MEDINFO 2021: One World, One Health – Global Partnership for Digital Innovation
P. Otero et al. (Eds.)

doi:10.3233/SHTI220157

A Qualitative Method for Learning Medical Expert Reasoning

Karima Sedki[a], Jean-Baptiste Lamy[a], Rosy Tsopra[b,c,d]

[a] Université Sorbonne Paris Nord, LIMICS, INSERM, UMR 1142, F- 93000, Bobigny, France
[b] INSERM, Université de Paris, Sorbonne Université, Centre de Recherche des Cordeliers Information Sciences to support Personalized Medicine, F-75006 Paris, France
[c] Inria Paris, 75012 Pari
[d] Department of Medical Informatics, Hôpital Européen Georges-Pompidou, AP-HP, Paris, France

Abstract

The aim of this paper is to propose a qualitative method for learning a model that represents the closest possible experts reasoning and strategies to provide recommendations of antibiotics. The learned model contains an integrity constraint and a preference formula. The former indicates the features that an antibiotic should have to be recommended. The later indicates the rank of recommendation of an antibiotic.

Keywords:

Preferences learning, Experts reasoning, CPGs.

Introduction

Clinical practice guidelines (CPGs) are textual documents written by a group of experts according to scientific evidence. They contain recommendations and their justifications. In the domain of antibiotic prescription, we analyzed these justifications and showed that experts used antibiotic features to recommend or not antibiotics [7, 8]. These features are grouped in two categories: necessary features and preference ones. 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. Preference features allows to indicate the rank of recommendation (1st, 2nd, 3rd and 4th line of treatment) of recommended antibiotics.

The aim of this paper is to start from a database containing the antibiotics characterized by their features and their rank of recommendations and learn a preference model that represents the closest experts reasoning to provide recommendations of antibiotics in primary care. The idea of preference learning [4] is to learn a preference model from observed preference information. For example, in our studied problem, we hypothesize that we could learn the preference model used by experts for recommending antibiotics by extracting preference formulas and integrity constraints on the features of antibiotics. Once the preference model is learnt, it could be used for various purposes (e.g. treatment predictions). Methods of preference learning can be quantitative or qualitative. Quantitative approaches [2,7] consist mainly to learn a utility function on training data. For qualitative approaches [3,5], the objective is to learn a binary preference relation that compares each pair of objects.

Our proposed method consists to learn a qualitative model that contains two types of formulas: integrity constraints defined on the necessary features, and a preference formula defined on the antibiotics' preference features. This paper extends the proposition given in [6] where the new model is more expressive and it integrates integrity constraints built on the necessary features of antibiotics.

Methods

We used a database containing the antibiotics, their features, and their rank of recommendations as defined in CPGs, for several diseases. This database was validated by antibiotic experts according to a Delphi process [7]. Let $V = \{v_1, v_2, ..., v_{11}\}$ be the set of features of antibiotics of the database D. For each feature $v_i \in V$, we denote the domain of v_i by $dom(v_i)=\{1=True, 0=False\}$. We note that there are missing values that are considered as false values in our method (precaution principle). There are 5 recommendation ranks in the knowledge base D: recommended antibiotics in (1^{st}, 2^{nd}, 3^{rd} and 4^{th}) line of treatment ($R_1=1$, $R_2=2$, $R_3=3$, $R_4=4$ respectively) or not recommended antibiotics ($R_5 = 0$). Antibiotics having R_1 are preferred to those having R_2 which are also preferred to those having R_3 and so on. Antibiotics having $R_5 = 0$ are not preferred since that are not recommended. See simplified example in Table 1.

Table 1– Simple example of the database with 7 antibiotics and 5 features

Antibiotics	v_1	v_2	v_3	v_4	v_5	R
a_1	1	1	1	1	1	1
a_2	1	1	1	1	0	3
a_3	0	1	1	0	1	1
a_4	0	0	1	0	0	0
a_5	1	1	0	1	1	2
a_6	1	1	0	1	0	3
a_7	0	1	0	0	0	2

Learning a preference model from the antibiotics database : In [7], the authors showed 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 integrity constraints). Preference features indicate the rank of recommendation of an antibiotic. 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. We define V_{ness} and V_{pref} , the sets of necessary and preference features, respectively. Table 2 gives the list of necessary and preference features as shown in [7].

Table 2– Necessary and preference features (description of features can be found in [7]).

Feature Name	Category
naturaly active	Necessary
probably active	Necessary
proved	Necessary
noContraindication	Necessary
protocol	Preference
not precious	Preference
side eff	Preference
efficacy level	Preference
spect	Preference
eco risk	Preference
Taste	Preference

The learned model contains an integrity constraint defined on the necessary features and a preference formula defined on the preference features. Before detailing our method, we first give the following definitions.

***Definition* 1 (Single and Conjunctive options)**. - *Each single option is defined on the set of features* V. - *Each conjunctive option is defined on the set of features* V *and the logical connective (¬ and ∧).*

- *We define* Sat(a, x) *a function that returns* T rue *if anti- biotic* a *satisfies option* x *and* unSat(a, x) *a function that returns* False *if antibiotic* a *does not satisfy option* x. *The set of options (single or disjunctive) is denoted by* X

***Definition* 2 (Integrity constraint).** *Each integrity constraint is defined on* V_{ness} *and the logical connective (¬ and ∧). Each single or conjunctive option that is defined on* V_{ness} *is an integrity constraint. The set of integrity constraints is denoted by* Const.

If we consider $V_{ness}=\{v_1, v_2\}$, then v_1, $\neg v_2$, $\neg v_1 \wedge v_2$ are examples of integrity constraints.

***Definition* 3 (Preference formula)**. *A preference formula is a partial order in the form of* $x_1 > x_2 > x_3 > x_4$ *where* x_i *is a single or conjunctive option defined on* V_{pref}.

Each preference formula contains 4 options (4 corresponds to the greatest recommendation rank of the database). Note that x_1 is the first preferred option of the preference formula, x_2 is the second preferred one, etc. Namely, antibiotics that satisfy x_1 (or having x_1 true, even if $x_{i=2...4}$ are true or false) have rank 1, antibiotics that satisfy x_2 and falsify x_1 have rank 2, etc. If we consider $V_{pref}=\{v_5 \ldots v_{11}\}$, then $\varphi = (v_5 \wedge v_6 \wedge v_7) > v_9 > (v_{10} \wedge v_{11}) > v_8$ is an example of a preference formula. As we aim to learn a best preference formula, we have to define what are the best options ($x_{i=1...4}$) to be appeared in φ.

Thus, we aim to learn a model M that contains an integrity constraint ψ and a preference formula φ defined as follows.

***Definition* 4**. *Let* D *be the database,* a *an antibiotic in* D *and* $R=\{R_1=1, R_2=2 \ldots R_5=0\}$ *and* $\forall a \in D$, $R(a) \in R$, then

$$\mathcal{M} = \begin{cases} \psi \quad s.t. \quad \psi \in Const \\ \phi = x_1 \succ x_2 \succ x_3 \succ x_4 \\ s.t. \quad x_{i=1...4} \in X \end{cases}$$

Learning preference formulas : The learned preference formula is in the form of $\varphi = x_1 > x_2 > x_3 > x_4$ where x_i is a single or a conjunctive option defined on V. In the following, we present our proposed method (Algorithm 1) for generating best options of φ. Our method is inspired from Apriori algorithm [1] for generating frequent item-sets. Instead of generating all possible options which can be very large (there are 2^7 - 1 possible options that can be generated on V_{pref}), we generate only frequent ones which correspond to those exceeding a minimal fixed support and confidence. The idea is that all antibiotics with rank $R_1=1$ should satisfy the option x_1 (even if x_2, x_3 and x_4 are satisfied or not) and all antibiotics with rank $R_2=2$ (resp. $R_3=3, R_4=4$, $R_5=0$) should not satisfy the option x_1. So, x_1 of the learned preference formula should be the option that is satisfied by the maximum number of antibiotics having the rank $R_1=1$ (Ideally all antibiotics, but in real applications, the database is often inconsistent), and falsified by the maxi- mum number of antibiotics with rank more than 1 (ideally all). The same reasoning for learning the best options x_2, x_3 and x_4. Thus, our definition of support and confidence are given as follows.

***Definition* 5 (Support).** *Let* $R=\{R_1=1, R_2=2 \ldots R_4=4, R_5=0\}$. *The support of an option* x *for antibiotics* a *having a recommendation rank* $R(a) = R_i$ *is defined as :*

$$Supp(x, R_i) = \frac{|\{a \mid Sat(a,x) \wedge R(a) = R_i\}|}{|\{a \mid R(a) = R_i\}|}$$

The support of an option x for a rank R_i is defined by the fraction of the number of antibiotics $a \in D$ having a rank $R(a) = R_i$ that satisfy x on the number of antibiotics having rank R_i. Its interest increases with its support (ideally 1). Even if the support of x is high for R_i, it is possible that its support for $R_{j \neq i}$ be also high, so there is need to compute its confidence.

***Definition* 6 (Confidence).** *The confidence of an option* x *for antibiotics* $a \in D$ *having a rank* $R(a) = R_i$ *is defined as* :$Conf(x,R_i)=$

$$\frac{|\{a \mid Sat(a,x) \wedge R(a) = R_i) \vee (a \mid unSat(a,x) \wedge R(a) = R_{i+1})\}|}{|\{a \mid R(a) = R_{i \ldots 5}\}|}$$

The confidence of an option x for rank R_i indicates the proportion of antibiotics that keep their initial rank if x appears in the learned formulas. An option is interesting for a rank R_i if its confidence is high (ideally 1).

***Example* 1.** *Let us consider data of Table 1 and assume that* V_{pref} = *{*v_3, v_4, v_5*}. Given the option* $v_3 \wedge v_4$, *we have* $Supp(v_3 \wedge v_4, R_1)$=*1/2 and* $Conf(v_3 \wedge v_4, R_1)$=*5/7 which means that this option is not the best for* x_1 *of the preference formula that we aim to learn.*

Generating best options defined on V_{pref}

For generating best options, we adapt the approach of association rules [1]. The idea is to start with all single options, count their support and find all single frequent options, combine them to form *candidate* 2-conjunctive options, go through data and count their support and find all *frequent* 2-conjunctive options, combine them to form *candidate* 3- conjunctive options and so on. Once frequent options are generated for each rank $R_{i\neq 0}$, we return only those exceeding a minimal confidence θ, called best options ($Best_{R_i}$). Algorithm 1 summarizes these steps.

The set C_1 for $R_1 \in R$

x	$Supp(x, R_1)$
v_3	1
$\neg v_3$	0
v_4	0.5
$\neg_4$	0.5
v_5	1
$\neg v_5$	0

The set F_1 for R_1

x	$Supp(x, R_1)$
v_3	1
v_5	1

The set C_2 for R_1

x	$Supp(x, R_1)$
$v_3 \wedge v_5$	1

The set F_2 for R_1

x	$Supp(x, R_1)$
$v_3 \wedge v_5$	1

F_{R_1}=$F_1 \cup F_2$

x	$Supp(x, R_1)$	$Conf(x, R_1)$
v_3	1	0.71
v_5	1	0.85
$v_3 \wedge v_5$	1	1

$Best_{R_1}$ with $\theta = 0.9$

x	$Supp(x, R_1)$	$Conf(x, R_1)$
$v_3 \wedge v_5$	1	1

Figure 1– Best options for R_1

It is not necessary to generate best options for rank 0 since these later do not appear in the learned preference formula.

Algorithm 1 Best options defined on V_{pref}

Input : The database D, R= {R_1=1, R_2=2…R_4=4, R_5=0}, a minimal support σ and a minimal confidence θ.
Output : Best options for each rank $R_{i=1...4}$ with support and confidence exceeding σ and θ ($Best_{R_{i=1...4}}$).
C_j :Candidate options of size j from V_{pref}.
F_j :Frequent options of size j.
F_1 :Frequent single options.
F_{R_i} :Frequent options for rank $R_{i=1...4}$.
for *each* $R_{i=1...4} \in R$ **do**
  for $j = 1; j \neq \emptyset; j++$ **do**
    C_{j+1} :candidate options generated from F_j
    for *each antibiotic a having* R_i **do**
      for *each* $x \in C_{j+1}$ **do**
        Compute $Supp(x, R_i)$
      end
    end
    F_{j+1}={$x \in C_{j+1} \mid Supp(x, R_i) \geq \sigma$}
  end
  $F_{R_i} = \cup_j F_j$
  $Best_{R_i} = \emptyset$
  for *each* $x \in F_{R_i}$ **do**
    if $Conf(x, R_i) \geq \theta$ **then**
      $Best_{R_i}$ =$Best_{R_i} \cup x$
    end
  end
  Return $Best_{R_i}$
end

However, antibiotics with R_i=0 are considered for computing the support and confidence of any option. Note that depending on the minimal support and confidence, it is possible to have more than one best option for a given rank and then more than one preference formula and preference model. However, we evaluate the quality of each learned model by computing its accuracy (definition 10).

***Example* 2.** *Let us consider data of Table 1 and suppose that* V_{pref} =*{*v_3, v_4, v_5*}. Figure 1 gives frequent single and conjunctive options for rank* R_1=*1 exceeding minimal support* σ = 0.7. *In Figure 1, we have* C_1=*{*v_3, ¬v_3, v_4, ¬v_4, v_5, ¬v_5*}. On the basis on the fixed minimal support, from* C_1, *we obtain* F_1=*{* v_3, v_5*}. From* F_1, *we obtain the set* C_2=*{*$v_3 \wedge v_5$*}. From* C_2, *we have* F_2 = *{*$v_3 \wedge v_5$*}. Then,* F_{R_1} =$F_1 \cup F_2$=*{*v_3, v_5,$v_3 \wedge v_5$*}.With minimal confidence* θ=0.9,$Best_{R_1}$=*{*$v_3 \wedge v_5$*}. Thus, the best option for* x_1 *of the learned formula is equal to* $v_3 \wedge v_5$.

Learning integrity constraints

For learning integrity constraint, we group all recommended antibiotics together whatever their recommendation rank. Thus, we will have antibiotics with rank R (recommended) and antibiotics with rank NR (not recommended). To learn best integrity constraint, we first compute the sup- port and confidence of each option from V_{ness} for all antibiotics with rank R.

***Definition* 7 (Support)**. *The support of an option* x *for antibiotics* a∈D *having a recommendation rank* R(a) = R *is defined as :*

$$Supp(x, R) = \frac{|\{a \mid Sat(a, x) \wedge R(a) = R\}|}{|\{a \mid R(a) = R\}|}$$

The support of an option x for a rank R is defined by the fraction of the number of antibiotics a ∈ D having a rank R that satisfy x on the number of antibiotics having rank R. Its interest increases with its support (ideally 1). Even if the support of x is high for R, it is possible that its support for NR be also high, so there is need to compute its confidence.

Definition 8 (Confidence). The confidence of an option x is defined as: Conf (x, R)=

$$\frac{|\{a \mid Sat(a, x) \wedge R(a) = R) \vee (a \mid unSat(a, x) \wedge R(a) = NR)\}|}{|D|}$$

As said above, an integrity constraint is a single or a conjunctive option defined on V_{ness}, so to learn a best integrity constraint we adapt Algorithm 1 for this purpose (see Algorithm 2).

Algorithm 2 Best integrity constraints defined on V_{ness}

Input : The database D, R= {R, NR}, a minimal support σ and a minimal confidence θ.
Output : Best integrity constraint with support and confidence exceeding σ and θ ($BestConstraint$).
C_j :Candidate options of size j from V_{ness}.
F_j :Frequent options of size j.
F_1 :Frequent single options.
F_R :Frequent options for rank R.
for $j = 1; j \neq \emptyset; j++$ **do**
  C_{j+1} :candidate options generated from F_j
  for *each antibiotic a having* R **do**
    for *each* $x \in C_{j+1}$ **do**
      Compute $Supp(x, R)$
    end
  end
  F_{j+1}={$x \in C_{j+1} \mid Supp(x, R) \geq \sigma$}
end
$F_R = \cup_j F_j$
$BestConstraint$=$\emptyset$
for *each* $x \in F_R$ **do**
  if $Conf(x, R) \geq \theta$ **then**
    $BestConstraint$ =$BestConstraint \cup x$
  end
end
Return $BestConstraint$

In order to evaluate the quality of the learned model, there is need to verify :i) if each antibiotic satisfies the learned integrity constraint and ii) if the rank inferred by the learned preference formula is the same for each antibiotic.

***Definition* 9.** *Let* M *be the learned model that contains the preference formula* $\varphi = x_1 > x_2 > x_3 > x_4$ *and the integrity constraint* ψ. *The rank inferred by* M *for antibiotic* a *is* Inf(M,a)= k *iff*

Sat(a, ψ), and Sat(a,$x_1 \vee x_2 \vee x_3 \vee x_4$) and k=min(j|Sat(a,$x_j$))

The rank inferred by M for antibiotic a is Inf(M,a)= 0 iff

unSat(a, ψ), or unSat(a, $x_1 \vee x_2 \vee x_3 \vee x_4$).

Definition 9 states that given a preference formula $\varphi = x_1 > x_2 > x_3 > x_4$ and an integrity constraint ψ, the rank of antibiotic a inferred by φ is equal to k, if a satisfies ψ and the kth option of φ (i.e. x_k) is true and the preceding ones (x_1, x_2 ..., x_{k-1}) are false. If a does not satisfy ψ, or no option x_k of φ is true, then the rank of a is equal to 0. We introduce the accuracy measure as follows:

Definition 10 (Accuracy). Given a database D and a learned model M. The accuracy of M is:

$$Accuracy(\mathcal{M}) = \frac{|\{a \in D \mid Inf(\mathcal{M}, d) = R(a)\}|}{|D|}$$

Experimental results

This section presents experimental results of our method in the domain of antibiotic prescription. Examples of the learned models for pharyngitis are given in Figure 2. Note that more than one preference model are learned because there are many best options with the same support and confidence for each rank. In addition, each learned model contains the same integrity constraint which is equal to naturallyActive ∧ probablyActive ∧ proved ∧ noContraindication. This means that if an antibiotic does not satisfy this constraint, it should not be prescribed and thus it is not recommended. Exemples of best options of R_1=1 are (Proto∧Precious∧¬SideEff∧Efficacy∧Spect), (Proto∧¬RiskResi) and (Proto∧Spect) with σ=1and θ=1. Best options for R_2=2 is (Proto ∧ ¬SideEff) with σ = 1 and θ = 0.7. Best options for R_3=3 are (Proto ∧ Efficacy), (Proto ∨ Taste) and Proto with σ = 1 and θ = 0.35.

Learned preference models	Accuracy
$\{naturallyActive \wedge probablyActive \wedge proved \wedge noContraindication$ $(Proto \wedge Precious \wedge \neg SideEff \wedge Efficacy \wedge Spect \wedge \neg RiskResi) \succ (Proto \wedge \neg SideEff) \succ (Proto \wedge Efficacy)$	0.91
$\{naturallyActive \wedge probablyActive \wedge proved \wedge noContraindication$ $(Proto \wedge \neg SideEff \wedge Efficacy \wedge Spect \wedge \neg RiskResi) \succ (Proto \wedge \neg SideEff) \succ (Proto \wedge Efficacy)$	0.91
$\{naturallyActive \wedge probablyActive \wedge proved \wedge noContraindication$ $(Proto \wedge Efficacy \wedge Spect \wedge \neg RiskResi) \succ (Proto \wedge \neg SideEff) \succ (Proto \wedge Efficacy)$	0.91
$\{naturallyActive \wedge probablyActive \wedge proved \wedge noContraindication$ $(Proto \wedge Efficacy \wedge Spect) \succ (Proto \wedge \neg SideEff) \succ (Proto \wedge Efficacy).$	0.91
$\{naturallyActive \wedge probablyActive \wedge proved \wedge noContraindication$ $(Proto \wedge Spect \wedge \neg RiskResi) \succ (Proto \wedge \neg SideEff) \succ (Proto \wedge Efficacy)$	0.91
$\{naturallyActive \wedge probablyActive \wedge proved \wedge noContraindication$ $Proto \wedge Spect) \succ (Proto \wedge \neg SideEff) \succ (Proto \wedge Efficacy)$	0.91
$\{naturallyActive \wedge probablyActive \wedge proved \wedge noContraindication$ $(Proto \wedge \neg RiskResi) \succ (Proto \wedge \neg SideEff) \succ (Proto \wedge Efficacy)$	0.91

Figure 2–The Learned preference model in pharyngitis. Note that in this case, the greatest recommendation rank is equal to 3. So, the length of the obtained preference rules is equal to 3.

Conclusion

We proposed a qualitative method for learning preferences of CPG experts from both recommendations and a data- base containing the antibiotics and their features. The learned preference model is powerful since it represents experts reasoning and strategies to provide recommendations. The proposed method is more general than quantitative methods and the learned preferences are qualitative and more expressive. It could be improved on several points: for example, visualizing the preference model to be more readable and explainable for the user.

Acknowledgements

This work was supported by Agence Nationale de Sécurité du Médicament et des Produits de santé (ANSM). [AAP 2016-RaMiPA Project - Reasoning for a better antibiotic prescription - Dr Rosy TSOPRA].

References

[1] Rakesh Agrawal, Tomasz Imielin´ski, and Arun Swami. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, SIGMOD '93, pages 207–216, New York, NY, USA, 1993. ACM.

[2] Christopher J. C. Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Gre- gory N. Hullender. Learning to rank using gradient descent. In *ICML*, volume 119 of *ACM Internatio*nal Conference Proceeding Series, pages 89–96. ACM, 2005.

[3] Sandra de Amo, Mouhamadou Saliou Diallo, Cheikh Talibouya Diop, Arnaud Giacometti, Do- minique Haoyuan Li, and Arnaud Soulet. Contextual preference mining for user profile construction. *Inf. Syst.*, 49 :182–199, 2015.

[4] Johannes Frnkranz and Eyke Hllermeier. *Preference Learning*. Springer-Verlag New York, Inc., New York, NY, USA, 1st edition, 2010.

[5] BinJiang,JianPei,XueminLin,DavidW.Cheung,and Jiawei Han. Mining preferences from superior and inferior examples. In *KDD*, pages 390–398. ACM, 2008.

[6] Sedki K, Lakrafli C, Lamy JB, and Tsopra R. An approach based on preference learning for identifying experts reasoning in antibiotic treatment. In *Stud Health Technol Inform*, pages 115–118, 2020 Jun 26.

[7] Rosy Tsopra, Jean-Baptiste Lamy, and Karima Sedki. Using preference learning for detecting inconsistencies in clinical practice guidelines : Methods and application to antibiotherapy. *Artificial Intelligence in Medicine*, 89 :24–33, 2018.

[8] Rosy Tsopra, Karima Sedki, Mélanie Courtine, Hector Falcoff, Antoine De Beco, Ronni Madar, Frédé- ric Mechaï, and Jean-Baptiste Lamy. Helping gps to extrapolate guideline recommendations to patients for whom there are no explicit recommendations, through the visualization of drug properties. the example of antibiohelp® in bacterial diseases. *JAMIA*, 26(10) :1010–1019, 2019.

Address for correspondence

Karima Sedki.
karima.sedki@univ-paris13.fr