Using preference learning for detecting inconsistencies in clinical practice guidelines: Methods and application to antibiotherapy

Rosy Tsopra, Jean-Baptiste Lamy, Karima Sedki

Using preference learning for detecting inconsistencies in clinical practice guidelines: methods and application to antibiotherapy

Rosy Tsopra[a,b,1,*], Jean-Baptiste Lamy[a,1], Karima Sedki[a]

[a]*LIMICS, Université Paris 13, Sorbonne Paris Cité, 93017 Bobigny, France, INSERM UMRS 1142, UPMC Université Paris 6, Sorbonne Universités, Paris, France*
[b]*AP-HP, Paris, France*

Abstract

Clinical practice guidelines provide evidence-based recommendations. However, many problems are reported, such as contradictions and inconsistencies. For example, guidelines recommend sulfamethoxazole/trimethoprim in child sinusitis, but they also state that there is a high bacteria resistance in this context. In this paper, we propose a method for the semi-automatic detection of inconsistencies in guidelines using preference learning, and we apply this method to antibiotherapy in primary care. The preference model was learned from the recommendations and from a knowledge base describing the domain.

We successfully built a generic model suitable for all infectious diseases and patient profiles. This model includes both preferences and necessary features. It allowed the detection of 106 candidate inconsistencies which were analyzed by a medical expert. 55 inconsistencies were validated. We showed that therapeutic strategies of guidelines in antibiotherapy can be formalized by a preference model. In conclusion, we proposed an original approach, based on preferences, for modeling clinical guidelines. This model could be used in future clinical decision support systems for helping physicians to prescribe antibiotics.

Keywords: Preference learning, Antibiotherapy, Clinical practice guidelines, Inconsistencies in guidelines

1. Introduction

In the 1990s, the concept of *Evidence-Based Medicine* was introduced and defined as "the integration of best research evidence with clinical expertise and patient values" [1]. This new paradigm led to the redaction and diffusion of Clinical Practice Guidelines (CPGs) by national health authorities [2]. CPGs are narrative documents providing recommendations stated by a group of experts according to a systematic review of the available clinical evidence. They aim at improving the quality of health care by providing standardized best practices for diagnosis and treatment. Their development is complex and requires time, rigor and multiple verification and validation steps to guarantee their quality [3, 4, 5, 6]. However, many problems are reported like incompleteness, contradiction, inconsistency, redundancy or ambiguity within CPGs [4]. This leads to a lack of confidence of physicians in CPGs [7], and thus a poor consideration of CPG recommendations in their daily routine clinical practice [8].

For verifying the quality of recommendations within CPGs, various methods were developed. The structure of CPGs can be verified by tools [9, 10] such as AGREE instrument [11]. These tools focus on quality criteria, *e.g.* presentation of guidelines, or independence of experts [12]. However, these methods are limited to the verification of the structure of CPGs, and do not consider the consistency and medical pertinence of recommendations.

The consistency of recommendations can be verified by formal methods [13]. The recommendations are first represented using an explicit and non-ambiguous model in a formal language. Several Computed Interpretable Guidelines (CIG) were developed [14]. They allow detecting ambiguity, incompleteness, inconsistency or redundancy within CPGs [6, 15, 13, 3, 16, 17]. For example, some authors [18, 19] state that, if narrative guidelines are encoded into logical language ("if... then..." rules), then the generation of all possible variable combinations allows the detection of incompleteness (*i.e.* variable combinations not covered by CPGs) and inconsistencies (*i.e.* similar variable combinations leading to different conclusions). But these methods are time-consuming and dependent on the formal language. Moreover, these formal approaches don't verify the medical pertinence (*e.g.* they do not verify that the recommended drug treatments are not contraindicated for the patient).

Few approaches have been proposed for verifying the medical pertinence of recommendations. These approaches require the formalization of the medical knowledge involved (*e.g.* drug properties such as contraindications) and the identification of the medical principles underlying the recommendations of CPGs. However, formalizing the knowledge and the reasoning principles is a complex task [13]. For example, in oncology, a medical domain where multiple drugs are often prescribed, the adverse events can be limited by checking the known adverse effects [20].

Recently, many approaches have been proposed for enriching recommendation by integrating additional information. These pieces of information concern particularly patient context (psycho-social, multi morbidity, *etc.*) and patient preferences [21, 22, 23, 24, 25, 26]. For example, in multi-criteria decision making, to recommend an appropriate manual wheelchair, user preferences that are often conflicting must be taken into account [27]. Nevertheless, the manual construction of preferences remains complex and time-consuming. Thus, it is more appealing to learn preferences from data, because in general, data are easily collected or observed.

In this article, we propose a method for the semi-automatic detection of inconsistencies in guidelines using preference learning, and we apply this method to antibiotherapy in primary care. In primary care, CPGs recommend prescribing antibiotics empiri-

*Corresponding author

This is an author file of the article published in Artificial Intelligence In Medicine 2018, DOI: 10.1016/j.artmed.2018.04.013 ; it is available under Creative Commons Attribution Non-Commercial No Derivatives License.

Email addresses: rosytsopra@gmail.com (Rosy Tsopra), jibalamy@free.fr (Jean-Baptiste Lamy), sedkikarima@yahoo.fr (Karima Sedki)

[1]These two authors contributed equally to this work.

	Input	Objective
Label ranking	A set of instances $\mathcal{X} = \{x_i \mid i = 1 \dots n\}$. A set of labels $\mathcal{L} = \{l_k \mid k = 1 \dots m\}$. For each instance x_i, a set of pairwise preferences of the form $l_k \succ_{x_i} l_j$.	Find a ranking function that maps any $x \in \mathcal{X}$ to a ranking $\succ_x \in \mathcal{L}$.
Instance ranking	A set of instances $\mathcal{X} = \{x_i \mid i = 1 \dots n\}$. A set of ordered labels $\mathcal{L} = \{l_k \mid k = 1 \dots m\}$ such that: $l_1 \succ l_2 \succ \dots l_k$. A label l_k is associated with each instance x_i.	Find a ranking function that allows to order a new set of instances according to their (unknown) preference degrees.
Object ranking	A set of objects $\mathcal{X} = \{x_i \mid i = 1 \dots n\}$. A finite set of pairwise preferences $x_i \succ x_j$.	Find a ranking function that assumes as input a set of objects and returns a permutation (ranking) of this set.

Table 1: Comparison between the different ranking problems.

cally, *i.e.* without knowing the causative bacteria and its susceptibility to the various antibiotics. The most likely bacteria are guessed from the infectious disease (*e.g.* cystitis is usually caused by E. coli). Then, CPGs recommend an antibiotic according to the various antibiotics features (*e.g.* susceptibility of the likely causative bacteria, side effects) and the patient profile (*e.g.* child or adult) [28, 29].

In order to detect inconsistencies in these CPGs, we made the following hypotheses: 1) it is possible to learn a preference model from the recommendations and a knowledge base describing the domain; 2) a generic model can be defined for all infectious diseases and all patient profiles encountered in CPGs; and 3) this preference model can be used to detect inconsistencies in CPGs.

The rest of the paper is organized as follows. Section 2 gives background about preference learning, and describes the optimization algorithm we used and the antibiotherapy knowledge base we previously designed. Section 3 describes the preference learning. Section 4 describes the detection of inconsistencies and their validation by a medical expert. Section 5 discusses the methods and the results obtained, and finally, concludes.

2. Background

2.1. State of the art in preference learning

Preferences are basically acquired in two ways: i) by elicitation from the user (for instance through a sequence of queries/answers) or ii) by directly learning them from data. Preference elicitation is often time-consuming, especially if the number of alternatives/outcomes is large. Moreover, different elicitation techniques are likely to provide different results. It is then more appealing to learn preference from data which is easy to observe and collect. Preference learning is one of the research problems that have recently received considerable attention in disciplines such as artificial intelligence, machine learning, data mining, decision making and others. It aims to learn and construct a preference model from observed preference information. Once the preference model learned, it can be used for decision making for instance. Preference learning can be formalized within various settings, depending for example on the underlying preference model and the type of input provided to the learning system. We can distinguish three common problems in preference learning [30]: i) learning from label preferences (also designed as label ranking in the literature because frequently, the predicted preference relation is required to form a total order), ii) learning from instance preferences (instance ranking) and iii) learning from object preferences (object ranking). Table 1 summarizes the different ranking problems.

In label preferences problem [31, 32], the training data contains a set of instances. A set of pairwise comparisons between labels is associated with each instance, expressing that one label is preferred over another for that instance. The objective is to use these pairwise preferences for predicting a ranking function that attributes for any instance a ranking (a total order in general) of all possible labels. Namely, the task is to rank the set of labels for a new instance (label ranking). Label ranking can be considered as a generalization of the supervised classification problem where an order over class labels is associated with an instance instead of only one class label. As an example of a label ranking problem, consider a set of labels $\mathcal{L}$ representing three types of activities: football, tennis and basket. The training data contains a set of students who have to give a list of pairwise preferences between activities (*e.g.* $\{(Adrien, [football \succ tennis]), (Marie, [tennis \succ football]\})$. Thus, the aim is to compute a ranking over the labels for each instance. For example, the possible prediction of the learnt function for a student x is football $\succ$ basket $\succ$ tennis.

In the setting of learning from instance preferences problem [33], the input contains a set of ordered labels and a set of instances, each one associated with a label. The objective is to find a ranking function that allows ranking a given new set of instances. In case where there are two ordered labels, the problem of learning is often called bipartite ranking problem [34]. In case where there are more than two ordered labels, the problem of learning is often called multipartite-ranking problem [35].

Concerning learning from objects [36, 37], the objective is to learn a model that allows determining which object is preferred to another. The training data is given in the form of pairwise comparisons between objects. For this type of learning problems, there is no supervision since no class label is associated with an object and each object is not necessarily represented by a set of features or attributes. As an example, to rank query results of a search engine, user clicks on some of the links in the query result and not on others can be exploited to provide training information. Thus, selected pages are preferred over pages that are not clicked.

Two approaches can be distinguished for preference modeling and learning: quantitative and qualitative approaches. Quantitative preferences learning [38, 39] consist mainly to learn a utility function on training data. This function assigns a utility degree (or a score) to each alternative (instance, object or label) following the learning problem. For learning problems that are based on qualitative approach [40, 41, 42, 43, 44], the objective is to learn a binary preference relation that compares each pairs of alternatives.

When it comes to modeling utility functions, the task is rather more complicated since users may not be used with this formal-

ism and the problem size could be very large. Utility functions (for example, the one a user is supposed to use while making decisions) can be inferred or estimated from past decisions. In [45], this problem is solved by imposing constraints derived from the data over the set of all utility functions. One could go one step further by searching for the optimal utility function given the available constraints. Among first works dealing with deriving utility functions from data, one can mention [46] where the authors aim at extracting reward functions given optimal behaviors in the context of Markov Decision Processes. The main issues dealt with the literature last years concern noise and data inconsistencies and uncertainty, large search spaces and taking into account data sequence, *etc*. In [47], the authors proposed an approach to learn utility functions allowing to monitor requirements of a dynamically adaptive system. The learned utility functions map at run time monitoring information to a value assessing how well a requirement is satisfied.

Once the preferred model is learned, there is need to measure its quality of prediction. For this, different performance measures can be used such as precision, recall, NDCG (Normalized Discounted Cumulative Gain), *etc*. In addition, preference learning methods require optimization algorithms. In this study, we will use the Artificial Feeding Birds (AFB) metaheuristics [48, 49]. We developed this metaheuristics in previous works, and we describe it in the following section.

2.2. Artificial Feeding Birds (AFB) metaheuristics

Nature-inspired metaheuristics [50, 51] are an optimization approach, which often leads to simple, efficient and adaptable algorithms. In this work, we chose Artificial Feeding Birds (AFB) [48, 49], a recent metaheuristic inspired by the behavior of pigeons.

AFB considers a population of artificial birds. The position of a bird corresponds to a candidate solution to the optimization problem. The optimization process aims at minimizing the $cost()$ function, (*i.e.* finding x such as $cost(x)$ is as low as possible). The value returned by the $cost()$ function is associated with the presence of food: if the value of the $cost()$ function for the new position of a bird is lower than his previous lowest value, the bird has found some food and he keeps his current position in memory.

In each iteration of the algorithm, each bird performs a move. Four possible moves are considered: (1) the bird walks to a close random position, (2) the bird flies to a completely random position, (3) the bird flies back to the best position he has encountered so far, and (4) the bird flies to join the position of another random bird. For a given bird, the next move is determined as follows: if the bird has flown in the previous cycle, he walks. If the bird has found a better solution in the previous cycle, he walks. Otherwise, the next move is chosen randomly, with different probabilities associated with each move. Two sizes of birds are considered, and the fourth move (join another bird) can only be performed by the largest birds (representing 25% of the population).

The optimization problem is defined by three functions: $cost()$, the function to minimize, $fly()$, a function that returns a random position in the solution space, and $walk(i)$, a function that modifies the current position of the bird i by performing a small random move.

The metaheuristic takes 5 parameters:
- $n = 20$, the number of artificial birds,
- $r = 0.75$, the ratio of small birds in the total bird population,
- $p_2 = 0.01$, the probability that a bird chooses move 2,
- $p_3 = 0.67$, the probability that a bird chooses move 3,
- $p_4 = 0.07$, the probability that a bird chooses move 4.

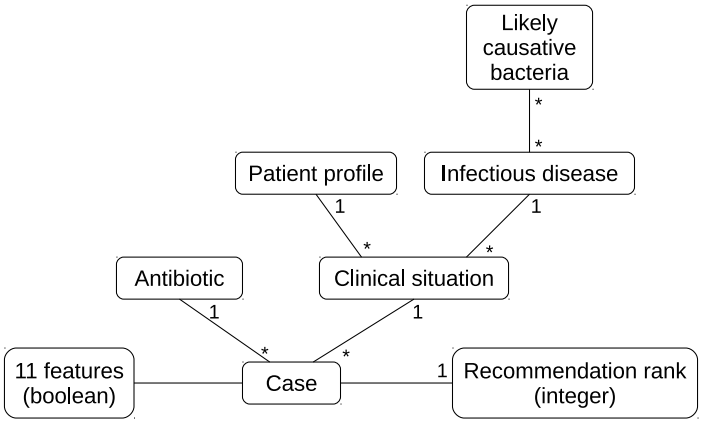

Figure 1: General model of the knowledge base.

The parameter values given here are the ones we used; they correspond to the default values proposed in [49]. The AFB algorithm is known to have a low sensitivity to parameter values, and these values were successfully used in very different optimization problems (non-linear global optimization on benchmark functions, neural network training, combinatorial optimization). The metaheuristics algorithm is given in Supplementary File 1.

2.3. Antibiotherapy knowledge base

In previous works [28, 29], we designed an antibiotherapy knowledge base for helping physicians with empirical prescription of antibiotics. The knowledge base contains information related to 11 infectious diseases, the 50 antibiotics marketed for use in primary care in France, and 21 patient profiles.

Figure 1 shows the general model of the knowledge base. Infectious diseases are associated with the likely causative bacteria (one or several; several types of bacteria can cause the same disease). A patient profile is described by the age class and the presence or absence of pregnancy, allergy, and history of antibiotic treatment. A clinical situation corresponds to the intersection of an infectious disease and a patient profile. Finally, a given antibiotic prescribed for a given clinical situation is called a *case* in the knowledge base. It is the lowest granularity level in the knowledge base. Each case has an integer *recommendation rank*, which is 0 if the antibiotic is not recommended by French CPGs in the clinical situation, 1 if it is recommended as a first-line treatment, 2 if it is recommended as a second-line treatment, and so on (up to 4).

In addition, cases are characterized by 11 features (listed in Table 2). In the paper, each feature is identified by a short name, such as *protocol.* For a given feature, the value may depend on the antibiotic, the patient profile, the infectious disease and/or the likely causative bacteria. For example, *no side ef* depends only on the antibiotic, whereas *no contraindication* depends on the antibiotic and the patient profile (*e.g.* a given antibiotic may be contraindicated for children but not for adults). Each feature is Boolean ($True$ if the feature holds for the case and $False$ otherwise), with possible missing values ($Unknown$) when no data is available in the medical literature. For each feature, $True$ corresponds to an advantage for the antibiotic, whereas $False$ is a disadvantage.

The knowledge base was built and populated by a medical doctor (RT) through a 2 steps-process. In step 1, RT extracted the features of antibiotics and their dependence relationships [28, 29], from the manual analysis of six CPGs related to urinary [52, 53] and respiratory [54, 55, 56, 57] infections. In step 2, RT populated the knowledge base. Because of the lack

#	Feature [*short name*] Definition	Antibiotic	Patient profile	Infectious disease	Causative bacteria
1	**Naturally active against the causative bacterium** [*naturally active*] Whether the causative bacterium is described as sensitive or of intermediate sensitivity for the antibiotic (*e.g.* Amoxicillin is naturally active against Group A streptococci)	X			X
2	**Probably active against the causative bacterium** [*probably active*] Whether the frequency of resistance in the causative bacterium is less than Y% (*) for the antibiotic (*e.g.* Ceftriaxone is probably active against E.coli)	X		X	X
3	**Clinical efficacy proven in the disease** [*proved*] Whether the antibiotic is described as clinically effective for treating the infection OR is (or has been) indicated/recommended for the infection (*e.g.* Penicillin G has proven its clinical efficacy for pharyngitis treatment)	X		X	
4	**Absence of contraindications for the patient** [*no contraindication*] Whether there is no absolute contraindication of the antibiotic for the patient profile (*e.g.* Pristinamycin is not contraindicated if the child is more than 6 years old)	X	X		
5	**Convenient protocol** [*protocol*] Whether the antibiotic is prescribed for oral administration AND for a duration of less than Z (*) days (*e.g.* Fosfomycin trometamol has a convenient protocol in uncomplicated cystitis)	X		X	
6	**Non-precious class** [*not precious*] Whether the antibiotic doesn't belong to a class of drugs that must be preserved for more serious infections (*e.g.* Amoxicillin is a non-precious class in sinusitis)	X		X	
7	**Absence of serious and frequent side effects** [*no side ef*] Whether there is no serious side effects mentioned AND the frequency of side effects is sufficiently low for antibiotic prescription to be allowed (e.g. Fosfomycin trometamol gives no serious and rare side effects)	X			
8	**High level of efficacy** [*ef ficacy level*] Whether the antibiotic is described as very effective (high clinical cure percentage, *e.g.* Levofloxacin is very effective in prostatitis)	X		X	
9	**Narrow antibacterial spectrum** [*spect*] Whether the antibiotic is described as having a "narrow" antibacterial spectrum (*e.g.* Nitrofurantoin has a narrow activity spectrum)	X			
10	**Low level of ecological adverse effects** [*low eco risk*] Whether the antibiotic is described as having a low risk of promoting bacterial resistance (*e.g.* Pivmecillinam has a low level of ecological risk)	X			
11	**Taste** [*taste*] Whether the antibiotic has an acceptable taste for the patient (*e.g.* Cefuroxime axetil has a bad taste and thus is not acceptable for children)	X	X		

Table 2: The 11 features of antibiotics in the knowledge base. The last four columns indicate on which the feature value depends. (*) Y and Z are values specific to each infectious disease.

of existing antibiotic knowledge base able to describe the features identified in step 1, only textual resources were used to populate the knowledge base. RT manually extracted the values for each feature (*i.e.* $True$, $False$, or $Unknown$) from seven French CPGs [52, 53, 54, 55, 56, 58, 59], two international CPGs [60, 61], five antibiotic guidelines produced by national authorities [62, 63, 64, 65, 66], and one reference textbook on infectious diseases [67]. For example, for fosfocmycin trometamol in women cystitis, $protocol = True$ was extracted from the following CPG excerpt [53]: "Fosfomycin trometamol is recommended in women cystitis because it can be given in single dose". Each value was then blindly validated by two antibiotic specialists during a 5-round Delphi Process (14.7% of the values were modified).

Finally, the knowledge base has been formalized as an OWL 2.0 ontology. It contains 144,038 RDF triples describing 5,696 classes, 19 properties and 34,483 axioms, and belongs to the $\mathcal{ALC}(\mathcal{D})$ family[2] of description logics (DL).

Case	x_i^a (antibiotic)	x_i^s (clinical situation)	*no contraindication*	*no side ef*	*protocol*	$\in \mathcal{Y}_{reco}$
x_1	Drug A	Situation X	*False*	*False*	*False*	
x_2	Drug B	Situation X	*True*	*False*	*False*	
x_3	Drug C	Situation X	*True*	*True*	*False*	yes
x_4	Drug D	Situation X	*True*	*False*	*True*	yes
x_5	Drug E	Situation X	*Unknown*	*True*	*True*	
x_6	Drug A	Situation Y	*Unknown*	*False*	*False*	
x_7	Drug B	Situation Y	*True*	*False*	*False*	
x_8	Drug C	Situation Y	*True*	*True*	*False*	
x_9	Drug D	Situation Y	*True*	*False*	*True*	yes
x_{10}	Drug E	Situation Y	*True*	*True*	*True*	yes

Table 3: Simple example showing the structure of the knowledge base.

3. Building the preference model

3.1. Modeling the knowledge base

With regard to preference learning, we consider an *instance* to be a given antibiotic in a given clinical situation (*i.e.* what we called a *case* in the ontology, Figure 1). Building a preference model from the case features and the recommendations of CPGs is an instance learning problem (as described in section 2.1). In the knowledge base, there are 5 recommendation ranks: recommended as 1st, 2nd, 3rd, 4th line of treatment (R_1, R_2, R_3, R_4, respectively) or not recommended (NR), which correspond to 5 labels that are ordered ($R_1 \succ R_2 \succ R_3 \succ R_4 \succ NR$). There are 102 cases labeled R_1, 62 labeled R_2, 30 labeled R_3, 2 labeled R_4 and 3104 labeled NR. Since there are very few cases having the R_3 and R_4 labels, for facilitating the learning of preferences, we divide this multipartite-ranking problem in two bipartite ranking problems. In the first problem, we consider the two following labels: R_1 *vs* $R_2 \cup R_3 \cup R_4 \cup NR$ (*i.e.* all cases not recommended as first-line treatment are grouped together). In the second problem, the labels are: $R_1 \cup R_2 \cup R_3 \cup R_4$ *vs* NR (*i.e.* all cases recommended are grouped together whatever their recommendation rank). Thus, we will learn two different preference models, respectively named $\mathcal{M}_1$ and $\mathcal{M}_{any}$, one for each problem. The two models will be learned following the same method, described below.

We model the antibiotic knowledge base as a *case space* $\mathcal{X} = \{x_1, ...x_i, ..., x_n\}$ with $x_i = (x_i^a, x_i^s, x_{i,1}, ..., x_{i,j}, ..., x_{i,p})$ where x_i^a is the identifier of the antibiotic, x_i^s is the identifier of the clinical situation, and $x_{i,j}$ are the feature values, with $x_{i,j} \in \{True = +1, False = -1, Unknown = 0\}$. n is the number of cases and p is the number of features ($p = 11$). We considered the 66 clinical situations for which at least one recommendation exists in CPGs, and the 50 antibiotics available on the market in primary care in France in 2014. Consequently, $n = 66 \times 50 = 3300$.

In our formalization, we added x_i^a and x_i^s in cases, in order to permit having several cases with the same feature values in set $\mathcal{X}$, provided that the antibiotic or the clinical situation differs. However, x_i^a and x_i^s are not considered as features for the purpose of learning. We also define $\mathcal{S}$, a partition of $\mathcal{X}$ according to clinical situations: $\mathcal{S} = \{\mathcal{S}_1, \mathcal{S}_2, ..., \mathcal{S}_k, ...\}$ where $\mathcal{S}_k$ is the set of all cases sharing a given clinical situation k: $\mathcal{S}_k = \{x \in \mathcal{X} \mid x_i^s = k\}$.

Finally, we represent the two labels as a set of two classes $\mathcal{Y} = \{\mathcal{Y}_{reco}, \mathcal{Y}_{not\,reco}\}$ which is a partition of the case space $\mathcal{X}$. $\mathcal{Y}_{reco}$ contains the cases recommended (in first line of treatment for $\mathcal{M}_1$, and in any line for $\mathcal{M}_{any}$), and $\mathcal{Y}_{not\,reco}$ contains the others. Therefore, $\mathcal{Y}_{reco}$ is preferred to $\mathcal{Y}_{not\,reco}$ (written $\mathcal{Y}_{reco} \succ \mathcal{Y}_{not\,reco}$).

[2] $\mathcal{AL}$: attribute language (including atomic negation, concept intersection, universal restriction, existential qualification limited to class Thing), $\mathcal{C}$: complex negation, ($\mathcal{D}$): use of datatypes [68].

Feature	N_j	w_j	**Category**
no contraindication	0%	-	Necessary
no side ef	50%	0.2	Preference
protocol	25%	0.3	
t_{ness}	10%		Threshold
t_{pref}		0.5	

Table 4: Results of the learning process on the example of Table 3.

Table 3 shows the structure of the knowledge base on a simplified example with only 5 antibiotics, 2 clinical situations, $p = 3$ features and $n = 10$ cases.

3.2. Determining necessary and preference features

We make the hypothesis that two categories of features exist: *necessary* features and *preference* features. Necessary features are mandatory for prescribing the antibiotic: if the feature does not hold for an antibiotic in a clinical situation, the antibiotic should not be prescribed and thus it is not recommended (necessary features can be viewed as constraints). On the contrary, preference features are not mandatory; however, an antibiotic having a preference feature is preferred to another antibiotic without the preference feature. For example, the absence of contraindications may be a necessary feature while the low rate of adverse effects may be a preference feature.

In order to learn which features are necessary, for each feature j (with $1 \leq j \leq p$) we compute N_j, the percentage of recommended cases for which the feature j is not necessary to recommend the antibiotic, *i.e.* feature j is not *True* but the antibiotic is nevertheless recommended in guidelines:

$$N_j = \frac{\left|\{x_i \in \mathcal{Y}_{reco} \mid x_{i,j} \neq 1\}\right|}{|\mathcal{Y}_{reco}|} \quad (1)$$

Table 4 shows the computed values for N_j on the simplified example of Table 3. In an "ideal" error-less system, a necessary feature has $N_j = 0$ (*e.g.* feature *no contraindication* in the example). In real life, there may be a few errors. Thus, we consider as necessary all features j having $N_j \leq t_{ness}$, where t_{ness} is a threshold that will be learned later. We define F_{ness} and F_{pref}, the sets of necessary and preference features, respectively (each feature being identified by its index).

$$F_{ness} = \left\{j \subseteq \{1, 2, ..., p\} \mid N_j \leq t_{ness}\right\} \quad (2)$$

$$F_{pref} = \{1, 2, ..., p\} \setminus F_{ness} \quad (3)$$

In the example (Table 4), the learning process leads to $t_{ness} = 10\%$. This means that *no contraindication* is a necessary feature, whereas *protocol* and *no side ef* are preference features.

3.3. Defining the utility function

For preference features, we define a utility function $f : \mathcal{X} \rightarrow \mathbb{R}$ that allows ordering cases according to their degree of preference, *i.e.* $f(x_i) \geq f(x_j) \Rightarrow x_i \geq x_j$. We define f as a linear combination of the preference features:

$$f(x_i) = \sum_{j \in F_{pref}} x_{i,j} \times w_j \quad (4)$$

The weights w_j will be learned later. The utility function quantifies the utility of an antibiotic in a clinical situation. However, the utility is meaningless *per se*, and it can only be exploited relatively to the utility of other antibiotics. For example, an antibiotic with a low utility could be recommended if no better antibiotic exists in a given clinical situation. Moreover, the antibiotics recommended may not be limited to the one with the best utility: other antibiotics with a high utility, but slightly lower than the best one, might be considered as recommended too. Consequently, in a given clinical situation, we consider as recommended all antibiotics having a utility highest to the best utility found in this clinical situation minus t_{pref}, a second threshold that must be learned.

In the example, Table 4 shows the learned weights and thresholds. The resulting $f(x_i)$ are given in the right part of Table 5. For example, for case x_4, $f(x_4) = False \times 0.2 + True \times 0.3 = 0.1$ (we remind that $False = -1$ and $True = 1$).

3.4. Determining antibiotics recommended by the preference model

For each clinical situation $\mathcal{S}_k \in \mathcal{S}$, the antibiotics recommended by the preference model can be determined in three steps. First, we compute $\mathcal{S}_{k\,cand}$, the set of candidate cases that can be prescribed with regards to necessary features:

$$\mathcal{S}_{k\,cand} = \left\{ x_i \in \mathcal{S}_k \mid \nexists j \in F_{ness}, x_{i,j} \neq 1 \right\} \quad (5)$$

If $\mathcal{S}_{k\,cand} = \emptyset$, no antibiotic is recommended by the preference model and the process stops here.

Second, we determine x_{best}, the case ranked the highest by the utility function:

$$x_{best} \in \mathcal{S}_k \text{ such as } f(x_{best}) = \max(f(x_i) \mid x_i \in \mathcal{S}_{k\,cand}) \quad (6)$$

Third, we compute $\mathcal{S}_{k\,reco}$, the set of cases classified as recommended, and including all candidate cases from $\mathcal{S}_{k\,cand}$ for which the utility function is higher than (or equal to) the best value found previously minus the threshold t_{pref}:

$$\mathcal{S}_{k\,reco} = \left\{ x_i \in \mathcal{S}_{k\,cand} \mid f(x_i) \geq f(x_{best}) - t_{pref} \right\} \quad (7)$$

In the example, the right part of Table 5 indicates which cases belong to $\mathcal{S}_{k\,cand}$ and $\mathcal{S}_{k\,reco}$. The example has two clinical situations X and Y. There is a best utility value $f(x_{best})$ for each clinical situation (shown in bold in Table 5). Then, for each clinical situation, we determine recommended cases using formula 7. For example, in situation X, x_4 has the best utility, and is therefore recommended. x_3 has a lower utility than x_4, but not lower than the utility x_4 minus $t_{pref} = 0.5$. Thus, x_3 is recommended too. On the contrary, x_2 has a utility lower than the utility x_4 minus $t_{pref} = 0.5$, and therefore is not recommended.

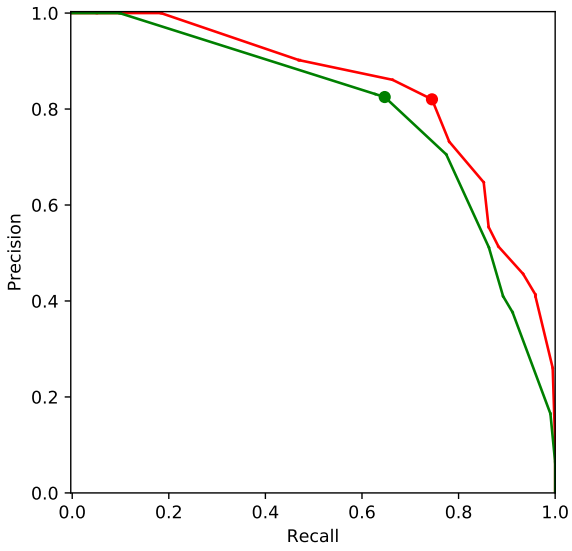

Figure 2: The PR curve (in green for model $\mathcal{M}_1$ and in red for model $\mathcal{M}_{any}$; the dots corresponds to $z = 1$).

3.5. Optimizing weights and thresholds

In order to learn necessary features and preferences, we need to find the optimal values for the weights $w_1, ..., w_p$ and the two thresholds t_{ness} and t_{pref}. The values of weights and thresholds were searched between 0 (the feature has no importance at all) and 1 (the feature has the maximal importance). The optimal weights and thresholds should minimize the total number of errors E, *i.e.* the number of recommended cases classified as non-recommended (false negatives) and the number of non-recommended cases classified as recommended (false positive) by the preference model:

$$E = \sum_k z \times |\mathcal{Y}_{reco} \cap (\mathcal{S}_k \setminus \mathcal{S}_{k\,reco})| + (2-z) \times |\mathcal{Y}_{not\,reco} \cap \mathcal{S}_{k\,reco}| \quad (8)$$

where z is a coefficient that allows giving more importance to either type of errors (false negatives or false positives). By changing the value of z, it is possible to obtain different results in terms of precision and recall, and to produce the PR curve (precision - recall).

This is an unconstrained global non-linear optimization problem with a solution space of $p + 2$ dimensions. We used the Artificial Feeding Birds (AFB) metaheuristics [48, 49] for solving the problem. The system has been implemented in Python 3 and executed with PyPy 3, a Python interpreter that includes a JIT (Just-In-Time) compiler. The cases were extracted from the OWL ontology. We used two open source Python module: Owlready for ontology-oriented programming [69], and Metaheuristic_Optimizer for optimization with AFB.

Figure 2 shows the PR curve. For the rest of the analysis, we fixed z to 1 (*i.e.* both types of errors have the same weight). We performed 20 runs of the optimization program and we stopped it after 15,000 solutions were tested. We kept the best result found. For $\mathcal{M}_1$, we obtained $E = 50$ (14 false positives and 36 false negatives), leading to a sensitivity/recall of 64.7%, a precision of 82.5%, and a specificity of 99.6%. For $\mathcal{M}_{any}$, we obtained $E = 82$ (32 false positives and 50 false negatives), leading to a sensitivity/recall of 75.5%, a precision of 82.0%, and a specificity of 99.0% (corresponding to the dots on the PR curve).

Case	x_i^a (antibiotic)	x_i^s (clinical situation)	*no contraindication*	*no side ef*	*protocol*	$\in \mathcal{Y}_{reco}$	$f(x_i)$	$\in \mathcal{S}_{k\,cand}$	$\in \mathcal{S}_{k\,reco}$
x_1	Drug A	Situation X	*False*	*False*	*False*		-		
x_2	Drug B	Situation X	*True*	*False*	*False*		-0.5	yes	
x_3	Drug C	Situation X	*True*	*True*	*False*	yes	-0.1	yes	yes
x_4	Drug D	Situation X	*True*	*False*	*True*	yes	**0.1**	yes	yes
x_5	Drug E	Situation X	*Unknown*	*True*	*True*		-		
x_6	Drug A	Situation Y	*Unknown*	*False*	*False*		-		
x_7	Drug B	Situation Y	*True*	*False*	*False*		-0.5	yes	
x_8	Drug C	Situation Y	*True*	*True*	*False*		-0.1	yes	
x_9	Drug D	Situation Y	*True*	*False*	*True*	yes	0.1	yes	yes
x_{10}	Drug E	Situation Y	*True*	*True*	*True*	yes	**0.5**	yes	yes

Table 5: Example of Table 3 with the values of the utility function $f(x_i)$ that are computed during preference learning, and indicating which cases belong to $\mathcal{S}_{k\,cand}$ and $\mathcal{S}_{k\,reco}$.

		$\mathcal{M}_1$			$\mathcal{M}_{any}$		
j	**Feature**	N_j	w_j	**Category**	N_j	w_j	**Category**
1	*naturally active*	3.9%	-	Necessary	8.2%	-	Necessary
2	*probably active*	10.8%	-		15.3%	-	
3	*proved*	1.0%	-		0.5%	-	
4	*no contraindication*	3.9%	-		3.6%	-	
5	*protocol*	51.0%	0.26	Preference	52.0%	0.97	Preference
6	*not precious*	41.2%	0.73		55.1%	0.16	
7	*no side ef*	42.2%	0.47		49.5%	0.25	
8	*ef f icacy level*	33.3%	0.0		40.8%	0.026	
9	*spect*	84.3%	0.0		89.8%	0.092	
10	*low eco risk*	81.4%	0.2		86.7%	0.052	
11	*taste*	100.0%	0.71		100.0%	0.98	
		$t_{ness} = 15\%$	$t_{pref} = 0.015$		$t_{ness} = 20\%$	$t_{pref} = 0.77$	

Table 6: Results of the learning process (N_j, learned weights and thresholds, for $z = 1$) for the two preference models $\mathcal{M}_1$ and $\mathcal{M}_{any}$.

Table 6 shows the results of the learning process, for $z = 1$, for the two preference models $\mathcal{M}_1$ and $\mathcal{M}_{any}$ (*i.e.* considering only cases recommended as first-line treatment or as any line of treatment, see section 3.1). 4 necessary features are found in both models: *naturally active*, *probably active*, *proved* and *no contraindication*. The seven others are preference features. For determining the first-line treatments (R_1 with model $\mathcal{M}_1$), the most important preference features are *not precious*, *taste* and *no side ef*. On the contrary, for determining treatments that are recommended in any ranks ($R_1 \cup R_2 \cup R_3 \cup R_4$ with model $\mathcal{M}_{any}$), the most important preference features are *taste* and *protocol*. Indeed, *taste* is more important than what we may expect, because many clinical situations involve children and children refuse to take drugs with a bad taste. Two features, *ef f icacy level* and *spect*, have no impact for determining the first-line treatments and a very low impact for determining the recommended treatments in any ranks.

4. Detection of inconsistencies in CPGs

4.1. Methods

The 106 errors obtained during preference learning can be seen as *candidate inconsistencies* in CPGs: for those cases, the recommendation of the CPGs does not match the result obtained using the preference model we built from the whole guideline recommendations. Therefore, our hypothesis is that there might be inconsistencies in the CPGs for those cases.

All these candidate inconsistencies were manually verified by a medical expert (RT). For each candidate, the inconsistency was confirmed if the medical expert retrieved arguments within CPGs in contradiction with the level of recommendation given in CPGs. If no contradictions were found within CPGs, then the medical expert searched for contradictions in the argumentation of other guidelines (*e.g.* previously published CPGs, CPGs from other countries). Then the medical expert categorized the inconsistencies into various groups.

An example of contradiction follows. The CPG [70] says "the proof of efficacy of fosfomycin in cystitis with risk of complication (including pregnancy) is not sufficient", but, surprisingly, the CPG nevertheless recommends fosfomycin for the treatment of cystitis in pregnant women.

4.2. Medical analysis of the candidate inconsistencies

Table 7 summarizes the various categories of candidate inconsistencies obtained from preference learning and the number of cases in each category. The expert validated 51.9% (55/106) of the candidates as inconsistencies in CPGs. The medical expert detected 29 contradictions related to the features *probably active*, 11 to *proved*, 9 to *no contraindication*, 4 to *spect*, and 2 related to a mix of several preference features.

Two categories of contradictions were distinguished:

(1) For 40 cases, CPGs recommend an antibiotic whereas there are arguments in the same or another CPG for not recommending this antibiotic. For example, the CPG [57] recommends sulfamethoxazole/trimethoprim in child sinusitis, but the CPG also states "because of the evolution of frequency of acquired resistance, sulfamethoxazole/trimethoprim is not recommended in sinusitis".

Categories	$\mathcal{M}_1$	$\mathcal{M}_{any}$	Total *
Inconsistencies in CPGs			
Related to *probably active*	11	29	29
Related to *proved*	6	8	11
Related to *no contraindication*	5	9	9
Related to *spect*	4	0	4
Related to several features	2	0	2
Flaws in the knowledge base			
Missing features	0	9	9
Precision of the coding	5	3	7
End of drug commercialization	0	1	1
Flaws in the preference model			
Missing principles	3	7	7
Genericity of the model	2	9	9
Not categorized	12	7	18
Total	50	82	106 *

Table 7: Categorization of the candidate inconsistencies obtained from preference learning. (*) The "total" column is not the sum of the two previous columns, because 26 inconsistencies were discovered twice, once with the $\mathcal{M}_1$ model and once with the $\mathcal{M}_{any}$ model.

(2) For 15 cases, CPGs do not recommend a given antibiotic, whereas there are arguments in the same or another CPG for recommending this antibiotic in similar clinical situation. For example, in otitis with conjunctivitis, CPG [57] do not recommend amoxicillin and prefers the combination of amoxicillin/clavulanic acid; this choice is justified as follows: "Amoxicillin/clavulanic acid should be preferred because of a strong suspicion of H. influenzae infection". On the contrary, in otitis without conjunctivitis, CPG [57] recommends amoxicillin rather than the amoxicillin/clavulanic acid combination and justifies this choice as follows: "Amoxicillin should be preferred because (...) it is active against H. influenzae strains".

The expert classified 16.0% (17/106) of the candidate inconsistencies as caused by flaws in the antibiotic knowledge base. Three categories of flaws were distinguished:

(1) For 9 cases, CPGs consider additional features that were missing in the knowledge base: bioavailability (*i.e.* antibiotic oral absorption), minimum inhibitory concentration (*i.e.* the lowest antibiotic concentration which prevents the growth of bacterium), and marketing authorization. For example, the preference model recommends levofloxacin in uncomplicated cystitis in women, but the argumentation of CPG [70] says " levofloxacin is not recommended because it does not have the marketing authorization" for this clinical situation in France.

(2) For 7 cases, the coding of features as Boolean is not precise enough. The two features involved are *no side ef* and *not precious*. For these cases, a more fine-grained gradation in at least three levels (*e.g.* none/moderate/high) is required. For example, the preference model recommends pristinamycin and telithromycin for pneumonia in children over the age of 12 years with betalactam allergy. Pristinamycin is recommended in CPG, but not telithromycin [54]. The argumentation of guidelines says that both may cause side effects, but telithromycin is associated with side effects more serious than pristinamycin. This information is missing in the knowledge base because values are coded as Boolean.

(3) For 1 case, the antibiotic recommended by the preference model is not marketed any more. The antibiotic should be removed from the knowledge base.

The expert classified 15.1% (16/106) of the candidate inconsistencies as caused by flaws in the preference model. Two categories of flaws were distinguished:

(1) For 7 cases, additional principles not considered in the model seemed to be used in CPGs. Two such principles were identified: (a) When several antibiotics are recommended, CPGs often propose antibiotics belonging to different therapeutic classes, because some patients have drug class allergy (*e.g.* penicillin allergy), and thus it is better to propose a wide variety of drug classes. (b) In a very few number of cases where no antibiotics have the necessary features, CPGs recommend some antibiotics without one of the necessary features (*probably active*).

(2) For 9 cases, the preference model seems different for children than from adult patient, whereas we hypothesized that a generic model exists for all patients and diseases (hypothesis #2 in introduction). More specifically, for these 9 cases, CPGs give more importance to side effects for children. For example: In adult sinusitis, CPGs recommends pristinamycin whereas in child, CPG [57] says "pristinamycin is not recommended because of the risk of side effects". However, no side effect specific to children (*e.g.* developmental disorders) were mentioned.

Finally, 17.0% (18/106) of the candidate inconsistencies could not be categorized by the medical expert.

5. Discussion and conclusion

In this paper, we showed that strategies followed by CPGs for establishing therapeutic recommendations in antibiotherapy can be formalized by a preference model. This model can be learned from the CPGs and a knowledge base describing the domain. In addition, the model allowed the detection of 106 candidate inconsistencies in CPGs, 55 of which were then validated by a medical expert. We also identified several flaws in the knowledge base. To the best of our knowledge, this is the first study that proposes using preference learning on drug properties for detecting inconsistencies within CPGs.

5.1. Discussion on the preference model and the learning process

Our preference model is based on a quantitative approach. It uses a simple utility function, which has the interest of producing results easily understandable by medical experts and physicians. In addition, the method we proposed allowed the learning of a single model from several clinical situations, each situation including several cases, and the various features can depend on the antibiotic, the patient profile, the infectious disease and/or the causative bacteria. As a consequence, the feature values can be assigned at various levels in the knowledge base. We used a formal ontology in OWL 2.0 for managing the knowledge base and for obtaining the feature values for each case, using inheritance. Despite the specificity of each infectious disease and patient profile, we successfully built a generic model satisfying all clinical situations described in CPGs, with the exception of 9 cases related to children.

Our utility function does not consider interactions between features, while in antibiotherapy, there might be some interactions. For example, two features a and b might have an impact on the utility, while the impact of $a \cup b$ is not the sum of the impact of a and b alone. More sophisticated methods, such as Choquet integral [71], can capture the dependencies between features. In the future, we will aim to study this point.

The proposed method is interesting since it allowed learning not only preferences, but also necessary features (*i.e.* constraints) that are very important in preference reasoning. In the literature, few models exist that permit modeling both constraints and preferences [72]. For example, let us consider an antibiotic

	AFB	ABC	FA
Mean	**83.60**	86.95	131.4
Best	**82.00**	**82.00**	118.0
# best	**7**	2	0
Std. deviation	1.319	6.119	6.946
Time	25.6s	25.6s	25.8s

Table 8: Performance comparison between three metaheuristics: Artificial Feeding Bird (AFB), Artificial Bee Colony (ABC) and Firefly Algorithm (FA). # best is the number of times the best known result (82.0) was found, over the 20 runs.

having *protocol* feature True but *no contradiction* False. If the learned preference model considers only preference features (*e.g.* *no contradiction* is preferred to *protocol*) as in most proposed learning preferences methods, then it is possible to recommend this antibiotic. However, in our proposed model, it cannot be recommended since the model contains also necessary features or constraints (such as *no contradiction*) which can be viewed as a propositional formula.

For optimizing weights and thresholds, we reused the AFB metaheuristics. We chose metaheuristics in general because they are known to adapt quite well to many kinds of problems, whatever the size of the solution space is. Furthermore, in the literature, metaheuristics have already been used for preference learning [73]. AFB was chosen since it results from our previous works, and because it was found to be especially adaptable, and quite independent from parameter values.

AFB seems efficient for the problem presented here. We compared it with two other metaheuristics, Artificial Bee Colony (ABC) [74] and Firefly Algorithm (FA) [75], for the learning of the $\mathcal{M}_{any}$ model with $z = 1$. We performed 20 runs for each algorithm. For each run, the optimization process was stopped after 15,000 solutions were tested. Table 8 shows the mean results for each algorithm. AFB yielded the best results, and found the lowest value (82.0) in 7 runs. ABC found the lowest value in only 2 runs. On the contrary, FA never found the lowest value. Therefore, AFB seems an interesting algorithm for preference learning.

We used the default parameter values for AFB. We performed parameter optimization *a posteriori*, and we obtained better results with slightly different values: $p_2 = 0.0049$, $p_3 = 0.78$ and $p_4 = 0.068$. With those values, the mean result was 82.9 (instead of 83.6), and the best result was obtained 11 times out of 20 runs (instead of 7 times). However, the parameter optimization took half a day, thus it may not worth the time to perform it since the gain is low. This confirms previous results suggesting that AFB has a low sensitivity to parameter values.

5.2. Medical discussion of results

The learning process allowed the identification of two types of features: necessary *vs* preference. This confirms the results of previous works based on the manual analysis of CPGs [28, 29]. The distinction between necessary and preference features allows classifying antibiotics in 3 categories, for a given clinical situation: (1) the inappropriate antibiotics that should never be prescribed because they are not efficient for treating the infectious disease or they cannot be used for a given patient (*i.e.* antibiotics not having the necessary features), (2) the appropriate antibiotics that could be prescribed, but that are not the ones recommended because better antibiotics exist (*i.e.* antibiotics having the necessary features, but not preferred), and (3) the recommended antibiotics (*i.e.* antibiotics having the necessary features, and preferred). This classification into 3 categories could be helpful in clinical practice.

In addition, the learning process identified the weight of antibiotic features. Not surprising, the feature *protocol* has an important weight. Indeed, in clinical practice, physicians often prefer to prescribe drugs with short treatment duration and/or single daily dose to increase patient observance. The feature *not precious* has also an important weight for antibiotic recommended in first line (model $\mathcal{M}_1$). This feature is specific to antibiotherapy where the objective is to rationalize the use of antibiotics because of bacteria resistance. Thus, some antibiotics are "preserved", and their use is restricted to specific clinical situations. The weight of the feature *no side ef* may seem lower than what we may expect. However, this can be explained because infectious diseases often require short treatments (one to 7 days), and thus the probability to develop side effects is lower than for longer treatments. Surprisingly, the features *spect* and *low eco risk* have a low weight. Since a current French campaign [76] encourages the use of antibiotics with narrow spectrum and low ecological risk, we expected higher weights for these features.

The preference learning on the antibiotic knowledge base allowed the detection of 106 errors. The analysis of these errors by a medical expert leads to the identification of 55 inconsistencies in CPGs (defined as contradictions between arguments retrieved in CPGs and the level of recommendation given in CPGs). Therefore, about half of the candidate inconsistencies were manually validated as inconsistencies in the CPGs. These inconsistencies could correspond to two subcategories: (i) either the level of recommendation given in CPGs is correct, but the arguments retrieved in CPGs are in contradiction, or (ii) the arguments retrieved in CPGs are correct, but the level of recommendation given in CPGs is in contradiction. Distinguishing between those two subcategories would require to gather all the experts who wrote the CPGs, which is hardly feasible.

5.3. Perspectives

In the future, we will aim at improving our approach in various ways. The missing antibiotic features (*e.g.* bioavailability) and medical principles identified during the medical analysis could be added to the knowledge base and the preference model, respectively. We would also like to test our approach in other medical domains. However, this would require a preliminary work by medical experts to identify the drug features involved in the recommendations, and the constitution of a knowledge base for representing the domain. The constitution of this knowledge base may be a long process: for antibiotherapy, it took several months and 5 rounds of Delphi Process. The extraction of the content of the knowledge base was not automated in the presented work, although the design of an automatic process is an interesting perspective.

It would be interesting to test a qualitative model, such as conditional preferences [44], in addition to the quantitative preference model proposed in this paper. We would like to check whether it would allow the detection of additional inconsistencies.

Another perspective is the design of a tool based on our preference model and the knowledge base, for supporting experts during the writing of CPGs. The tool could help experts to detect inconsistencies in their recommendations before CPGs publications, but could also suggest a list of recommended antibiotics for each clinical situation. This could speed up the development process of CPGs, which is currently too long [5].

Furthermore, these findings might be used for designing a clinical decision-support system for helping physicians to prescribe antibiotics [77]. Indeed, the preference model could be used for

suggesting antibiotics when no recommendation exists in CPGs. Further works are needed to confirm this hypothesis.

Competitive interest statement

None.

Acknowledgements

This work was supported by the French drug agency (ANSM, Agence Nationale de Sécurité du Médicament et des produits de santé) through the RaMiPA project AAP 2016.

References

[1] D. L. Sackett, S. E. Straus, W. S. Richardson, W. Rosenberg, R. B. Haynes, Evidence-based medicine: How to practice and teach EBM, Elsevier/Churchill Livingstone, Edinburgh, 2000.

[2] H. Woolf S, R. Grol, A. Hutchinson, M. Eccles, J. Grimshaw, Clinical guidelines: potential benefits, limitations, and harms of clinical guidelines, BMJ (Clinical research ed.) 318 (7182) (1999) 527–30.

[3] R. Goud, A. Hasman, A. M. Strijbis, N. Peek, A parallel guideline development and formalization strategy to improve the quality of clinical practice guidelines, Int J Med Inf 78 (8) (2009) 513–20. doi:10.1016/j.ijmedinf.2009.02.010.

[4] M. Tierney W, M. Overhage J, Y. Takesue B, E. Harris L, D. Murray M, L. Vargo D, J. McDonald C, Computerizing guidelines to improve care and patient outcomes: the example of heart failure, J Am Med Inform Assoc 2 (5) (1995) 316–22.

[5] H. Woolf S, G. DiGuiseppi C, D. Atkins, B. Kamerow D, Developing evidence-based clinical practice guidelines: lessons learned by the US Preventive Services Task Force, Annu Rev Public Health 17 (1996) 511–38.

[6] M. Peleg, V. L. Patel, V. Snow, S. Tu, C. Mottur-Pilson, E. H. Shortliffe, R. A. Greenes, Support for guideline development through error classification and constraint checking, Proceedings. AMIA Symposium (2002) 607–11.

[7] D. Cabana M, S. Rand C, R. Powe N, W. Wu A, H. Wilson M, A. Abboud P, R. Rubin H, Why don't physicians follow clinical practice guidelines? A framework for improvement, JAMA 282 (15) (1999) 1458–65.

[8] E. A. McGlynn, S. M. Asch, J. Adams, J. Keesey, J. Hicks, A. DeCristofaro, E. A. Kerr, The quality of health care delivered to adults in the United States, The New England journal of medicine 348 (26) (2003) 2635–45.

[9] T. M. Shaneyfelt, M. F. Mayo-Smith, J. Rothwangl, Are guidelines following guidelines? The methodological quality of clinical practice guidelines in the peer-reviewed medical literature, JAMA 281 (20) (1999) 1900–5.

[10] A. Cluzeau F, P. Littlejohns, M. Grimshaw J, G. Feder, E. Moran S, Development and application of a generic methodology to assess the quality of clinical guidelines, International journal for quality in health care : journal of the International Society for Quality in Health Care 11 (1) (1999) 21–8.

[11] Development and validation of an international appraisal instrument for assessing the quality of clinical practice guidelines: the AGREE project, Quality & safety in health care 12 (1) (2003) 18–23.

[12] U. Siering, M. Eikermann, E. Hausner, W. Hoffmann-Eßer, E. A. Neugebauer, Appraisal tools for clinical practice guidelines: a systematic review, PloS one 8 (12) (2013) e82915. doi:10.1371/journal.pone.0082915.

[13] A. ten Teije, M. Marcos, M. Balser, J. van Croonenborg, C. Duelli, F. van Harmelen, P. Lucas, S. Miksch, W. Reif, K. Rosenbrand, A. Seyfang, Improving medical protocols by formal methods, Artif Intell Med 36 (3) (2006) 193–209.

[14] M. Peleg, Computer-interpretable clinical guidelines: A methodological review, Journal of Biomedical Informatics 46 (4) (2013) 744–763.

[15] W. Miller D, J. Frawley S, L. Miller P, Using semantic constraints to help verify the completeness of a computer-based clinical guideline for childhood immunization, Comput Methods Programs Biomed 58 (3) (1999) 267–80.

[16] A. Galopin, J. Bouaud, S. Pereira, B. Séroussi, Using an ontological modeling to evaluate the consistency of clinical practice guidelines: application to the comparison of three guidelines on the management of adult hypertension, Stud Health Technol Inform 205 (2014) 38–42.

[17] G. Duftschmid, S. Miksch, Knowledge-based verification of clinical guidelines by detection of anomalies, Artif Intell Med 22 (1) (2001) 23–41.

[18] N. Shiffman R, A. Greenes R, Improving clinical guidelines with logic and decision-table techniques: application to hepatitis immunization recommendations, Medical decision making : an international journal of the Society for Medical Decision Making 14 (3) (1994) 245–54.

[19] Shiffman R N, Representation of clinical practice guidelines in conventional and augmented decision tables, J Am Med Inform Assoc 4 (5) (1997) 382–93.

[20] P. Hammond, J. Sergot M, C. Wyatt J, Formalisation of safety reasoning in protocols and hazard regulations, Proceedings. Symposium on Computer Applications in Medical Care (1995) 253–7.

[21] S. Wilk, A. Fux, M. Michalowski, M. Peleg, P. Soffer, Using constraint logic programming for the verification of customized decision models for clinical guidelines, in: Artificial Intelligence in Medicine - 16th Conference on Artificial Intelligence in Medicine, AIME 2017, Vienna, Austria, June 21-24, 2017, Proceedings, 2017, pp. 37–47.

[22] S. Wilk, M. Michalowski, W. Michalowski, D. Rosu, M. Carrier, M. Kezadri-Hamiaz, Comprehensive mitigation framework for concurrent application of multiple clinical practice guidelines, Journal of Biomedical Informatics 66 (2017) 52–71.

[23] L. Sacchi, S. Rubrichi, C. Rognoni, S. Panzarasa, E. Parimbelli, A. Mazzanti, C. Napolitano, S. G. Priori, S. Quaglini, From decision to shared-decision: Introducing patients' preferences into clinical decision analysis, Artificial Intelligence in Medicine 65 (1) (2015) 19–28.

[24] K. Sedki, C. Duclos, J. B. Lamy, A Preference-based framework for medical decision making, Stud Health Technol Inform 205 (2014) 63–7.

[25] L. Sacchi, S. Rubrichi, C. Rognoni, S. Panzarasa, E. Parimbelli, A. Mazzanti, C. Napolitano, S. G. Priori, S. Quaglini, From decision to shared-decision: Introducing patients' preferences into clinical decision analysis, Artif Intell Med 65 (1) (2015) 19–28.

[26] E. Parimbelli, S. Quaglini, C. Napolitano, S. Priori, R. Bellazzi, J. Holmes, Use of Patient Generated Data from Social Media and Collaborative Filtering for Preferences Elicitation in Shared Decision Making, in: Proceedings of AAAI, 2014, pp. 35–38.

[27] V. Delcroix, K. Sedki, F. X. Lepoutre, A Bayesian network for recurrent multi-criteria and multi-attribute decision problems: Choosing a manual wheelchair, Expert Systems with Applications 40 (7) (2013) 2541–2551.

[28] R. Tsopra, A. Venot, C. Duclos, An algorithm using twelve properties of antibiotics to find the recommended antibiotics, as in CPGs , in: AMIA Annu Symp Proc, Vol. 1115-24, 2014.

[29] R. Tsopra, A. Venot, C. Duclos, Towards evidence-based CDSSs implementing the medical reasoning contained in CPGs: application to antibiotic prescription, in: Stud Health Technol Inform, Vol. 205, 2014, pp. 13–7.

[30] J. Frnkranz, E. Hllermeier, Preference Learning, 1st Edition, Springer-Verlag New York, Inc., New York, NY, USA, 2010.

[31] E. Hüllermeier, J. Fürnkranz, W. Cheng, K. Brinker, Label ranking by learning pairwise preferences, Artif. Intell. 172 (16-17) (2008) 1897–1916.

[32] S. Vembu, T. Gärtner, Label ranking algorithms: A survey, in: Preference Learning., 2010, pp. 45–64.

[33] W. W. Cohen, R. E. Schapire, Y. Singer, Learning to order things, CoRR abs/1105.5464. arXiv:1105.5464.

[34] S. Clémençon, N. Vayatis, Ranking the best instances, Journal of Machine Learning Research 8 (2007) 2671–2699.

[35] C. Bergeron, J. Zaretzki, C. M. Breneman, K. P. Bennett, Multiple instance ranking, in: Machine Learning, Proceedings of the Twenty-Fifth International Conference, ICML 2008, Helsinki, Finland, June 5-9, 2008, 2008, pp. 48–55.

[36] W. Waegeman, B. De Baets, A survey on roc-based ordinal regression learning, in: J. Fürnkranz, E. Hüllermeier (Eds.), Preference learning, Springer, 2010, pp. 127–154.

[37] T. Joachims, L. Granka, B. Pan, H. Hembrooke, G. Gay, Accurately interpreting clickthrough data as implicit feedback, in: SIGIR'05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, ACM Press, New York, NY, USA, 2005, pp. 154–161.

[38] O. Dekel, C. D. Manning, Y. Singer, Log-linear models for label ranking, in: NIPS, MIT Press, 2003, pp. 497–504.

[39] C. J. C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, G. N. Hullender, Learning to rank using gradient descent, in: ICML, Vol. 119 of ACM International Conference Proceeding Series, ACM, 2005, pp. 89–96.

[40] K. Dembczynski, W. Kotlowski, R. Slowinski, M. Szelag, Learning of rule ensembles for multiple attribute ranking problems, in: Preference Learning, Springer, 2010, pp. 217–247.

[41] B. Jiang, J. Pei, X. Lin, D. W. Cheung, J. Han, Mining preferences from superior and inferior examples, in: KDD, ACM, 2008, pp. 390–398.

[42] S. de Amo, M. S. Diallo, C. T. Diop, A. Giacometti, D. H. Li, A. Soulet, Contextual preference mining for user profile construction, Inf. Syst. 49 (2015) 182–199.

[43] Y. Chevaleyre, F. Koriche, J. Lang, J. Mengin, B. Zanuttini, Learning ordinal preferences on multiattribute domains: The case of cp-nets, in: Preference Learning, Springer, 2010, pp. 273–296.

[44] C. Boutilier, R. I. Brafman, C. Domshlak, H. H. Hoos, D. Poole, Cp-nets: A tool for representing and reasoning with conditional ceteris paribus preference statements, CoRR abs/1107.0023.

[45] U. Chajewska, D. Koller, D. Ormoneit, Learning an agent's utility function by observing behavior, in: Proceedings of the Eighteenth International Conference on Machine Learning, ICML '01, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2001, pp. 35–42.

[46] A. Y. Ng, S. J. Russell, Algorithms for inverse reinforcement learning, in: Proceedings of the Seventeenth International Conference on Machine Learning, ICML '00, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2000, pp. 663–670.

[47] A. J. Ramirez, B. H. C. Cheng, Automatic Derivation of Utility Functions for Monitoring Software Requirements, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 501–516.

[48] Lamy JB, Les oiseaux picorant artificiels : une nouvelle méta-heuristique inspirée du comportement des pigeons, in: Actes des Journées d'Intelligence Artificielle Fondamentale (JIAF), Caen, 2017.

[49] Lamy JB, Advances in nature-inspired computing and applications, Vol. under press, Springer, 2018, Ch. Artificial Feeding Birds (AFB): a new metaheuristic inspired by the behavior of pigeons.

[50] Lones MA, Metaheuristics in nature-inspired algorithms, in: Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation, Vancouver, BC, Canada, 2014, pp. 1419–1422.

[51] Yang XS, Nature-inspired metaheuristic algorithms (second edition), Luniver Press, 2010.

[52] AFSSAPS, Guidelines - diagnostic et antibiothérapie des infections urinaires bactériennes communautaires du nourrisson et de l'enfant (2007).

[53] AFSSAPS, Guidelines - diagnostic et antibiothérapie des infections urinaires bactériennes communautaires chez l'adulte (2008).
[54] ANSM, Guidelines - Antibiothérapie par voie générale dans les infections respiratoires basses de l'adulte et de l'enfant - Recommandations et Argumentaires (2005).
[55] AFSSAPS, SPILF, SPLF, Guidelines - antibiothérapie par voie générale dans les infections respiratoires basses de l'adulte (2010).
[56] AFSSAPS, Guidelines - antibiothérapie par voie générale en pratique courante dans les infections respiratoires hautes de l'adulte et de l'enfant (2005).
[57] SPILF SFP, Guidelines - Antibiothérapie par voie générale en pratique courante dans les infections respiratoires hautes de l'adulte et de l'enfant (2011).
[58] HCSP, Guidelines - rapport relatif à la conduite à tenir devant un ou plusieurs cas de coqueluche (2008).
[59] AFSSAPS, Guidelines - mise au point traitement antibiotique probabiliste des urétrites et cervicites non compliquées (2008).
[60] European Society of Clinical Microbiology and Infectious Diseases, Guidelines for the management of adult lower respiratory tract infections (2011).
[61] IDSA, Guidelines - international clinical practice guidelines for the treatment of acute uncomplicated cystitis and pyelonephritis in women: A 2010 update by the infectious diseases society of america and the european society for microbiology and infectious diseases (2011).
[62] AFSSAPS, Guidelines - fiches de transparence médicaments anti-infectieux en pathologies communautaires (2004).
[63] AFSSAPS, Guidelines - livret médicaments et grossesse (2005).
[64] AFSSAPS, Guidelines - spectre d'activité antimicrobienne (2005).
[65] HAS, Guidelines - stratégie d'antibiothérapie et prévention des résistances bactériennes en établissement de santé (2008).
[66] ANSM, Guidelines - caractérisation des antibiotiques considérés comme critiques (2013).
[67] SPILF, CMIT, SNMInf, FFI, ECN - PILLY [internet]. [cited 2017 jan 21]. available from: http://www.infectiologie.com/fr/ecnpilly.html (2017).
[68] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, P. L. Patel-Schneider, The description logic handbook: theory, implementation and applications, Cambridge University Press, 2007.
[69] Lamy JB, Owlready: Ontology-oriented programming in Python with automatic classification and high level constructs for biomedical ontologies, Artif Intell Med 80 (2017) 11–28.
[70] SPILF, Guidelines - Diagnostic et antibiotherapie des infections urinaires bactériennes communautaires de l'adulte (2015).
[71] A. F. Tehrani, W. Cheng, E. Hullermeier, Preference learning using the Choquet integral: The case of multipartite ranking (2012).
[72] S. Benferhat, K. Sedki, Two alternatives for handling preferences in qualitative choice logic (2008).
[73] L. Jinyan, W. Ouerdane, V. Mousseau, A Methaheuristic approach for preference Learning in multi criteria ranking based on reference points., in: Workshop from multiple criteria Decision aid to Preference Learning (DA2PL), 2014.
[74] Karaboga D, An idea based on honey bee swarm for numerical optimization, Technical report.
[75] Yang XS, Firefly algorithms for multimodal optimization, Stochastic Algorithms: Foundations and Applications - Lecture Notes in Computer Sciences 5792 (2009) 169–178.
[76] ANSM, Guidelines - Liste des antibiotiques critiques - Rapport (2016).
[77] R. Tsopra, J. P. Jais, A. Venot, C. Duclos, Comparison of two kinds of interface, based on guided navigation or usability principles, for improving the adoption of computerized decision support systems: application to the prescription of antibiotics, J Am Med Inform Assoc.