Summarizing Conceptual Descriptions using Knowledge Representations

Sébastien Harispe*, Jacky Montmain, Massissilia Medjkoune
École des mines d'Alès – LGI2P/EMA Research Centre,
Parc scientifique G. Besse, 30035 Nîmes, France
Email: {firstname.name}@mines.ales.fr

Abstract—**Summarizing a body of information is a complex task which mainly depends on the ability to distinguish important information and to condense notions through abstraction. Considering a knowledge representation partially ordering concepts into a directed acyclic graph, this study focuses on the problem of summarizing several human descriptions expressed through sets of concepts. We formally define the problem of summarization in this context and we propose a model mimicking a Human-like Intelligence for scoring alternatives with regard to a specific objective. Several interesting theoretical results related to this problem (e.g. for optimization) are also given. Finally, the evaluation of the proposed approach performed in the domain of odor analysis highlights the benefits of our proposal and shows how it could be used to automatize time-consuming expert summarizing processes. Source code implementing the proposed approach as well as datasets are made available to the community.**[1]

I. Introduction

The ability of abstracting specific observations by means of distinguishing general patterns or concepts is central for the emergence of complex conceptual processes, e.g. learning. It is indeed one of the essential abilities defining intelligent agents that are able to analyze potentially complex and faintly related situations to acquire knowledge from these analyses [1], [2]. Abstraction is tightly linked to the process of analyzing observations through the lens of structured categories defined w.r.t. specific properties of the observations. This ability human have explains for instance that any child burning himself with a wood-fired oven will most often extract knowledge from this experiment and learn that the wood is not the cause of the unfortunate experiment, but rather the heat caused by its combustion. Therefore, thanks to this capacity of abstraction, children will identify similar situations (e.g. a lighter flame) as potentially harmful and take extra care while being confronted to them. Similarly, if a person tells you that a specific candy smells `Lemon`, and another person that the same candy smells `Orange`, you'll naturally and easily be able to abstract these two descriptions by summarizing the provided information to `Citrus Fruit`. Studying processes related to abstraction is therefore of major importance for Artificial Intelligence and has been central to numerous areas of research related to this domain - machine learning among others [2]. Interestingly, a growing number of knowledge representations (e.g. ontologies or even taxonomies) today formally express specific domain-knowledge, and provide concept organizations that can be automatically analyzed. These knowledge representations, based on the notion of abstraction, therefore offer the opportunity to design Human-like intelligent procedure taking advantage of prior knowledge regarding concept organizations [3]. Mimicking Human-like intelligence able to process such knowledge representations to abstract knowledge in a meaningful way is however still a complex, important and open challenge.

In this study we focus on the problem of summarizing a set of conceptual descriptions expressed in the form of sets of concepts. The notion of concept here refers to the traditional notion of *concept* or *class* used in Knowledge Representation; concepts are assumed to be partially ordered with regard to specific properties they share – knowledge representations are domain-specific. Figure 1 shows an example of concept ordering for odor evaluation. It specifies for instance that the concept `Rose` refers to the concept `Floral`. Thanks to the ordering of concepts, intuitively, considering the two following sets of annotations corresponding to odor sample descriptions ({`Rose`, `Mint`}, {`Violet`, `Orange`}) several summaries could be proposed, e.g. {`Floral`, `Fresh`}, {`F-Class`}.

Even if summarizing is an intuitive process in which an evaluator wants to abstract as much information as possible while keeping the resulting abstraction meaningful and informative, the relevance of a summary is context-dependent and no formal consensual definition of *best* summary can be given - even if general constraints on the definition of a summary could be defined, as we will see in Section II. Indeed, considering aforementioned example, summarizing the conceptual descriptions by one of both summaries could make sense depending on how much we want the summary to be concise and on how much information loss we accept.

The paper is organized as follows: Section II introduces theoretical notions on which is based our approach as well as notations; it next formally defines the summarizing problem; Section III introduces the proposed model for evaluating summaries; Section III introduces algorithms enabling to use the model for searching for relevant summaries and discusses interesting properties of the search space; Section V presents results related to evaluation; State-of-the-art is proposed in Section VI. Finally, section VII summarizes our results and distinguishes perspectives.

[1] Source code and datasets are publicly available at https://github.com/sharispe/Conceptual-Summary.

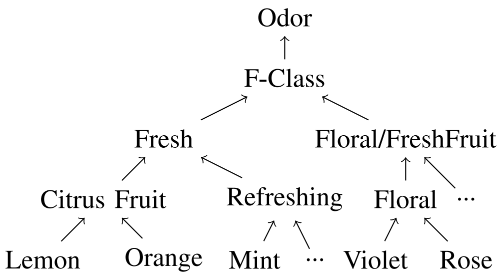

Fig. 1. Exemple of partial ordering of concepts related to odor evaluation

II. Problem setting: Formalization

Prior to discuss the technical aspects of our contribution and to formally define the notion of summary, we introduce some notations and notions on which is based our approach.

A. Notations and preliminary notions

In this study we consider an *a priori* domain knowledge on the form of a set of concepts C partially ordered into a poset $O = (\preceq, C)$. $\forall (x,y) \in C^2, x \prec y$ means that x implies y, e.g. in the context of odor evaluation it means that smelling `Rose` implies smelling a `Floral` compound – cf. to Figure 1.

To further ease the introduction of formal definitions, we define the following functions $\mathcal{D} : \mathcal{P}(C) \rightarrow \mathcal{P}(C)$, $\mathcal{A} : \mathcal{P}(C) \rightarrow \mathcal{P}(C)$, with $C' \subseteq C$:

$$\mathcal{D}(C') = \bigcup_{c \in C'} \bigcup_{c' \preceq c} \{c'\} \qquad \mathcal{A}(C') = \bigcup_{c \in C'} \bigcup_{c \preceq c'} \{c'\}$$

The commonly used notion of *Information Content* (IC) of concepts refers to concepts' degree of specificity; several models have been proposed. Some of them are based on the analysis of topological properties of the partial order (intrinsic models), sometimes mixed with additional prior knowledge w.r.t. concept usage (extrinsic models) [4]. For any IC function we have: $\forall (x,y) \in C^2, x \prec y \implies IC(x) > IC(y)$. An example of simple intrinsic IC formulation is [5]:

$$IC(c) = 1 - \frac{log(|\{x | x \preceq c\}|)}{|C|} \quad (1)$$

The IC function which is used hereafter refers to this expression - any intrinsic IC with $IC : C \rightarrow [0,1]$ could be used.

When occurrences over concepts are available, we consider the *mass function* $m : C \rightarrow [0,1]$, satisfying $\sum_{c \in C} m(c) = 1$; $m(c)$ corresponds to the number of observations of c among the total number of observations. The belief and plausibility functions $bel : C \rightarrow [0,1]$ and $pl : C \rightarrow [0,1]$ proposed in the Dempster–Shafer theory are next defined such as:[2]

$$bel(c) = \sum_{x \preceq c} m(x) \quad (2)$$

$$pl(c) = \sum_{x \in C, \mathcal{D}(x) \cap \mathcal{D}(c) \neq \emptyset} m(x) \quad (3)$$

B. Problem Formalization

The aim of the study is to summarize the information given by a set of evaluators (E) providing conceptual evaluations, such as each evaluator $e_n \in E$ is associated to a set of concepts $X_n \in \mathcal{P}(C)$. For convenience, we will always denote the sequence of annotations to summarize $\hat{X} = (X_1, X_2, \ldots, X_n)$, and X the set of concepts mentioned in the sequence of annotations $\hat{X}$, such as:

$$X = \bigcup_{i=1}^{n} X_i$$

The simplest type of summary function f can be defined such as $f : \hat{X} \rightarrow \mathcal{P}(C)$. Nevertheless, in accordance to the process of summarization in most contexts of use, we want any summary $Y \in \mathcal{P}(C)$ of a set of concepts X to respect specific properties:

1) *summarizing* $|Y| \leq |X|$, with most often $|Y| \ll |X|$
2) *faithful* $\forall y \in Y, \exists x \in X$ such as $x \preceq y$
3) *non-total redundancy* $\forall (x,y) \in Y^2, x \npreceq y \wedge y \npreceq x$.

We denote $\mathcal{S} \subseteq \mathcal{P}(C)$ the subsets of C respecting the *non-total redundancy* property. Furthermore we consider $\mathcal{S}^X \subseteq \mathcal{S}$ the set of summaries of a sequence of annotations $\hat{X}$ – each summary respects the properties of being *summarizing* and *faithful* w.r.t. X. Based on these preliminary definitions we formally define by S the function summarizing a sequence of n annotations $\hat{X} \in \mathcal{P}(C)^n$ by a single summary from $\mathcal{S}^X$:

$$S : \mathcal{P}(C)^n \rightarrow \mathcal{S}, \text{ with } S(\hat{X}) \in \mathcal{S}^X$$

We define the problem of summarizing a sequence of annotations $\hat{X}$ by finding $Y \in \mathcal{S}^X$, the *best* summary for $\hat{X}$. The following section introduces the model we use for defining function S. Additional properties of interest for characterizing the solution space $\mathcal{S}^X$ are given first.

A summary $Y \in \mathcal{S}^X$ is said to be covering a sequence of annotations $\hat{X}$ if all the concepts mentioned in $\hat{X}$ (i.e. all concepts of X) are abstracted by at least a concept of the summary: $\forall x \in X, \exists y \in Y$ such as $x \preceq y$. We denote the covering summaries of $\hat{X}$ by $\mathcal{S}^X_{cov}$. We also consider $\mathcal{S}^X_{p-cov} = \mathcal{S}^X \setminus \mathcal{S}^X_{cov}$ the set of summary *partially covering* X, i.e. $\forall Y \in \mathcal{S}^X_{p-cov}, \exists X' \subset X$ such as $Y \in \mathcal{S}^{X'}_{cov}$. Figure 2 illustrates the studied space exposed so far.

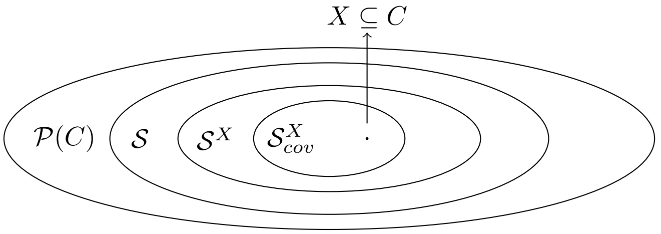

Fig. 2. Graphical representation of the studied space

As we mentioned, the notion of *best* summary is context-dependent. We can however distinguish some notions and quantities that could be used to evaluate the relevance of a summary and define the function S.

[2] We adopt the classical formalism used for defining the *mass*, *belief* and *plausibility* functions w.r.t knowledge representations. Note however that these definitions are not rigorous and are only used to lighten the formalism; they should be understood as the rigorous definitions proposed in [6].

III. Automatic summary: proposal

For studying the problem of finding a summary $Y \in \mathcal{S}^X$ for any $\hat{X} \in \mathcal{P}(C)^n$ we define the following objective function:

$$S(\hat{X}) = \underset{Y \in \mathcal{S}^X}{\arg\max} \left(\Psi(Y, \hat{X}) - \mathcal{L}(Y, \hat{X}) \right) \tag{4}$$

$$\mathcal{L}(Y, \hat{X}) = \Delta(Y, \hat{X}) + \lambda(Y) + \gamma(Y, \hat{X})$$

$\Psi(Y, \hat{X})$ models the amount of information from $\hat{X}$ covered by Y and $\mathcal{L}(Y, \hat{X})$ the penalty associated to the abstraction, with:

- $\Delta(Y, \hat{X})$ the penalty induced by abstracting by Y the information conveyed by $\hat{X}$ – incorporating penalties regarding *loss*, *addition* and *distortion* of information.
- $\lambda(Y)$ a function evaluating the conciseness of the summary - w.r.t redundancy of information.
- $\gamma(Y, \hat{X})$ a function that can be used to express additional constraints over Y, e.g. to exclude uncovering summaries.

The specific components of the objective function (eq. 4) are detailed hereafter.

A. $\Psi(Y, \hat{X})$ - Amount of abstracted information

We denote $\Psi(Y, \hat{X})$ the amount of information from $\hat{X}$ covered by Y. It is used to estimate the amount of conceptual information conveyed by $\hat{X}$ which is summarized by Y. We are therefore interested in studying the coverage of abstract notions among those mentioned by $\hat{X}$ and those mentioned by Y. Intuitively this quantity could be defined as follows:

$$\Psi(Y, \hat{X}) = f(\mathcal{A}(Y) \cap \mathcal{A}(X))$$

A specific expression could for instance be:

$$\Psi(Y, \hat{X}) = \sum_{c \in \mathcal{A}(Y) \cap \mathcal{A}(X)} w(c) \times IC(c)$$

with

$$w(c) = \sum_{c' \in X \cap \mathcal{D}(\{c\})} m(c')$$

with $w : C \to [0, 1]$ the function used to weigh the importance given to any concept w.r.t. to its degree of evocation according to the masses specified by $\hat{X}$ – recall that the elements of X are the focal elements, i.e. the concepts that have non null masses. In this case it corresponds to defining $w(c) = bel(c)$ with bel the classical belief function (Eq. 2) – this is the case since the focal elements are the members of X.[3] Since the more abstract a concept will be, the more it will be implicitly mentioned, and therefore easily summarized whatever the selected summary is, we regulate the importance given to any concept $c \in \mathcal{A}(Y) \cap \mathcal{A}(X)$ w.r.t its intrinsic information content.

Let's consider the following simple example in which $\hat{X} = (\{\text{Violet}, \text{Rose}\})$. In this case, considering the summaries $\{\text{Rose}\}$ (or $\{\text{Violet}\}$) and $\{\text{Floral}\}$ we would observe: $\Psi(\{\text{Floral}\}, \hat{X}) < \Psi(\{\text{Violet}\}, \hat{X})$. Otherwise stated the current model will consider that, based on the quantity of information from $\hat{X}$ which is abstracted, summarising $\hat{X}$ by $\{\text{Rose}\}$ is more relevant than using $\{\text{Floral}\}$. Putting aside the subjective notion of *best* summary, and avoiding arguing on which is the *best* summary, we want to penalize the fact that compared to $\{\text{Floral}\}$, $\{\text{Rose}\}$ only partially covers $\hat{X}$. In the current formulation we have $\Psi(\{\text{Rose}\}, \hat{X}) = \Psi(\{\text{Floral}\}, \hat{X}) + bel(Rose) \times IC(Rose)$. However, it can be counter-intuitive to consider that $\Psi(\{\text{Floral}\}, \hat{X})$ will contribute similarly to estimating the quantity of information from $\hat{X}$ abstracted by both $\{\text{Floral}\}$ and $\{\text{Rose}\}$. Otherwise stated, we would like to lower the contribution of $\Psi(\{\text{Floral}\}, \hat{X})$ while computing $\Psi(\{\text{Rose}\}, \hat{X})$ considering that not all the masses coming from specializations of Floral are subsumed by the summary $\{\text{Rose}\}$. Indeed reducing $\hat{X}$ to Rose suppresses the given information that $\hat{X}$ also mentioned Violet. This idea is related to the notion of *distortion* of information we will discuss later while defining the penalties associated to a summary. Nevertheless, we want to integrate this notion while estimating the quantity of information from $\hat{X}$ which is abstracted by a summary. To this end we propose to reconsider the definition of the weighting function as follows:

$$bel_Y(c \in C) = \sum_{c' \in X \cap \mathcal{D}(\{c\}) \cap \mathcal{D}(Y)} m(c')$$

We finally obtain:

$$\Psi(Y, \hat{X}) = \sum_{c \in \mathcal{A}(Y) \cap \mathcal{A}(X)} bel_Y(c) \times IC(c) \tag{5}$$

Note that $\mathcal{D}(X) \cap \mathcal{D}(Y)$ also conveys potentially useful information since $\mathcal{D}(X) \setminus \{X\}$ refers to the information not conveyed by X that are plausible; considering the partial ordering of concepts defined in Figure 1 someone referring to `Floral` could refer to `Violet` without being able to refer to this specific odor. It could therefore be interesting to study this quantity as it could be used to characterize the quantity of plausible information captured by the summary. Interesting properties could be achieved analysing this quantity since e.g. $\forall (Y, Y') \in \mathcal{S}^X_{cov} \times \mathcal{S}^X_{p-cov}, \mathcal{D}(Y') \cap \mathcal{D}(X) \subset \mathcal{D}(Y) \cap \mathcal{D}(X)$ could be used to favour covering summaries. Even if this quantity would not be useful for analysing covering summaries, i.e. *per* definition $\forall Y \in \mathcal{S}^X_{cov}$, $\mathcal{D}(X) \subseteq \mathcal{D}(Y)$, it could be worth considering it for discussing partially covering summaries. Nevertheless, since we consider the rational assumption that analysing exact information is more important than analysing plausible information while criticizing a summary, integrating this quantity in the definition of Ψ is not furthered explored in this paper; a refinement of the proposed approach could be $\Psi(Y, \hat{X}) = f(\mathcal{A}(Y) \cap \mathcal{A}(X), \mathcal{D}(Y) \cap \mathcal{D}(X))$.

$\Psi(Y, \hat{X})$ models the information conveyed by $\hat{X}$ which is conveyed by Y by considering covered masses. We now

[3] Note that we don't want the weighting function w to behave like an extrinsic information content, e.g. Resnik like [4]; since we don't want to give high importance to concepts that are marginal w.r.t $\hat{X}$ but very specific w.r.t any intrinsic IC formula.

introduce how we model the various components of the penalty factor $\mathcal{L}(Y, \hat{X}) = \Delta(Y, \hat{X}) + \lambda(Y) + \gamma(Y, \hat{X})$ (Eq. 4).

B. $\Delta(Y, \hat{X})$ – *Penalty of abstraction*

In the previous section we have defined a model for estimating the quantity of exact information conveyed by $\hat{X}$ which is conveyed by a summary. For criticizing the relevance of a summary it is also important to discuss penalties regarding *loss*, *addition* and *distortion* of information. We define the penalty of abstraction by:

$$\Delta(Y, \hat{X}) = f(\Delta^{E-}, \Delta^{P+}, \Delta^{P-}, \Delta^{D})$$

with

- Δ^{E-} penalty w.r.t to the deletion of exact info
- Δ^{P+} penalty w.r.t to the addition of plausible info
- Δ^{P-} penalty w.r.t to the deletion of plausible info
- Δ^{D} penalty w.r.t to distortion of information

We define those functions such as for each Δ function we have $\Delta(Y, \hat{X}) \in \mathbb{R}^{+}$.

Modelling Δ^{E-} and Δ^{P+}, Δ^{P-}:

Δ^{E-} models the amount of exact information conveyed by $\hat{X}$ which is not conveyed by Y – deletion of exact information:

$$\Delta^{E-}(Y, \hat{X}) = f(\mathcal{A}(X) \setminus \mathcal{A}(Y))$$

Δ^{P+} models the amount of plausible information conveyed by Y which is not conveyed by $\hat{X}$ – addition of plausible information; Δ^{P-} models the amount of plausible information conveyed by $\hat{X}$ which is not conveyed by Y – deletion of plausible information:

$$\Delta^{P+}(Y, \hat{X}) = f(\mathcal{D}(Y) \setminus \{\mathcal{D}(X) \cup \mathcal{A}(X)\})$$

$$\Delta^{P-}(Y, \hat{X}) = f(\mathcal{D}(X) \setminus \mathcal{D}(Y))$$

Note that *per* definition, and due to the property of *faithfulness*, a summary cannot (i) add exact information, i.e. provide information which is not conveyed by $\hat{X}$.[4] Considering aforementioned operators Δ^{E-} and Δ^{P+} (resp. Δ^{P-}), specific expressions can easily be obtained:

$$\Delta^{E-}(Y, \hat{X}) = \sum_{x \in \mathcal{A}(X) \setminus \mathcal{A}(Y)} (bel(x) \cdot IC(x))$$

$$\Delta^{P+}(Y, \hat{X}) = \sum_{y \in \mathcal{D}(Y) \setminus \{\mathcal{D}(X) \cup \mathcal{A}(X)\}} (pl(y) \cdot IC(y))$$

$$\Delta^{P-}(Y, \hat{X}) = \sum_{y \in \mathcal{D}(X) \setminus \mathcal{D}(Y)} (pl(y) \cdot IC(y))$$

Modelling Δ^{D} Penalty - distortion:

The aim of Δ^{D} is to penalize the distortion which is made considering a specific choice among partially-covering summaries. Considering the previously mentioned simple example in which $\hat{X} = (\{\text{Violet}, \text{Rose}\})$ with the summaries $\{\text{Rose}\}$ (or $\{\text{Violet}\}$) and $\{\text{Floral}\}$ we would have $\Delta^{E-}(\{\text{Rose}\}, \hat{X}) < \Delta^{E-}(\{\text{Floral}\}, \hat{X})$. Even if some adaption of Ψ have been proposed to penalize the bias induced by the choice of an uncovering summary, an additional penalty has to be modeled for considering additional potential distortion of information that could be made during summarizing. This penalty should be a function of $X \setminus \mathcal{D}(Y)$, i.e., all the elements of X that have not been summarized, and associated masses. We propose the following model to estimate the distortion.

$$\Delta^{D}(Y, \hat{X}) = \tau \sum_{x \in X \setminus \mathcal{D}(Y)} \sum_{x' \in \mathcal{A}(\{x\}) \setminus \mathcal{A}(Y)} (bel_{\{x\}}(x') \cdot IC(x'))$$

The parameter τ is used to weigh the importance of a specific uncovering and will be introduced later. For each uncovered concept in $x \in X \setminus \mathcal{D}(Y)$, the penalty associated to it is a function of its specificity and the specificity of the concepts abstracting x that are not covered by Y. We however want this penalty to be a function of the amount of masses associated to the concepts mentioned by $\hat{X}$ that are not covered. We want to penalize any distortion that is not motivated by very low masses associated to part that are excluded by the summary. To model this choice we use any f function ensuring $f(1) = 0$ and $f(0) = \infty$ or any very large value. Here we adopt the following function: $\tau = -\ln(1 - \beta_{YX}^{\alpha})$, with $\alpha \in \mathbb{N}^{*}$:

$$\beta_{YX} = \frac{\sum_{x \in X \setminus \mathcal{D}(Y)} m(x)}{\sum_{x \in X} m(x)} = \sum_{x \in X \setminus \mathcal{D}(Y)} m(x)$$

$\beta_{YX} \in [0, 1]$ represents the amount of masses relative to $\hat{X}$ that are not covered by Y – for any $Y \in \mathcal{S}_{cov}^{X}$ we will have $\beta_{YX} = 0$, $\beta_{YX} = 1$ for any $Y \in \mathcal{S} \setminus \mathcal{S}^{X}$. The more the distortion will be, the more β_{YX} tends to 1, and the more τ will induce an important penalty. The tuning parameter α is used to define the penalization ratio according to the loss of masses we accept – the lower α is, the more the model will penalize summaries implying mass losses.

We finally simply define:

$$\Delta(Y, \hat{X}) = \delta_{E-}\Delta^{E-} + \delta_{P+}\Delta^{P+} + \delta_{P-}\Delta^{P-} + \delta_{D}\Delta^{D} \quad (6)$$

With $\delta_{E-}, \delta_{P+}, \delta_{P-}, \delta_{D}$ input parameters used to set the importance of each abstraction penalty factor.

C. *Additional penalties to improve summarizing*

Modelling λ - Conciseness and redundancies penalties:

$\lambda(Y) \in \mathbb{R}^{+}$ is a penalty used to evaluate the conciseness of the summary - w.r.t the number of descriptors, by penalizing redundant information implicitly conveyed by a summary.

$$\lambda(Y) = \epsilon \sum_{y' \in \mathcal{A}(Y)} ((|\{y \in Y | y' \in \mathcal{A}(y)\}| - 1) \times IC(y'))$$

[4] Due to the definition of a summary we have no addition of exact information $\Delta^{E+}(Y, \hat{X}) = f(\mathcal{A}(Y) \setminus \mathcal{A}(X) = \emptyset)$. It could however be interesting to consider this quantity in specific contexts of use in which the definition of a summary would be less constraining than the one considered.

Using this expression, by avoiding large redundancies we favor abstraction and therefore conciseness, i.e. summaries that do not summarize enough the information carried by $\hat{X}$ will automatically be penalized. The penalization is designed such as each abstracted notions that are repeated more than once will be penalized the number of time the redundant information appears – taking into account of the intrinsic information content of concepts since redundancy cannot be avoided in most cases, and redundancies of very abstract concepts are of minor concern. Tuning ϵ can therefore be used to control the number of descriptors composing a summary.

γ - Additional constraints

$\gamma(Y, \hat{X})$ is a function that can be used to express additional constraints over Y. This constraints can be used to apply specific restrictions on the type of solution we are interested in, e.g. in particular if we relax the definition of a summary:

$$\gamma(Y, \hat{X}) = \begin{cases} 0 & \text{if } valid(Y, X) \\ +\infty & \text{otherwise} \end{cases}$$

with $valid : \mathcal{P}(C) \times \mathcal{P}(C) \rightarrow \{true, false\}$. As an example, exploring the covering summaries corresponds to defining the following *valid* function:

$$valid(Y, X) = \forall x \in X, \exists y \in Y \text{ such as } x \preceq y$$

Additional/Other constraints can naturally be defined, for instance, on the size of the summary we would like to generate or on the degree of specificity of the concepts it contains, i.e. by avoiding too abstract concepts.

Note that defining the γ function corresponds to reduce the solution space without explicitly defining a specially designed solution search algorithm. Indeed, by defining γ, restrictions on the solutions can be expressed while using the general search algorithm introduced in the following section.

The model proposed so far (Eq. 4) can be used to evaluate the relevance of a summary and rank several summary alternatives. By providing a summary $Y \in \mathcal{S}^X$ summarizing a sequence of annotation $\hat{X}$, knowing the importance of each concept of Y, i.e. how many sets of $\hat{X}$ (evaluators) evoke each concept, is of great importance for further data analysis. Therefore, considering the proposed setting, we can define the weight of any concept $y \in Y$ as a function of its belief. The next sections discusses details related to search space construction and analysis.

IV. Summary generation

In the previous section, we have introduced a framework for searching for relevant summaries considering $\hat{X}$ a sequence of conceptual annotations. This section now discusses algorithmic implications and discusses elements of information to practically define a strategy for exploring the search space $\mathcal{S}^X$. First let's recall some information related to the search space; the number of partitions of C is $2^{|C|}$ and in the worst case, considering that any knowledge representation always has an abstract concept generalizing all the others (a $root$), we have the following theoretical bound $\forall X \subset C, |\mathcal{S}^X| \leq 2^{|C-1|} + 1$. [5] However in practice, the theoretical bound is always far from the size of real search spaces thanks to the constraints defining a summary, in particular the *faithfulness* and the *non-total redundancy*. The size of $\mathcal{S}^X$ is nevertheless to be taken with high consideration since it largely impacts the computational time of the approach. Indeed, for any sequence of annotations $\hat{X}$ mentioning a large number of concepts (big $|X|$) applying a naive iterative search over $\mathcal{S}^X$ is not feasible. In this section we first propose an algorithm to construct $\mathcal{S}^X$. We then propose restrictions that can be applied on $\mathcal{S}^X$ to deal with sequence of annotations mentioning a large number of concepts.

A. Building $\mathcal{S}^X$

Considering a set of concepts $X \subset C$, Algorithm 1 defines how to generate $\mathcal{S}^X$, the set of summaries for $\hat{X}$. First recall that $\mathcal{S}^X = \mathcal{S}^X_{cov} \cup \mathcal{S}^X_{p-cov}$. Note also that:

$$\mathcal{S}^X_{p-cov} = \bigcup_{X' \in \mathcal{P}(X) \setminus X} \mathcal{S}^{X'}_{cov}$$

This enables to reformulate the problem of building $\mathcal{S}^X$ as finding the set $\mathcal{S}^X = \bigcup_{X' \in \mathcal{P}(X)} \mathcal{S}^{X'}_{cov}$. Given a sequence of annotations $\hat{X}$, Algorithm 1 computes $\mathcal{S}^X$. The algorithm uses a classical directed acyclic graph representation of $O = (\preceq, C)$, named $G = (E, C)$ with $E \subset C \times C$ and $(c_1, c_2) \in E$ means c_2 generalizes c_1, i.e. which corresponds to $c_1 \prec c_2$. For optimization reasons, G is expected to be reduced according to the transitivity of the relationships defining O (line 1) – otherwise stated there is no relationship in G that can be inferred according to the transitive relationships composing G. Thus for each subset $X' \in \mathcal{P}(X)$ the set S^X used to compute $\mathcal{S}^X$ is extended by adding the summaries covering X' that have not been found (line 4). Conceptually, this steps corresponds to $S^X = S^X \cup \mathcal{S}^{X'}_{cov}$, except that we don't want to cover any subset of $\mathcal{S}^{X'}_{cov}$ that have already been covered. Finally the algorithm returns $\mathcal{S}^X$ as S^X (line 6).

Algorithm 1 Generate summaries $\mathcal{S}^X \subseteq \mathcal{S}$ for $X \subset C$

```
1: transitive reduction $G$ – graph of $O = (\preceq, C)$
2: $S^X \leftarrow \emptyset$
3: for $X'$ in $\mathcal{P}(X)$ do
4:     $S^X = S^X \cup extend_{cov}(G, X', S^X)$
5: end for
6: return $\mathcal{S}^X$ as $S^X$
```

Given a set of concepts X, and a set of summaries $S_r \subset \mathcal{S}$ respecting the property that $\forall Y \in S_r, \mathcal{S}^Y_{cov} \subset S_r$, Algorithm 2 defines how to compute the subset of $\mathcal{S}^X_{cov}$ that have not been covered considering S_r. The constraint on S_r enables to use a greedy algorithm while computing $\mathcal{S}^X_{cov}$. Note that for a set of concepts X, defining $S_r = \emptyset$, Algorithm 2 computes $\mathcal{S}^X_{cov}$,

[5] The bound is reached when $O = (\preceq, C)$ is weakly structured and all the pairs composed of the other concepts except the $root$ only refer to non-ordered concepts – with $|C| = 100$ it means that the worst case is $|\mathcal{S}^X| = 6.3 \times 10^{29}$.

i.e. $\mathcal{S}^X_{cov} = extend_{cov}(G, X, \emptyset)$. The idea of the algorithm is simple; given a set of concepts X, it will recursively compute the covering summaries for each summary of X that can be obtained by substituting an element $x \in X$ by one of its parents – or any subset of its parents. Line 2 iterates over the element of X. Line 3 computes the parents of the current element c – recall that a transitive reduction has been applied on the graph. Next, the aim is to generate a summary of X by substituting c by one of its abstract representation (subset of parents) – line 3. Line 5 builds a summary of X by adding P'_c the selected subset of parents of x and by removing any concept from X that are generalized by a member of P'_c, e.g. x. This ensures to obtain a set of concepts respecting the *non-redundancy property*. The *faithfulness* property is ensured by the fact that any concept that is replaced is only replaced by one of its abstract representation. The *summarizing* property which ensures that we are processing a summary will next be tested in line 6. Before considering any summary we check if it hasn't been already processed (already in s^X_{cov}) and if it's not in the set of restricted summaries (in S_r). If this is not the case we add the current summary into the set of summaries, and, by applying a recursive procedure, we also add all summaries covering it by excluding any summary already encountered (line 7). Considering that the set $\mathcal{A}(c)$ is finite for each concept $c \in C$ and that in line 5, the set X' obtained is necessarily covering X, it ensures that the algorithm terminates.

Algorithm 2 $extend_{cov}$: Given a set of concepts X, a set of summaries $S_r \subset \mathcal{S}$ such as $\forall Y \in S_r, \mathcal{S}^Y_{cov} \subset S_r$ it generates $s^X_{cov} = \mathcal{S}^X_{cov} \setminus S_r$ with $s^X_{cov} \subseteq \mathcal{S}^X_{cov} \subseteq \mathcal{S}^X$. By defining $S_r = \emptyset$, the alorithm generates $\mathcal{S}^X_{cov}$.

```
 1: $s^X_{cov} \leftarrow \emptyset$
 2: for $c$ in $X$ do
 3:     $P_c = \{p | \exists (c, p) \in E\}$ // parents of $c$
 4:     for $P'_c$ in $\mathcal{P}(P_c)$ do
 5:         $X' = X \setminus \{x \in X | P'_c \cap \mathcal{A}(\{x\}) \neq \emptyset\} \cup P'_c$
 6:         if $|X'| < |X| \wedge X' \notin S_r \wedge \notin s^X_{cov}$ then
 7:             $s^X_{cov} = s^X_{cov} \cup \{X'\} \cup extend_{cov}(G, X', s^X_{cov} \cup S_r)$
 8:         end if
 9:     end for
10: end for
11: return $s^X_{cov}$
```

Considering the ordering of concepts introduced in Figure 3, Figure 4 presents the summaries $\mathcal{S}^X$ for $X = \{a, b, c, e\}$ as well as the partial ordering of $\mathcal{S}^X \cup \{X\}$. This ordering is built such as two groups of concepts $(Y, Y') \in \mathcal{S}^X$, are ordered such as $X \preceq_{\mathcal{S}^X} X'$ if $\mathcal{D}(X) \subseteq \mathcal{D}(X')$.

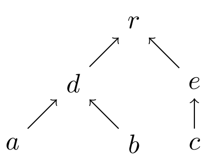

Fig. 3. $O = (\preceq, C)$ example of ordering of concepts

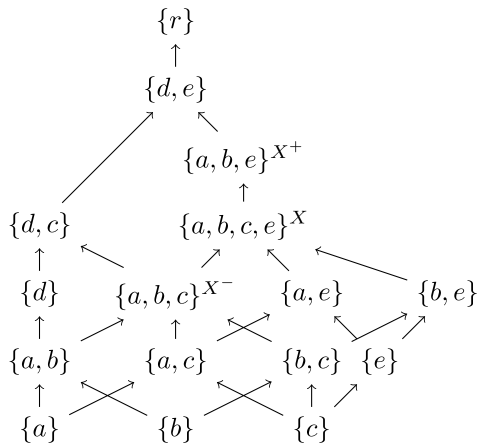

Fig. 4. Considering the set of concepts ordered in Figure 3, ordering of the set of summaries $\mathcal{S}^X$ that could be evaluated to summarize any sequence of annotations $\hat{X}$, with $X = \{a, b, c, e\}$, $X^+ = \{a, b, e\}$, $X^- = \{a, b, c\}$; the ordering also includes X.

Indeed, in addition to be the basis for computing S^X_{cov} and S^X, note also that Algorithm 2 can be used to compute $\preceq_{\mathcal{S}^X}$ the partial order of summaries of any $X \subseteq \mathcal{P}(C)$, i.e. the structure presented in Figure 4 for a specific example. Indeed, in line 5, each X' respecting the summarizing property refers to an ordering $X \preceq_{\mathcal{S}^X} X'$.[6]

For convenience, we denote $X^- \subseteq X$ and $X^+ \subseteq X$, the two largest subsets of X respecting the *non-total redundancy* such as $\forall x \in X^-$ (resp. X^+), $\nexists x' \in X, x' \neq x$ such as $x' \prec x$ (resp. $x \prec x'$). Considering Figure 3 and defining $X = \{a, b, c, e\}$ we would have $X^- = \{a, b, c\}$ and $X^+ = \{a, b, e\}$. We also use notations such as $\mathcal{S}^{X'} = \mathcal{S}^{X'}_{cov} \cup \mathcal{S}^{X'}_{p-cov}$, for any subset $X' \subseteq X$, e.g. X^-, X^+.

Interesting properties helping to better understand $\mathcal{S}^X$:

Property 1 - If X respects the *non-total redundancy* we have $X^- = X^+ = X$. More particularly we have $X^- \neq X \Leftrightarrow X^+ \neq X$ and $X^- = X \Leftrightarrow X^+ = X$ – this is easily proved using the respect or violation of *non-total redundancy*.
Property 2 - $(X, X') \in \mathcal{S}^2, X \preceq_{\mathcal{S}} X' \implies \mathcal{S}^{X'} \subseteq \mathcal{S}^X$. By definition of $\preceq_{\mathcal{S}}$, $X \prec_{\mathcal{S}} X' \implies \mathcal{D}(X) \subset \mathcal{D}(X')$. It implies that $A(X') \subset A(X)$. Knowing that $X' \in \mathcal{S}$ we have $X' \in \mathcal{S}^X$ and therefore $\mathcal{S}^{X'} \subseteq \mathcal{S}^X$.
Property 3 - $\mathcal{S}^{X^+} \subseteq \mathcal{S}^{X^-}$ – according to property 2.
Property 4 - $\mathcal{S}^X_{cov} = \mathcal{S}^{X^+}_{cov} \cup \{X^+\}$.
Property 5 - $\mathcal{S}^X_{p-cov} = \mathcal{S}^{X^-} \cup \{X^-\} \setminus \mathcal{S}^{X^+}_{cov}$.
Property 6 - $\mathcal{S}^{X^+}_{cov} \cup \{X^+\} \subseteq \mathcal{S}^{X^-}_{cov}$.

These properties are of interest for defining efficient algorithms and heuristics for searching relevant summaries, i.e. distinguishing *best* summaries w.r.t the defined objective function (Eq. 4). Due to space restriction, algorithmic optimization for searching $\mathcal{S}^X$ considering a given set of annotations $\hat{X}$ are not further discussed. Some details regarding the reduction of $\mathcal{S}^X$ that can be applied without introducing additional

[6] We denote $\preceq_Z$ the ordering relation over $Z \subseteq \mathcal{P}(C)$ according to the same rule.

technical notions are nevertheless introduced in the following subsection.

B. Thought on complexity reduction

Additional properties that can be used to reduce the complexity, e.g. by reducing the set $\mathcal{S}^X$, are proposed. Computational time reductions can first be obtained while constructing $\mathcal{S}^X$ by applying reduction on X. Two reductions of X are proposed. (1) *Remove redundant concepts from* X. It can easily be proved that, with $X' \preceq_{\mathcal{P}(C)} X$, $\mathcal{S}^X \subseteq \mathcal{S}^{X'}$. X can therefore be substituted by the smaller subset $X' \subseteq X$ such as $\mathcal{A}(X') = \mathcal{A}(X)$ and X' respects the *non-total redundancy property* – this is indeed the more specific covering summary. Thanks to this construction we ensure that $\mathcal{S}^X = \mathcal{S}^{X'}$, with the interesting property $|\mathcal{P}(X')| \leq |\mathcal{P}(X)|$ – with most often in practice $|\mathcal{P}(X')| << |\mathcal{P}(X)|$. This approach does not reduce $\mathcal{S}^X$. (2) *Abstract lower outliers*. Any concept $x \in X$ that has been observed a significantly lower amount of time (e.g. only once) can be substituted by the more specific abstraction of x (element of $\mathcal{A}(x) \setminus \{x\}$) which has the lower mass increasing the one of x. Even if this strategy may reduce $\mathcal{S}^X$ for any cut-off greater that one, considering a cut-off equals to 1 will distinguish all summaries corresponding the abstraction of at least two concepts – these summaries are the ones of interest in most practical applications. This idea can be extended to reduce the number of summaries to evaluate by considering the assumption that any interesting summary should factorize information in order to be meaningful. In accordance to this assumption we can exclude any summary $Y \in \mathcal{S}^X$ for which $\exists Y' \in \mathcal{S}^X$ with $Y' \preceq_{\mathcal{S}^X} Y$, $|Y'| = |Y|$.[7] This restriction can easily be computed by removing useless concept from G prior to applying Algorithm 1 – this is related to the hypernym closure. Other strategies could be to remove too abstract concepts and to consider specific concepts w.r.t the analysis of specific topological properties – e.g. concepts that are deep but have a large number of descendants are interesting candidate for summarization. Those optimizations techniques are context dependent and must be chosen in agreement with the defined objective function.

V. Evaluation & Discussion

Evaluating automatic summarization systems is a complex task as well as an open research topic – and no gold-standard dataset exists for evaluating the type of models we are studying. In this section we discuss preliminary results on the evaluation of the performance of the proposed model. This evaluation is based on human defined summaries of conceptual annotations provided by domain experts related to odor evaluation. Considering a domain ontology, several experts have, for several products, provided conceptual annotations in the form of sets of concepts. To a specific product, the sequence of annotations $\hat{X}$ has been analysed applying techniques experts use while performing sensorial analyses. Using this protocol we obtained radar charts corresponding to horizontal cuts of the poset, which forms a group of concepts that are homogeneous in term of concept specificity - this is the approach used by domain experts to analyse such data; Figure 5 provides an example. Domain experts finally build a summary by applying a subjective selection of concepts analysing such radar charts. For each product, 3 radar charts of different degrees of specificity have been provided - summaries are further generated by selecting a subset of the most frequently observed concepts. Note that the model studied in this paper can be used to generate more refined summaries composed of concepts having various degrees of specificity. The preliminary analyses we performed have shown that considering specific concepts of interest (those corresponding to the different degrees of specificity) the proposed model was able to find expected summaries, i.e. to rank relevant summaries according to the importance given to each concept in the radar view. Based on the analyses of the model that have been performed, without applying any constraints on the degree of specificity of the concepts to consider, we have also shown that such an approach could be used to automatize the summarizing process by, interestingly, generating more informative summaries.

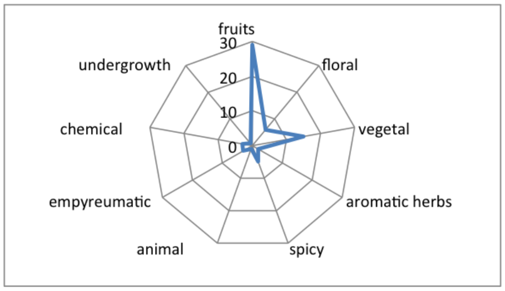

Fig. 5. Example of a radar chart used for sensorial analyses. The weight of each dimension (concept defined in the poset) corresponds to the number of (implicit) occurrences. Here, expected summary is Fruit, Vegetal and Floral.

VI. Related Work

Automatic summarization is a broad research topic related to several domains including Machine Learning, Data Mining, Natural Language Processing, and Information Retrieval. Two main types of fully automatic summarization approaches exist: extraction-based and abstraction-based. Extraction-based methods try to find the most informative elements (e.g. sentences) and to remove repetitive elements (e.g. scenes in videos). Abstraction-based methods first build a representation of the dataset to further analyze this representation to generate a summary - in this case the summary may information that are not explicitly in the original dataset.

Automatic summarization is also intricately linked to clustering and indexing tasks. Literature related to document representation (e.g. vector and probabilistic models), clustering and indexing is of interest [8]–[10]. As an example, LDA [11] clusters the documents and produces a set of topics into which the documents are clustered. Extensions taking account of topic hierarchies and correlation between topics

[7] which means that $\forall y \in Y$, $\exists y' \in Y'$ with $m(y') = m(y)$.

have also been proposed [12], [13]. Approaches for clustering and labeling a collection of resources indexed by concepts of a taxonomy have also been studied [14]. Nevertheless, they remain inspired from information retrieval approaches; the required objective function of the labeling optimization problem is reminiscent of the clustering and diversification processes in information retrieval: the similarity between the inner items of a cluster is maximized while the outer distance between items of two distinct clusters is maximized. The labeling process of clusters is then seen as a continuous optimization problem of distances whereas the feasible solutions are intrinsically discrete since they are related to a specificity level of the taxonomic hierarchy. HSLDA [15] also introduces a hierarchically supervised LDA model to infer hierarchical labels for a document. It assumes an existing label hierarchy in the form of a tree (e.g. multiple inheritance considered in our model is not allowed). The model infers one or more labels such that, if a label is inferred as relevant to a document, then all the labels from to the root of the tree are also inferred as relevant to the document. [16] highlights that applying the proposed inference rule, it is likely that many abstract labels will be classified as relevant without control on the specificity of the labels selected for summarization. To tackle this issue, they introduces a family of submodular functions to identify an appropriate set of topics from a DAG of topics for a group of documents. They characterize topic appropriateness through a set of desirable properties such as coverage, diversity, specificity, clarity, and relevance. Submodular functions are associated to these properties and mixed through a weighted average mean defining the objective function of the optimization problem the best summary results from. The coverage property is central in this approach. Indeed, in [16], unlike Human-like reasoning, no approximate reasoning is allowed over this property since it does not deal with frequency of occurrences, e.g. topics cannot be excluded from a summary even when they are poorly represented, as it is the case in our approach. Their axiomatic approach of the expected properties for summarization is close to our proposal but their related indicators and their management differ from ours: our indicators explicitly integrate masses or beliefs related to concepts; they also allow introducing control rules in the summarization process.

VII. Conclusion

Defining mathematical models enabling to automatically abstract and summarize bodies of information in a Human-like manner is a key challenge for Artificial Intelligence. We have proposed a general model to automatically summarize several conceptual annotations by considering knowledge representations providing *a priori* knowledge in the form of a *poset* formalizing the underlying structure of the concepts composing the annotations to analyze. A rigorous definition of the problem and a formal definition of a summary have been proposed; in addition, several interesting theoretical aspects highlighting the complexity of the challenge, as well as important properties of the search space have been discussed. Applications for data analysis and definition of intelligent agents are numerous considering the growing number of knowledge representations today available for a diversity of domains – e.g., gene analyses, information retrieval, and sensorial analyses. As an example, the evaluation of the proposed model performed in the domain of odor analysis, highlights the benefits of our proposal and shows how it could be used to automatize complex and time-consuming expert summarizing processes. Interestingly for the community, source code implementing the proposed approach as well as datasets are made available.

Additional large-scale experiments in several domains are currently performed to further criticize the model and discuss parameters tuning for specific use cases. Extended theoretical works are also performed to reduce the algorithmic complexity of finding the best summary w.r.t the proposed model. This aspect is of major concern to ensure method efficiency and practicality when applied to large knowledge representations. Interesting results based on the properties of the search space highlighted in this paper are currently studied.

References

[1] L. Saitta and J.-D. Zucker, *Abstraction in artificial intelligence and complex systems*. Springer, 2013, vol. 456.

[2] J.-D. Zucker, "A grounded theory of abstraction in artificial intelligence," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 358, no. 1435, pp. 1293–1309, 2003.

[3] S. Staab and R. Studer, *Handbook on ontologies*. Springer Science & Business Media, 2013.

[4] S. Harispe, S. Ranwez, S. Janaqi, and J. Montmain, "Semantic similarity from natural language and ontology analysis," *Synthesis Lectures on Human Language Technologies*, vol. 8, no. 1, pp. 1–254, 2015.

[5] N. Seco, T. Veale, and J. Hayes, "An Intrinsic Information Content Metric for Semantic Similarity in WordNet," in *16th European Conference on Artificial Intelligence*. IOS Press, 2004, pp. 1–5.

[6] S. Harispe, A. Imoussaten, F. Trousset, and J. Montmain, "On the consideration of a bring-to-mind model for computing the information content of concepts defined into ontologies," in *International Conference on Fuzzy Systems FUZIEEE*, 2015.

[7] V. Ranwez, S. Ranwez, and S. Janaqi, "Subontology extraction using hyponym and hypernym closure on is-a directed acyclic graphs," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 12, pp. 2288–2300, 2012.

[8] R. Baeza-Yates, B. Ribeiro-Neto *et al.*, *Modern information retrieval*. ACM press New York, 1999, vol. 463.

[9] F. Role and M. Nadif, "Beyond cluster labeling: Semantic interpretation of clusters contents using a graph representation," *Knowledge-Based Systems*, vol. 56, pp. 141–155, 2014.

[10] A. Bernardini, C. Carpineto, and M. D'Amico, "Full-subtopic retrieval with keyphrase-based search results clustering," in *Web Intelligence and Intelligent Agent Technologies, 2009. WI-IAT'09. IEEE/WIC/ACM International Joint Conferences on*, vol. 1. IET, 2009, pp. 206–213.

[11] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[12] D. Blei and J. Lafferty, "Correlated topic models," *Advances in neural information processing systems*, vol. 18, p. 147, 2006.

[13] D. Griffiths and M. Tenenbaum, "Hierarchical topic models and the nested chinese restaurant process," *Advances in neural information processing systems*, vol. 16, p. 17, 2004.

[14] N. Fiorini, "Semantic similarities at the core of generic indexing and clustering approaches," Ph.D. dissertation, University of Montpellier, 2015.

[15] A. J. Perotte, F. Wood, N. Elhadad, and N. Bartlett, "Hierarchically supervised latent dirichlet allocation," in *Advances in Neural Information Processing Systems*, 2011, pp. 2609–2617.

[16] R. B. Bairi, R. Iyer, G. Ramakrishnan, and J. Bilmes, "Summarization of multi-document topic hierarchies using submodular mixtures," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, 2015, pp. 553–563.