, INRIA Paris, France*
[d] *Department of Medical Informatics, Hôpital Européen Georges-Pompidou, AP-HP, Paris, France*

Abstract. Potentially inappropriate medications (PIMs) have adverse health consequences, particularly in elderly patients. Various explicit criteria have been developed to detect PIMs. However, it is difficult to apply these criteria without the help of an electronic decision support tool. Programming these tools can be very complex. Indeed, for computer scientists it is difficult to understand medical issues and for clinicians it is difficult to program in a computer programming language. In this work we present Speak-PIM, a framework for formalizing the PIM's rules. Speak-PIM is based on a very simple semantics which is suitable for the declaration of PIMs without embarking on all the complexity of description logic or computer languages. It aims to offer an efficient collaboration between the computer scientists and clinicians.

Keywords. Therapeutic guidelines, Potentially inappropriate medications, Terminology, Computer programming, Decision Support System.

1. Introduction

Appropriate prescribing for the elderly is a challenge for prescribers. Several assessment tools are available. The explicit tools provide recommendations to avoid over-prescription of drugs that are not clinically indicated, omission of drugs that are necessary and incorrect prescriptions of drugs that may be indicated. The term "Potentially Inappropriate Medicines (PIMs) for the Elderly" is used to refer to medicines that should not be prescribed to this population.

Existing tools for detecting PIMs are clinical practice guidelines in natural language and include several rules. In practice, it is time-consuming for the clinician to review all these rules for each drug and for each patient. Thus, there is a great interest in implementing a computer program for automatically executing the rules.

1Corresponding Author, Abdelmalek MOUAZER, Université Sorbonne Paris Nord, LIMICS, INSERM, F-93000, Bobigny, France; E-mail: malikmouazer@gmail.com

The passage of the text to a formalized language understandable by the computer poses several problems: (i) the understanding of the text which requires medical knowledge to understand the meaning and overcome the ambiguities, (ii) the difficulties of formalizing the medical language which requires computer and medical computer skills, (iii) the implementation of the rules in a computer language which requires computer skills, (iv) the updating of the rules can be frequent regarding the evolution of knowledge. Many different types of experts are needed to solve all these problems which can generate considerable costs. We hypothesize that a rule-building system could overcome these problems. Some approaches have been proposed in the literature [1][2]. They are presented as formalization systems for all types of clinical practice guidelines, which means that they retain great complexity.

In this work, we propose Speak-PIM, a framework to formalize the rules found in PIMs guidelines, with a simplified syntax easily understandable for clinicians whether for formalization or for validation, and a formal semantics allowing its translation into SPARQL queries.

2. Methods

2.1. Analysis of PIM guidelines

The design of the framework is first preceded by the analysis of the logic of the rules detecting the PIMs (rules found in the PIMs guides). At this stage, we analyzed the STOPP & START guideline [3]. We have taken this guideline as a reference because it is widely used in the literature and because it includes more complex rules, compared to other guidelines, some of them being limited to a list of unconditionally inappropriate drugs. The purpose of this step is to identify the logic elements needed for detecting PIMs (such as logical operations), the clinical elements and the attributes necessary for their declaration.

The logic of expression of the criteria:
In all STOPP & START guidelines, each rule is formulated as follows:

In the case of a STOPP rule: a prescription is considered potentially inappropriate if it is present, possibly with some additional conditions. The conditions may be the presence and/or absence of one or more clinical elements. These elements can be mandatory (succession of logical “and” between these elements), or one of these elements is enough to declare a prescription as being potentially inappropriate (combination of elements with logical “or”). For example, stopp_H1 rule states : “*Stop non-steroidal anti-inflammatory drug (NSAID) other than COX-2 selective agents with history of peptic ulcer disease or gastrointestinal bleeding, unless with concurrent PPI or H2 antagonis*t”. To declare NSAID as being a PIM, the patient must present at least *gastrointestinal bleeding* or a have an history of peptic ulcer (presence peptic ulcer history "or" *gastrointestinal bleeding*) and at the same time his overall treatment does not include PPI or h2 antagonists (absence of PPI “and” absence of H2 *antagonis*t).

In the case of a START rule: the recommended prescription must be absent from the current drug order, and some additional conditions must be satisfied. The conditions are expressed in the same way as for STOP rules.

2.2 .Coding and mapping

Once the rule is declared with Speak-PIM, it is automatically translated into a SPARQL query using a program written in Python.

The input data model is based on the OMOP-CDM [4] model translated in a previous work in OWL.

Several terminologies can be used to code clinical elements when using Speak-PIM. For simplicity, the terminologies used in this work are: ATC for prescriptions, ICD10 for diseases and LOINC for observations. These terminologies have been managed by Owlready2 [5].

3. Results

3.1 Presentation of the framework

The formalization of a rule with the SPEAK-PIM framework must be done in two steps: the declaration of the clinical elements and the writing of the rule.

The declaration of the clinical elements: The first step to formalize a rule with SPEAK-PIM consists in declaring the clinical elements necessary for the expression of the rule with the framework. There are three categories of clinical elements: prescriptions, diseases, and observations (results of biological tests). Each element can be defined by a set of attributes which can be mandatory or optional. Table 1 shows the different attributes available for defining a clinical element.

Table 1. List of possible attributes for each clinical element

Prescription	Disease	Observation
-concepts -indication -is_ongoing -duration -dose_unit -lower_than -greater_than	**-concepts** -is_active	**-concepts** -value -lower_than -greater_than

Concepts: terminological code(s) of the clinical element; indication: terminology code(s) of the disease for which the drug was prescribed; is_ongoing: if the prescription is in progress (true or false); duration: treatment duration in days; dose_unit: dose expression unit; lower_than(Prescription): lower critical dose threshold per day; greater_than(Prescription): upper critical dose threshold per day; is_active: if the disease is current or it is an antecedent; value: observation value; lower_than(Observation): critical lower threshold of the observation; greater_than(Observation): critical upper threshold of the observation.

The Form of the rule: Each rule is in the form:

RULE_XX = Check_pim(action, target, presents = [EP1, EP2, ... EPn One_among(EPO1, EPO2, ... EPOn)], absents =[EA1, EA2, … EAn])

arguments:

action: can take two values “STOP” to stop a prescription or “START” to start a new prescription**.**

target: target prescription, this is the prescription to be stopped or started.
presents = [Epi, One_among(EPOi)]: all the clinical elements in this list must be present to trigger the rules. If only one of a set of clinical elements is sufficient to trigger the rule, all of these elements are put in the "One_among()" object. Between all the objects in the "present" list there are logical connectives "AND".
absents = [EAi]: all the clinical elements in this list must be absent to trigger the rule.

Rule execution: The execution of the rule, namely: carrying an action (stop or start) on a target prescription is carried out when the condition (present AND absent is true). In a more formal way, we can write the following formula:

$$[EP1 \wedge EP2 \wedge ... \wedge EPn \wedge (EPO1 \vee EPO2 \vee ... \vee EPOn)] \wedge [\neg EA1 \wedge \neg EA2 \wedge ... \wedge \neg EAn] \Rightarrow action(target)$$

3.2 Application to STOPP & START CRITERIA

To test the capacity of this framework to formalize PIMs, we used it to formalize the STOPP & START criteria. These criteria have previously been translated into a structured format [6]. We used this work as a basis for comparison and pre-validation of Speak-PIM. Figure 1 shows an example of formalization of the "stopp D4" criterion from the STOPP & START criteria.

```
texte_stopp_d4 = """Stop selective serotonin re-uptake inhibitors (SSRI's)
with current or recent significant hyponatraemia i.e. serum Na+ < 130 mmol/l
(risk of exacerbating or precipitating hyponatraemia)."""hypo_na =
Disease(concepts=["E87.1"])hypo_na_mesure =
Observation(concepts=["77139-4"], lower_than(130))ssris =
Prescription(concepts=["N06AB", "N06CA03"])rule_D4 = Check_pim(
action = "STOP", target = ssris), presents = [ssris, One_among(hypo_na,
hypo_na_mesure)]
)
```

Figure 1. Example of formalization of the stopp D4 criterion from the STOPP & START criteria

Of the 80 STOPP rules, 77 rules were formalized; (A1, A2 and A3 have not been formalized because they are very general and do not target specific drugs). The 77 formalized rules required 83 rules with Speak-PIM. Indeed, a few guideline rules had high-level disjunctions and needed two Speak-PIM rules to be formalized. Moreover, all 34 START rules have been formalized with 41 rules with Speak-PIM.

Applied to a test patient, all the STOPP & START rules could be applied in 0.05 seconds (Dell Inc. Latitude 7410; 15,3 Gio; Intel® Core™ i7-10810U CPU @ 1.10GHz × 12).

4. Discussion and Conclusions

The framework we present is easy to use and understand, whether for the formalization of PIMs guidelines or for their validation. It makes it possible to target PIMs with taking into account the patient's conditions, which is important to avoid "alert-fatigue". The

execution time of all the rules is satisfactory for a real-time implementation in a computer application. Speak-PIM offers a limited and rigid syntax different from standard reference syntax like an “Arden syntax” which offers greater freedom of expression of the rules. This rigidity is desired to allow health professionals to be able to express the rules of PIMs guidelines without computing knowledge.

The data model based on OMOP-CDM makes it interoperable with all patient record systems based on this model without recourse to mapping between different models.

Speak-PIM does not take into account the severity of diseases or the effectiveness of a drug on a patient, information that is sometimes a decisive criterion for declaring a drug as PIM. Technically, we can add an attribute (efficiency, severity of the disease, etc.) but in practice this piece of information is missing from patient records.

Currently, Speak-PIM offers limited support for drug dose. Indeed, our model can only capture the quantity of an active ingredient administered per day, but the characterization of the dosage is more complex than what we propose. However, in the PIM guidelines, all the doses identified were relative to the quantity administered per day.

In the future, we plan to translate Speak-PIM into other computer languages.

Acknowledgments

This work was partly funded by the French Research Agency (ANR) through the ABiMed project [grant number ANR-20-CE19-0017-02].

References

[1] Kluegl P, Toepfer M, Beck PD, Fette G, Puppe F. UIMA Ruta: Rapid development of rule-based information extraction applications. Natural Language Engineering. 2016 Jan;22(1):1-40.

[2] Lamy JB, Ebrahiminia V, Riou C, Seroussi B, Bouaud J, Simon C, Dubois S, Butti A, Simon G, Favre M, Falcoff H. How to translate therapeutic recommendations in clinical practice guidelines into rules for critiquing physician prescriptions? Methods and application to five guidelines. BMC medical informatics and decision making. 2010 Dec;10(1):1-4.

[3] O'Mahony D, O'Sullivan D, Byrne S, O'Connor MN, Ryan C, Gallagher P. STOPP/START criteria for potentially inappropriate prescribing in older people: version 2. Age and ageing. 2014 Oct 16;44(2):213-8.

[4] Lamy JB, Mouazer A, Sedki K, Tsopra R. Translating the Observational Medical Outcomes Partnership-Common Data Model (OMOP-CDM) electronic health records to an OWL ontology. InMEDINFO 2021.

[5] Lamy JB. Owlready: Ontology-oriented programming in Python with automatic classification and high level constructs for biomedical ontologies. Artificial intelligence in medicine. 2017 Jul 1;80:11-28.

[6] Huibers CJ, Sallevelt BT, de Groot DA, Boer MJ, van Campen JP, Davids CJ, Hugtenburg JG, van Hout HP, van Marum RJ, Meulendijk MC. Conversion of STOPP/START version 2 into coded algorithms for software implementation: a multidisciplinary consensus procedure. International Journal of Medical Informatics. 2019 May 1;125:110-7.