MATHÉMATIQUES ET SCIENCES HUMAINES

RÉGIS GIRARD
HENRI RALAMBONDRAINY

Recherche de concepts à partir de données arborescentes et imprécises

Mathématiques et sciences humaines, tome 147 (1999), p. 87-111
<http://www.numdam.org/item?id=MSH_1999__147__87_0>

Math. Inf. Sci. hum., (37e année, n° 147, *1999*, pp. 87-111)

RECHERCHE DE CONCEPTS À PARTIR DE DONNÉES ARBORESCENTES ET IMPRÉCISES

Régis GIRARD[1], Henri RALAMBONDRAINY[1]

RÉSUMÉ – *Dans cet article, nous proposons un formalisme de représentation de données structurées et imprécises, les Arborescences Symboliques Nuancées (ASN), qui est fondé sur la notion d'attribut-valeur. Les ASN nous permettent de représenter des entités composées de parties et sous-parties dont les caractéristiques peuvent être imprécises, inconnues ou bien inapplicables et prenant en compte les liens pouvant exister entre les valeurs des différentes caractéristiques.*

Nous nous intéressons à la recherche de concepts à partir d'un ensemble d'entités décrites par les ASN. La définition des concepts repose sur une extension des treillis de Galois au cas de données arborescentes et nuancées. Pour rechercher les concepts, nous présentons un algorithme incrémental permettant de calculer un treillis extrait du treillis de Galois en élagant les concepts trop généraux.

MOTS-CLÉS – Données arborescentes, nuances, concepts, treillis de Galois.

SUMMARY – Finding concepts from fuzzy symbolic rooted tree data

In this article, we propose a formalism (ASN) to deal with imprecise and structured data described with attributes and imprecise values. The ASN allow us to represente entities that are composed with parts and sub-parts ; values may be imprecise, unknown and the attributes may be not applicable. We can also take into account constraints that exist between the values of the attributes.

We aim to find concepts from a set of entities described with ASN. Concepts are defined from an extension of the Galois lattice theory to deal with imprecise and structured data. To find concepts, we propose an incremental algorithm that compute a lattice concepts extracted from the Galois lattice where the too general concepts – in regard to a given criteria – are not computed.

KEYWORDS – Structured data, imprecision, concepts, Galois Lattice.

1 INTRODUCTION

Classer des objets ayant des propriétés communes est une nécessité dans bien des disciplines scientifiques. Avant de pouvoir automatiser un processus de classification, une étape incontournable est celle du transfert de la connaissance par un expert du domaine. Or, dans beaucoup de disciplines, et en particulier dans les domaines des Sciences de la Nature, lorsqu'on demande à un expert de décrire un objet ou la connaissance qu'il a de son domaine, on constate que l'information qu'il transmet est d'une part, structurée et d'autre part, entachée d'imprécision et de nuances qu'il exprime par des termes linguistiques.

La modélisation de donnés et concepts structuraux en Apprentissage a été abordée sous différents angles. On peut citer à titre d'exemple les représentations qui utilisent

[1] IREMIA, Université de La Réunion, 15 avenue Renée Cassin, BP 7151, 97715 Saint-Denis Messag. Cedex 9, e-mail : rgirard@univ-reunion.fr, ralambon@univ-reunion.fr

des réseaux sémantiques [24], le formalisme des graphes [5] et les ensembles des termes [8, 9, 17]. Ces systèmes de représentation sont plus puissants que la représentation par attribut-valeur mais supposent toujours qu'une propriété est soit vraie soit fausse. Cela pose un problème pour le traitement de données issues par exemple des Sciences de la Nature. En effet, les experts utilisent très souvent des termes imprécis exprimant des fréquences, des intensités pour qualifier les valeurs qu'ils accordent aux propriétés leur permettant de décrire des espèces ou des genres ; leurs descriptions sont ainsi nuancées. Il est aussi courant que des propriétés soient applicables dans certains cas et inapplicables dans d'autres.

Mais comment manipuler des données structurées et imprécises?

D'un côté, les méthodes usuelles de l'analyse de données et de la classification automatique supposent que les observations sont décrites de manière précise et ne manipulent pas de données structurées. Certains algorithmes d'apprentissage [18, 20] prennent en compte les valeurs inconnues ou impossibles, mais de manière ad hoc, l'intégration de ces valeurs dans le langage de représentation n'est pas toujours claire.

D'un autre côté, la plupart des travaux en classification floue concernent l'adaptation d'algorithmes de classification automatique manipulant des données décrites par des vecteurs de $\mathbb{R}^p$. En particulier, beaucoup d'auteurs ont travaillé sur la généralisation des algorithmes de type « c-means » en introduisant le concept de partition floue [3, 4]. À partir de l'ensemble des observations, on calcule des partitions floues. Les classes ainsi trouvées sont décrites de manière extensive : à chaque classe est associé un ensemble d'objets lui appartenant avec un degré plus ou moins élevé. Il s'ensuit qu'il n'est pas aisé d'interpréter ces classes et d'en extraire des concepts.

2 REPRÉSENTATION DES DONNÉES ET DES PROPRIÉTÉS

Il existe différentes manières de représenter des données sous forme d'arborescences. Nous décrivons le modèle des termes avant d'aborder notre formalisme.

2.1 Le modèle des termes

Le langage des termes est trés utilisé dans de nombreux domaines de l'informatique (calcul formel, logique, théorie des langages,...). Il sert, en particulier, pour la spécification de programmes. Daniel-Vatonne [8, 9] propose de représenter à l'aide d'un tel modèle un ensemble d'objets pour l'apprentissage de connaissances. Les termes sont définis à partir d'une signature, qui permet leur construction :

DÉFINITION 2.1 *Une signature est un quintuplet* $(S, F, \sigma, \alpha, base)$ *où :*

- S *est un ensemble fini de* types
- F *est un ensemble fini de* symboles
- σ *est une application surjective de* F *dans* S. *Pour* $f \in F$, $\sigma(f)$ *désigne le type de* f
- α *est une application de* F *dans* S^* *(l'ensemble des mots formés à partir des éléments de* S*). Pour* $f \in F$, $\alpha(f)$ *fixe l'ordre et le type des arguments de* f. *Si* f *n'a pas d'argument,* $\alpha(f)$ *est le mot vide noté* ε.
- *base est une élément de* S *appelé* type de base

Cette signature permet de générer pour chaque type un ensemble de termes particulier. Afin de pouvoir manipuler des descriptions partielles, pour chaque type s on ajoute à la signature un symbole spécial Ω_s interprétable par « il existe une information de type s mais on ne connaît pas ou ne s'intéresse pas à sa valeur ». Les termes contenant au moins un symbole Ω_s sont des *descriptions partielles* relativement aux termes entièrement

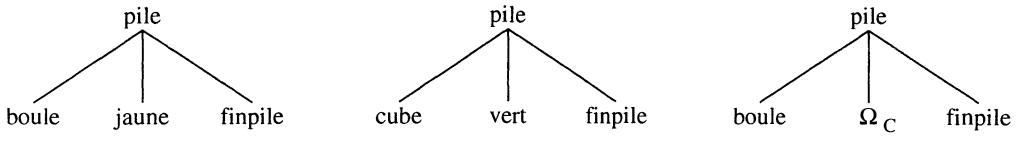


FIG. 1 – *Représentations arborescences de termes*

spécifiés. Ils permettent de représenter des classes et des données sont représentées dans un même formalisme.

Exemple 2.1 Soient $S = \{p, f, c\}$ *et* $F = \{pile, finpile, boule, cube, jaune, vert\} \cup \{\Omega_p, \Omega_f, \Omega_c\}$ *et l'application* σ *définie comme suit: pile* : $fcp \rightarrow p$; *finpile* : $p \rightarrow \epsilon$; *boule* :$\rightarrow$ ϵ; *cube* : $f \rightarrow \epsilon$; *jaune* : $c \rightarrow \epsilon$; *vert* :$\rightarrow$ ϵ;. *Une boule jaune: pile(boule, jaune, finpile), un cube vert: pile(cube, vert, finpile) et une boule dont on ne s'intéresse pas à la couleur: pile(boule,*Ω_c*,finpile) sont représentés dans la figure 1.*

L'ensemble des termes est muni d'une structure de treillis et une correspondance de Galois est définie entre un ensemble d'objets et le treillis des termes. Par rapport aux méthodes classiques de classification conceptuelle [13, 23], le formalisme des termes présente l'avantage de pouvoir traiter des données plus riches car arborescentes. Cependant la sémantique de la description partielle ne prend pas en compte de manière satisfaisante l'information pertinente qu'est l'absence d'un composant (branche, sous-arbre) pour des données structurées. D'autre part ce modèle n'intègre pas la notion de domaine (ensemble de valeurs possibles) d'un attribut. Ceci ne permet pas l'exploitation des propriétés (structure hiérarchique) ou contraintes pouvant exister sur les valeurs des attributs pour la création de concepts. Enfin, le modèle suppose les données parfaitement définies.

2.2 Les Arborescences Symboliques Nuancées

En analyse conceptuelle, l'extraction de concepts à partir de données tabulaires, à valeurs binaires ou ordinales en utilisant les treillis de Galois a été largement étudiée [12, 13, 23]. Nous proposons dans cet article de traiter des données non plus tabulaires, mais arborescentes et dont les valeurs sont structurées sous forme de treillis.

Le formalisme des Arborescences Symboliques Nuancées est fondé sur les travaux de Ralambondrainy [21]. Celui-ci repose sur une description à partir d'attributs et de valeurs auxquelles on associe des ensembles de nuances structurés sous forme de treillis. Ces treillis de nuances permettent de modéliser l'imprécision relative aux valeurs des caractéristiques d'une observation donnée et offre la possibilités d'avoir des valeurs non comparables. Remarquons que l'utilisation de treillis est prévue dans les théories de la logique et des ensembles flous pour évaluer le degré de véracité d'un énoncé mais peu utilisée dans la pratique. Citons les travaux de Ginsberg qui s'appuie sur un bitreillis de « croyances » [14]. pour gérer la mise-à-jour d'un énoncé dans les bases de connaissance, les travaux de Sallantin [22] en apprentissage symbolique. Les attributs dans notre modèle peuvent être de type simple ou structuré offrant ainsi la possibilité de manipuler des données de nature arborescente. Enfin la définition de contraintes entre les valeurs des attributs permet de représenter des connaissances sur le domaine d'application.

2.2.1 Représentation de l'imprécision

Pour prendre en compte l'imprécision, nous qualifions les valeurs d'un attribut par des nuances. Le terme de nuance que nous employons est un terme générique regroupant les différentes manières de qualifier la valeur d'un attribut. Les nuances nous permettent de

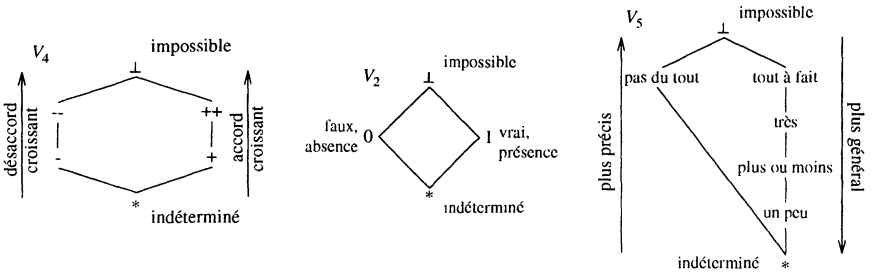


FIG. 2 – *Treillis de nuances d'accord et d'intensité*

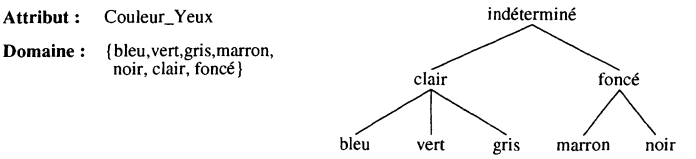


FIG. 3 – *Exemple de taxinomie pour la couleur des yeux*

prendre en compte l'imprécision, l'intensité, la croyance ou la subjectivité relative à la valeur observée d'un attribut par rapport à sa valeur réelle. D'une manière générale, on représentera une nuance par un ensemble de symboles muni d'une structure de treillis dont la relation d'ordre s'interprète en terme de précision. Par convention les extremum d'un treillis de nuances sont toujours désignés par les mêmes symboles et représenteront toujours la même information. L'élément maximal est noté $\perp$ et représente l'impossibilité ou l'absurdité de donner une valeur particulière de son domaine à un attribut. L'élément minimal est noté $*$ et représente l'indétermination totale concernant la nuance que l'on associe à une valeur particulière d'un attribut.

La figure 2 montre des treillis représentant différentes nuances que l'on utilisera dans les exemples qui suivront. Pour représenter les données précises usuelles, on utilise la chaîne de nuances $V_0 = \{vrai - faux\}$. Afin de facilement pouvoir interpréter sémantiquement les treillis de nuances, nous limitons ces derniers à des treillis unions disjointes de chaînes.

DÉFINITION 2.2 *Un treillis de nuances est un ensemble V de symboles ordonné ayant un plus grand élément $\perp$, un plus petit élément $*$, tel que $V \setminus \{*, \perp\}$ est union disjointe de une ou plusieurs chaînes C_i :*

$$V = * \oplus (C_1 \biguplus \ldots \biguplus C_n) \oplus \perp$$

où $\oplus$ désigne la somme linéaire et $\biguplus$ l'union disjointe.

2.2.2 Domaine des attributs

Nous ne considérons que des attributs simples dont l'ensemble des valeurs possibles D, ou domaine, est un ensemble discret, éventuellement muni d'une structure d'arbre.

Par exemple, pour représenter la couleur des yeux d'une personne, on pourra utiliser un attribut symbolique *Couleur_Yeux* de domaine $\{bleu, vert, gris, marron, noir\}$. Mais si l'on veut pouvoir décrire une personne ou un type de personne qui a les yeux clairs ou bien dont la couleur des yeux est indéterminée, on préférera utiliser une taxinomie telle celle de la figure 3.

Il est toujours possible de munir un domaine d'une structure de sup-demi treillis en prenant comme relation d'ordre l'inclusion et comme opérateur sup l'union. Pour représenter dans un même cadre des domaines structuré ou non, on définit sur chaque domaine,

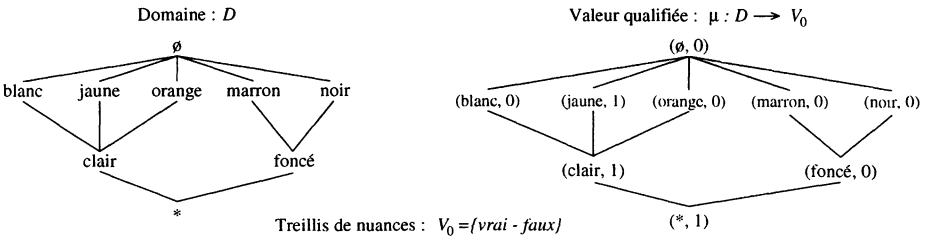


FIG. 4 – *Exemple de valeur qualifiée*

une structure de treillis où la relation d'ordre est interprétée comme une relation de généralisation/spécialisation.

DÉFINITION 2.3 *On munit chaque domaine discret D, d'une relation d'ordre et d'une structure de treillis $(D, \leqslant, \vee, \wedge, *, \emptyset)$ en lui adjoignant un maximum et minimum. La relation d'ordre, notée $\leqslant$, est obtenue en inversant l'ordre induit par l'opérateur d'inclusion :*

$$\forall x, y \in D : x \leqslant y \Longleftrightarrow y \subset x$$

la valeur x est dite plus générale que y.

Notons que le choix de choisir comme relation d'ordre la relation inverse de d'inclusion est motivé par le fait que plus une valeur est générale, moins on a d'information sur cette valeur.

2.2.3 Attributs simples

DÉFINITION 2.4 *Un attribut A, dit simple, est un triplet $(D, V, [\![A]\!])$ où :*

$D \in \mathcal{D}$ est son domaine, c'est à dire un ensemble de valeurs élémentaires ayant une structure de treillis

$V \in \mathcal{V}$ est un treillis de nuances

$[\![A]\!] \subseteq V^D$ est son treillis de valeurs, valeurs qui sont dites qualifiées

DÉFINITION 2.5 *Une valeur qualifiée pour un attribut A est une application μ de son domaine D dans le treillis de nuances V qui lui est associé.*

La valeur qualifiée μ peut être vue comme une fonction d'appartenance de la valeur de l'attribut A à l'ensemble D qui prend ses valeurs dans le treillis V, c'est donc un sous-ensemble flou de D, au sens large, que l'on appellera ensemble T-flou (le T indiquant la structure de treillis de l'ensemble de nuances).

La représentation classique du degré d'appartenance dans la théorie des sous-ensembles flous est une valeur réelle dans l'intervalle $[0, 1]$. La plupart du temps les descriptions d'observations du monde réel sont faites de manière discrète et les termes linguistiques que l'on utilise pour exprimer diverses nuances sur ces descriptions sont par essence discrets, c'est pourquoi dans notre cadre les ensembles T-flous sont plus appropriés que les ensembles flous classiques.

Exemple 2.2 Considérons un attribut Couleur *dont le domaine D est le treillis de gauche de la figure 4, construit à partir d'une taxinomie et auquel on associe le treillis de nuances V_0. La valeur précise* jaune *pour cet attribut est représentée par le morphisme μ_{jaune} tel que : $\mu_{jaune}(*) = \mu_{jaune}(jaune) = \mu_{jaune}(clair) = 1$ et $\mu_{jaune}(blanc) = \mu_{jaune}(orange) = \mu_{jaune}(marron) = \mu_{jaune}(noir) = \mu_{jaune}(fonce) = \mu_{jaune}(\emptyset) = 0$, c'est l'ensemble T-flou représenté par le treillis de droite de la figure 4.*

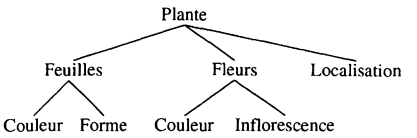


FIG. 5 – *Représentation graphique d'un attribut structuré*

L'ensemble des valeurs qualifiées d'un attribut $(D, V, [\![A]\!])$ est muni d'une structure de treillis induite par celle existant sur V telle que :

$$\forall \mu, \nu \in V^D : \mu \leqslant \nu \Longleftrightarrow \forall d \in D, \mu(d) \leqslant \nu(d)$$

Le plus petit élément du treillis V^D est la valeur qualifiée $* : D \longrightarrow V$ telle que $\forall d \in D, *(d) = *$ c'est à dire la fonction qui, à chaque élément du domaine, associe la nuance indéterminé. Le plus grand élément est la valeur qualifiée $\perp : D \longrightarrow V$ telle que $\forall d \in D, \perp(d) = \perp$ c'est à dire la fonction qui, à chaque élément du domaine, associe la nuance impossible ou inapplicable. La relation d'ordre $\leqslant$ est une relation de généralisation ; on dira que la valeur qualifiée μ est plus générale que ν, et on appelle $\mu \vee \nu$ le généralisé de μ et ν, c'est à dire la plus spécifiques des valeurs plus générales que μ et ν.

L'ensemble des valeurs $[\![A]\!]$ d'un attribut simple A n'est pas forcément égal à l'ensemble de ses valeurs qualifiées, en revanche c'est toujours un sous-treillis de V^D dont le maximum est $\perp$, le minimum est $*$ et qui est ordonné par la relation de généralisation induite de celle existant sur V^D. $[\![A]\!]$ est l'ensemble des valeurs qualifiées permises pour l'attribut A.

2.2.4 Attributs structurés

Pour représenter des composants d'une entité structurée, on considère la notion suivante :

DÉFINITION 2.6 *Un attribut A est dit de type structuré s'il regroupe un ensemble d'attributs A_i, où les A_i pour $i = 1 \ldots p$ sont des attributs distincts de type simple* ou *structuré, on note : $A = \langle A_1, \ldots, A_p \rangle$.*

Exemple 2.3 On pourrait, par exemple, décrire une plante en utilisant les attributs simples suivants :

$Couleur = (D_C, V_4, T_C)$, où D_C est un ensemble de symboles désignant des couleurs,

$Forme = (D_F, V_4, T_F)$, où $D_F = \{ovale, pointue, large\}$ désigne les formes possible des feuilles,

$Inflorescence = (D_I, V_2, T_I)$, où $D_I = \{$en épi, en grappe, isolée$\}$ désigne le type d'inflorescence,

$Localisation = (D_L, V_0, T_L)$, où D_L est un ensemble de symboles désignant les localisations possibles de la plante,

et à partir de ces attributs, on définit les attributs structurés $Feuilles = \langle Forme, Couleur \rangle$, $Fleurs = \langle Couleur, Inflorescence \rangle$ et finalement $Plante = \langle Feuilles, Fleurs, Localisation \rangle$. La figure 5 montre la représentation graphique de l'attribut structuré Plante.

PROPOSITION 2.1 *L'ensemble $[\![A]\!]$ des valeurs de A : $[\![A]\!] = \{(\mu_1, \ldots, \mu_p) | \forall i \in \{1..p\}, \mu_i \in [\![A_i]\!]\}$ a une structure de treillis.*

L'ensemble des valeurs $[\![A]\!]$ est défini récursivement, si A_j est un attribut simple $[\![A_j]\!]$ est le treillis de ses valeurs qualifiées permises, si $A_j = \langle A_{j1}, \ldots, A_{jk} \rangle$ alors $[\![A_j]\!] = [\![A_{j1}]\!] \times \ldots \times [\![A_{jk}]\!]$, c'est un treillis comme produit de treillis. Pour ces même raisons, l'ensemble

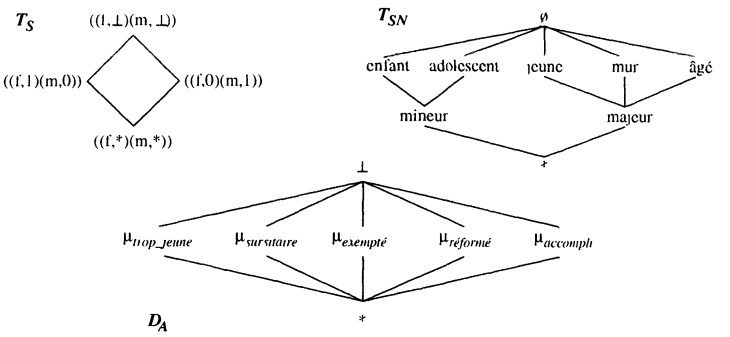


FIG. 6 –

$[\![A]\!]$ est un treillis. La relation d'ordre sur $[\![A]\!]$ est induite de celles existant sur les $[\![A_j]\!]$, l'on a :

$$\forall \mu = \langle \mu_1, \ldots, \mu_p \rangle \in [\![A]\!], \forall \mu' = \langle \mu'_1, \ldots, \mu'_p \rangle \in [\![A]\!], \mu \leqslant \mu' \Longleftrightarrow \forall j \in \{1 \ldots p\}, \mu_j \leqslant \mu'_j$$

Cet ordre est une relation de généralisation et on dira que la valeur μ est plus générale que la valeur μ'.

2.2.5 Contraintes entre les attributs

Les valeurs des attributs peuvent être liées entre elles par des contraintes, comme le montre l'exemple suivant. Ces contraintes peuvent, a priori, aussi bien porter sur des valeurs d'attributs simples que structurés.

On note $\sqcap_{A_j}(\mu)$ la projection sur l'attribut A_j telle que pour l'attribut structuré $A = \langle A_1, \ldots, A_p \rangle$, l'on ait : $\forall \mu = (\mu_1, \ldots, \mu_p) \in [\![A]\!], \quad \sqcap_{A_j}(\mu) = \mu_j$

Exemple 2.4 On veut décrire des personnes à partir d'un attribut structuré Personne = ⟨Sexe, Age, ServiceNational⟩, *défini à partir des attributs simples* $Sexe = (D_S, V_S, T_S)$, $Age = (D_A, V_A, T_A)$, *et* $ServiceNational = (D_{SN}, V_{SN}, T_{SN})$ *tels que :*

$D_S = \{m, f\}$, $V_S = V_2$ *et l'ensemble des valeurs possibles de l'attribut* Sexe *est réduit au treillis* T_S *de la figure 6.*

$V_A = V_0$ *et* D_A *est le domaine hiérarchique de la figure 6.*

D_{SN} =⟨trop jeune, sursitaire, exempté, réformé, accompli⟩, $V_{SN} = V_2$ *et l'ensemble des valeurs* T_{SN} *est réduit au sous-treillis de* $V_2{}^{D_{SN}}$ *que montre la figure 6 où on a par exemple,* μ_{trop_jeune}=((trop jeune, 1)(sursitaire, 0)(exempté, 0) (réformé, 0)(accompli, 0)).

La description d'une personne est représentée par une valeur de $[\![Personne]\!]$. *A priori, pour une description, les attributs* Sexe *et* ServiceNational *peuvent prendre n'importe quelle valeur, mais puisqu'une personne de sexe féminin ne fait pas de service national l'attribut* ServiceNational *devient non pertinent dans ce cas. Ce qu'on exprime par la contrainte :* « *Si dans une description l'attribut* Sexe *a la valeur* féminin, *alors l'attribut* ServiceNational *ne peut avoir que la valeur* inapplicable », *que l'on peut écrire :*

$$\sqcap_{Sexe}(\mu) = \mu_{feminin} \Longrightarrow \sqcap_{ServiceNational}(\mu) = \bot$$

où $\sqcap$ *est l'opérateur de projection.*

La contrainte suivante, quant à elle, exprime le fait que, pour toute description d'une personne mineure par une valeur $\mu \in [\![Personne]\!]$, *l'attribut* ServiceNational *est applicable, mais sa valeur est forcément* trop jeune *:*

$$\sqcap_{Age}(\mu) \geqslant \mu_{mineur} \Longrightarrow \sqcap_{ServiceNational}(\mu) = \mu_{trop\_jeune}$$

où μ_{mineur}=((enfant, 1) (adolescent, 1) (mineur, 1) (jeune, 0) (mur, 0) (âgé, 0) (majeur,0) ($*$, 1) ($\emptyset$, 0)) *représente l'âge d'une personne mineure, qui peut être un adolescent ou un enfant.*

Étant donné un ensemble de contraintes Γ portant sur les attributs constituant un attribut structuré A, tous les éléments de $[\![A]\!]$ ne vérifient pas forcément ces contraintes. On définit la validité d'une valeur de A en regard des contraintes portant sur les attributs $Att(A)$ de la manière suivante.

DÉFINITION 2.7 *Validité de la valeur d'un attribut.*
Soient $A = \langle A_1, \ldots, A_p \rangle$ *un attribut structuré et* $\Gamma(A) = \{C_1, \ldots C_k\}$ *un ensemble de contraintes, on dit qu'une description* $\mu \in [\![A]\!]$ *est valide si et seulement si :*

soit $\mu = *$ *ou* $\mu = \perp$

soit $\mu = (\mu_1, \ldots, \mu_p)$ *et l'ensemble des valeurs* $\{\mu_1, \ldots, \mu_p\}$ *satisfait l'ensemble des contraintes* Γ.

On trouvera dans [15] une étude détaillée de la structure des ensembles de valeurs valides en fonction des différentes formes de contraintes. On y montre que les seules contraintes pour lesquelles l'ensemble des valeurs valides d'un attribut est stable pour l'opérateur de généralisation $\wedge$ utilisé pour la classification sont les suivantes :

$$\begin{array}{ll} \sqcap_{A_j}(\mu) > a \Rightarrow \sqcap_{A_k}(\mu) = b & \sqcap_{A_j}(\mu) \geqslant a \Rightarrow \sqcap_{A_k}(\mu) = b \\ \sqcap_{A_j}(\mu) > a \Rightarrow \sqcap_{A_k}(\mu) \geqslant b & \sqcap_{A_j}(\mu) \geqslant a \Rightarrow \sqcap_{A_k}(\mu) \geqslant b \\ \sqcap_{A_j}(\mu) > a \Rightarrow \sqcap_{A_k}(\mu) < b & \sqcap_{A_j}(\mu) \geqslant a \Rightarrow \sqcap_{A_k}(\mu) < b \\ \sqcap_{A_j}(\mu) > a \Rightarrow \sqcap_{A_k}(\mu) \leqslant b & \sqcap_{A_j}(\mu) \geqslant a \Rightarrow \sqcap_{A_k}(\mu) \leqslant b \end{array}$$

La forme de ces contraintes traduit le fait que la valeur d'un attribut A_j ne peut avoir d'effet sur la valeur d'un autre attribut A_k que si elle est suffisament précise ou connue.

2.2.6 Le treillis des Arborescences Symboliques Nuancées

Soient un ensemble de p attributs $A_j = (D_j, V_j, [\![A_j]\!])$ et un ensemble de contraintes $\Gamma(A)$.

DÉFINITION 2.8 *On appelle Arborescence Symbolique Nuancée un élément valide du treillis* $[\![A]\!]$ *défini à partir de l'attribut structuré* $A = \langle A_1, \ldots, A_p \rangle$.

2.3 La représentation des observations

On suppose donnés :

- un ensemble de n observations $\mathcal{E}$
- un ensemble de p attributs $A_j = (D_j, V_j, [\![A_j]\!])$
- un ensemble de contraintes $\Gamma(A)$

Chaque observation e est décrite à l'aide des attributs A_j. Un attribut A_j peut-être vu comme une fonction attribuant à une observation e une valeur qualifiée : $A_j(e) = \mu_j$ avec $\mu_j \in [\![A_j]\!]$. En considérant l'attribut structuré $A = \langle A_1, \ldots, A_p \rangle$. La description complète de l'observation ϵ est alors une ASN valide $\mu = (\mu_1, \ldots, \mu_n)$ du treillis $[\![A]\!]$.

3 CLASSIFICATION CONCEPTUELLE

Nous définissons les concepts à partir d'une correspondance de Galois entre les observations et l'ensemble de leurs descriptions qui sont des arborescences symboliques nuancées du treillis $[\![A]\!]$.

Lorsque le treillis $[\![A]\!]$ est très grand ou lorsque $\mathcal{E}$ contient beaucoup d'entités, le treillis de Galois engendré est constitué d'un nombre considérable de concepts, le maximum étant $\min(2^{\mathcal{E}}, card([\![A]\!]))$. De plus, beaucoup de ces concepts ont des descriptions contenant beaucoup de valeurs inconnues, c'est pourquoi nous proposons une méthode permettant de construire incrémentalement un treillis de concepts qui est une partie finissante du treillis de Galois. Ce treillis est obtenu en élaguant les concepts dont le niveau de généralité est supérieur à un seuil fixé (si le seuil de généralité est fixé à 1, on génére le treillis de Galois en entier). Nous calculons le niveau de généralité d'un concept à l'aide d'un indice de distance défini sur les ASN.

3.1 Définition des concepts

Les treillis de Galois sont utilisés en analyse conceptuelle [2, 12, 23] pour générer tous les concepts d'un ensemble d'objets décrits par des vecteurs d'attributs dont les valeurs sont précises. Nous étendons cette méthode au cas d'entités structurées dont les caractéristiques sont imprécises. L'ensemble $[\![A]\!]$ ayant une structure de treillis, nous pouvons mettre en correspondance le treillis des parties $\mathcal{E}$ et $[\![A]\!]$.

Soit l'application $\rho : \mathcal{E} \rightarrow [\![A]\!]$ qui associe à chaque entité $e \in \mathcal{E}$ sa description par une ASN de A. Le triplet $(\mathcal{E}, [\![A]\!], \rho)$ forme alors un contexte à partir duquel on définit la correspondance de Galois constituée des deux applications :

$$\begin{array}{rccl} int: & \mathcal{P}(\mathcal{E}) & \rightarrow & [\![A]\!] \\ & E & \mapsto & int(E) = \bigwedge\limits_{e \in E} \rho(e) \end{array} \qquad \begin{array}{rccl} ext: & [\![A]\!] & \rightarrow & \mathcal{P}(\mathcal{E}) \\ & \mu & \mapsto & ext(\mu) = \{e \in \mathcal{E} \mid \mu \leqslant \rho(e)\} \end{array}$$

où le symbole $\leqslant$ désigne la relation d'ordre existant sur l'ensemble des valeurs structurées de l'attribut A, et où $\wedge$ désigne l'opération de généralisation entre valeurs structurées.

Le couple d'applications (int, ext) forme une correspondance de Galois entre les treillis $\mathcal{P}(\mathcal{E})$ et $[\![A]\!]$.

Un concept est alors défini comme dans l'Analyse Conceptuelle [13] par son intension I et son extension E, où $I \in [\![A]\!]$ et $E \in \mathcal{P}(\mathcal{E})$ sont tels que $I = int(E)$ et $E = ext(I)$.

Rappelons que l'opération de généralisation est stable pour les contraintes que l'on autorise sur l'attribut A , cela nous assure que si les entités sont décrites par des arborescences symboliques nuancées, c'est à dire des éléments de $[\![A]\!]$ qui satisfont l'ensemble des contraintes $\Gamma(A)$, alors les intensions des concepts calculés satisfont aussi ces contraintes.

3.2 Construction d'un treillis de concepts

3.2.1 Indice de distance

Nous construisons un indice de distance sur les ASN qui permet d'évaluer le degré de généralité d'une ASN en comparant celle-ci avec l'ASN la plus générale, c'est à dire celle pour laquelle chaque attribut à la valeur *.

Le but est d'utiliser ce degré de généralité lors de la construction du treillis afin de ne pas générer les concepts dont l'intension est trop générale, i.e. dont trop d'attributs ont une valeur indéterminée. Si l'on veut que le sous-ensemble de concepts ainsi calculé soit une partie finissante du treillis de Galois, l'indice de distance doit vérifier la propriété suivante :

PROPRIÉTÉ 3.1 *Soit $\mathcal{G}$ le treillis de Galois entre un ensemble d'entités $\mathcal{E}$ et l'ensemble des ASN d'un attribut structuré A. L'indice de distance $\mathcal{D}$ définit sur $[\![A]\!]$ doit vérifier :*

$$\forall\ C = (I, E),\ C' = (I', E') \in \mathcal{G}\ :\ I < I' \Longrightarrow \mathcal{D}(I, *) < \mathcal{D}(I', *)$$

Cette propriété nous assure que si un concept n'est pas généré alors aucun autre concept plus général ne l'est.

3.2.2 Indice de distance sur les treillis de nuances

L'indice de distance suivant sur les treillis de nuances tient compte de la forme particulière de ces treillis et de leur sémantique. En particulier si deux nuances sont incomparables, la valeur de l'indice est maximum et si deux nuances sont comparables, la valeur de l'indice est en rapport avec la longueur de la chaîne reliant ces deux éléments.

PROPOSITION 3.1 *Soit V un treillis de nuances, la fonction δ définie par :*

$$\forall x, y \in V : \delta(x, y) = \frac{l(x \vee y, x \wedge y)}{l(\bot, x \vee y) + l(x \vee y, x \wedge y) + l(x \wedge y, *)}$$

où $l(x, y)$ est la longueur de la plus courte chaîne reliant x à y, est un indice de distance.

Preuve :
La fonction δ vérifie les propriétés d'un indice de distance :
$\forall x, y \in V : \delta(x, y) \geqslant 0$ car la longueur d'une chaîne est positive ou nulle
δ est symétrique car $\forall x, y \in T : x \vee y = y \vee x$ et $x \wedge y = y \wedge x$
$\forall x \in T : l(x \vee x, x \wedge x) = l(x, x) = 0$
$\forall x, y \in T : \delta(x, y) = 0 \Rightarrow l(x \vee y, x \wedge y) = 0 \Rightarrow x \vee y = x \wedge y \Rightarrow x = y$
□

PROPOSITION 3.2 *Soit V un treillis de nuances, alors l'indice de distance δ est tel que :*

$$\forall x, y \in V : x < y \Longrightarrow \delta(x, *) < \delta(y, *)$$

Preuve :
Étant donné la définition des treillis de nuances, si $x < y$ alors :
soit $x = *$ et $y = \bot$, et on a bien $\delta(*, *) = 0 < \delta(\bot, *) = 1$,
soit x et y appartiennent à la même chaîne, et dans ce cas $l(x, *) < l(y, *)$ donc $\delta(x, *) < \delta(y, *)$
□

3.2.3 Indice de distance sur les ASN

La structure arborescente d'une ASN nous permet de représenter une entité structurée en tenant compte des dépendances structurelles existant entre les parties et les sous-parties de cette entité, ainsi que des liens existant entre les valeurs des attributs. Dans une telle description, les valeurs décrivant les propriétés élémentaires d'une entité sont uniquement représentées par les valeurs des attributs simples composant A, c'est à dire les feuilles de l'ASN.

Intuitivement, une ASN donnée est d'autant plus générale qu'elle est constituée d'attributs simples dont les valeurs sont proches de la valeur inconnue. L'indice de distance que

nous définissions sur les ASN découle directement de cette vision intuitive, il est uniquement fonction des valeurs des attributs simples que nous considérons tous d'importance égale.

DÉFINITION 3.1 *Étant donné un ensemble d'attributs $\mathcal{A}$, on définit une fonction $p : \mathcal{A} \to I\!N$ qui affecte à chaque attribut $A \in \mathcal{A}$ un poids égal au nombre de feuilles de l'arborescence par laquelle on peut représenter graphiquement A.*

- *si A est un attribut simple alors $p(A) = 1$*
- *si $A = \langle A_1, \ldots A_k \rangle$ est un attribut structuré alors $p(A) = \sum_{i=1}^{k} p(A_i)$*

PROPOSITION 3.3 *Soit A un attribut, la fonction $\mathcal{D} : [\![A]\!] \times [\![A]\!] \to [0,1]$ définie par : $\forall \mu_A, \mu'_A \in [\![A]\!]$,*

1) si $A = (D, V, T)$ est un attribut simple, on a $\mu_A = ((d_1, v_1) \ldots (d_q, v_q))$, $\mu'_A = ((d_1, v'_1) \ldots (d_q, v'_q))$ et

$$\mathcal{D}(\mu_A, \mu'_A) = \frac{\sum_{i=1}^{q} \delta(v_i, v'_i)}{q}$$

2) si $A = \langle A_1, \ldots A_k \rangle$ est un attribut structuré, on a $\mu_A = \langle \mu_{A_1}, \ldots, \mu_{A_k} \rangle$, $\mu'_A = \langle \mu'_{A_1}, \ldots, \mu'_{A_k} \rangle$ et

$$\mathcal{D}(\mu_A, \mu'_A) = \frac{\sum_{i=1}^{k} p(A_i).\mathcal{D}(\mu_{A_i}, \mu'_{A_i})}{p(A)}$$

est un indice de distance sur les valeurs de A.

L'indice de distance $\mathcal{D}$ ainsi défini vérifie la propriété : $\forall \mu, \mu' \in [\![A]\!] : \mu < \mu' \Longrightarrow \mathcal{D}(\mu, *) < \mathcal{D}(\mu', *)$; il tient compte de la sémantique accordée aux treillis de nuances et nous l'utilisons pour évaluer le degré de généralité des ASN et générer un treillis de concepts, formé uniquement des concepts du treillis de Galois dont l'intension ne dépasse pas un seuil donné de généralité.

DÉFINITION 3.2 *Soient A un attribut structuré, $\mathcal{D}$ l'indice de distance défini sur $[\![A]\!]$ et $\mu \in [\![A]\!]$, on appelle degré de généralité de μ le réel*

$$1 - \mathcal{D}(\mu, *)$$

compris dans l'intervalle $[0,1]$. Si le degré de généralité de μ vaut 1 alors $\mu = *$. S'il vaut 0 alors $\mu = \perp$.

3.2.4 Définition du treillis engendré

Étant donné un contexte $(\mathcal{E}, [\![A]\!], \rho)$, soit $\mathcal{G}$ le treillis des concepts de ce contexte, on note $\widetilde{\mathcal{G}}$ le treillis composé uniquement des concepts de $\mathcal{G}$ dont l'intension a un degré de généralité inférieur à un seuil donné :

$$\widetilde{\mathcal{G}} = \{(*, \mathcal{E})\} \cup \{(I, E) \in \mathcal{G} \mid 1 - \mathcal{D}(I, *) < seuil\} \subset \mathcal{G}$$

PROPOSITION 3.4 *L'ensemble $\widetilde{\mathcal{G}}$ muni de la relation d'ordre induite par celle existant sur le treillis $(\mathcal{G}, <, \wedge, \vee)$ et des deux opérations $\widetilde{\wedge}$ et $\widetilde{\vee}$ définies par :*
$\forall (I,E),(I',E') \in \widetilde{\mathcal{G}}$,

1) si $(I,E) \wedge (I',E') \in \widetilde{\mathcal{G}}$ *alors* $(I,E)\widetilde{\wedge}(I',E') = (I,E) \wedge (I',E')$
sinon $(I,E)\widetilde{\wedge}(I',E') = (*, \mathcal{E})$

2) $(I,E)\widetilde{\vee}(I',E') = (I,E) \vee (I',E')$

est un treillis dont les opérations de borne inférieure et de borne supérieure sont respectivement $\widetilde{\wedge}$ et $\widetilde{\vee}$.

Preuve :
Soient (I,E) et (I',E') deux éléments de $\widetilde{\mathcal{G}}$, il faut montrer que $(I,E)\widetilde{\wedge}(I',E')$ est le plus grand des minorants de (I,E) et (I',E') dans $\widetilde{\mathcal{G}}$.

Notons Min l'ensemble des minorants de (I,E) et (I',E') dans $\mathcal{G}$ et $\widetilde{Min}$ l'ensemble des minorants de (I,E) et (I',E') dans $\widetilde{\mathcal{G}}$, on a $\widetilde{Min} = Min \cap \widetilde{\mathcal{G}}$.
Si $(I,E) \wedge (I',E') \in \widetilde{\mathcal{G}}$ alors $(I,E) \wedge (I',E') \in \widetilde{Min}$ et c'est bien le plus grand élément de $\widetilde{Min}$.
Si $(I,E) \wedge (I',E') \notin \widetilde{\mathcal{G}}$, par définition de $\widetilde{\mathcal{G}}$ cela signifie que $1 - \mathcal{D}(I \wedge I', *) \geqslant seuil$, donc d'après la proposition 3.1 :
$\forall (J,F) \in Min, (J,F) \neq (*, \mathcal{E})$ on a :
$(J,F) < (I,E) \wedge (I',E') \Rightarrow \mathcal{D}(J,*) < \mathcal{D}(I \wedge I', *) \Rightarrow 1 - \mathcal{D}(J,*) > 1 - \mathcal{D}(I \wedge I', *) \geqslant seuil \Rightarrow (J,F) \notin \widetilde{\mathcal{G}}$
l'ensemble $\widetilde{Min}$ est donc réduit au singleton $\{(*, \mathcal{E})\}$. $(*, \mathcal{E})$ est donc le seul minorant de (I,E) et (I',E') dans $\widetilde{\mathcal{G}}$ et donc la borne inférieure.

Soient (I,E) et (I',E') deux éléments de $\widetilde{\mathcal{G}}$. Montrons que $(I,E)\widetilde{\vee}(I',E')$ est le plus petit des majorants de (I,E) et (I',E') dans $\widetilde{\mathcal{G}}$.

- Notons Maj l'ensemble des majorants de (I,E) et (I',E') dans $\mathcal{G}$ et $\widetilde{Maj}$ l'ensemble des majorants de (I,E) et (I',E') dans $\widetilde{\mathcal{G}}$.
  Soit $(J,F) \in Maj$. Si (J,F) n'appartenait pas à l'ensemble $\widetilde{\mathcal{G}}$ alors, par définition on aurait $1 - \mathcal{D}(J,*) \geqslant seuil$, et par conséquent aucun élément (J',F') de $\mathcal{G}$ tel que $(J',F') < (J,F)$ ne serait dans $\widetilde{\mathcal{G}}$. En particulier, (J,F) étant un majorant de (I,E) et (I',E') dans $\mathcal{G}$, (I,E) et (I',E') ne seraient pas dans $\widetilde{\mathcal{G}}$.
  On a donc $Maj = \widetilde{Maj}$ et par conséquent $(I,E) \vee (I',E') = (I,E)\widetilde{\vee}(I',E')$ est la borne supérieure de (I,E) et (I',E') dans $\widetilde{\mathcal{G}}$.

Tout couple d'éléments de $\widetilde{\mathcal{G}}$ a donc une borne inférieure et une borne supérieure.
□

3.2.5 Principe de génération du treillis $\widetilde{\mathcal{G}}$

Pour construire le treillis de concepts $\widetilde{\mathcal{G}}$ à partir des entités, nous nous fondons sur la relation d'ordre existant sur $\widetilde{\mathcal{G}}$. Nous procédons par une mise à jour incrémentale du treillis selon un principe inspiré de l'algorithme de construction incrémentale du treillis de Galois décrit par Godin et al. [16].

Les concepts constituant $\widetilde{\mathcal{G}}$ sont ceux du treillis de Galois entier $\mathcal{G}$ qui ont un degré de généralité inférieur au seuil fixé. Un nouveau concept dans $\widetilde{\mathcal{G}}$ est obtenu à partir d'un concept (I,E) déjà présent dans $\widetilde{\mathcal{G}}$ et tel que la généralisation $I \wedge \rho(e)$ n'apparaît pas

comme intension d'un concept déjà dans $\widetilde{\mathcal{G}}$. Il peut y avoir plusieurs couples (I, E) différents générant cette nouvelle intension mais un seul est le générateur du nouveau concept.

DÉFINITION 3.3 *Soit $\rho(e)$ la description d'une nouvelle entité. Le concept $(I', E') \in \widetilde{\mathcal{G}}$ est générateur d'un nouveau concept (I, E) si et seulement si :*

l'intension I n'est présente dans aucun concept de $\widetilde{\mathcal{G}}$
et si (I', E') est le plus petit élément de l'ensemble $\{(I'', E'') \in \widetilde{\mathcal{G}} \mid I = I'' \wedge \rho(e)\}$
et si le degré de généralité $1 - \mathcal{D}(I' \wedge \rho(e), *)$ de (I, E) est inférieur au seuil fixé.

Remarque : Si le générateur (I', E') d'un nouveau concept (I, E) dans le treillis $\mathcal{G}$ n'appartient pas à $\widetilde{\mathcal{G}}$, l'ensemble $\{(I'', E'') \in \widetilde{\mathcal{G}} \mid I = I'' \wedge \rho(e)\}$ n'est pas forcément vide, cependant aucun de ces éléments n'est générateur du concept (I, E). En effet, un nouveau concept étant plus général que son générateur, si (I', E') n'est pas dans $\widetilde{\mathcal{G}}$ car d'un degré de généralité supérieur au seuil fixé, alors le concept (I, E) lui aussi n'appartient pas à $\widetilde{\mathcal{G}}$.

Entrées : Le treillis $\widetilde{\mathcal{G}}$ pour un ensemble d'entités $\mathcal{E}$, et e une nouvelle entité de description $\rho(e)$.
Sortie : Mise à jour de $\widetilde{\mathcal{G}}$ après avoir pris en compte e.

1. Si $1 - \mathcal{D}(\rho(e), *) > \textit{seuil}$, alors la mise à jour de $\widetilde{\mathcal{G}}$ consiste uniquement à ajouter l'entité e dans l'extension du concept $(*, \mathcal{E})$.
2. Les sommets (I, E) de $\widetilde{\mathcal{G}}$ tels que $I \leqslant \rho(e)$ sont mis à jour en ajoutant l'entité e dans leur extension E.
3. Si $(I, E) \in \widetilde{\mathcal{G}}$ est un générateur et si $1 - \mathcal{D}(I \wedge \rho(e), *) < \textit{seuil}$, alors on génère le nouveau concept $(I \wedge \rho(e), E \cup \{e\})$ et on met à jour les arcs.

L'idée pour créer les nouveaux concepts et mettre à jour les arcs du treillis $\widetilde{\mathcal{G}}$ est d'utiliser la définition précédente afin de trouver les générateurs. En effet, le générateur d'un nouveau concept étant le sommet le plus général qui produit une nouvelle intension par généralisation avec $\rho(e)$, nous sommes certain qu'il existe un arc – qui n'est pas un arc de transitivité – entre le générateur et le nouveau concept.

Dans le cas classique [16] où l'on calcule un treillis de Galois entre un ensemble d'entités et un ensemble d'attributs binaires, l'algorithme produit les nouveaux sommets concept en essayant systématiquement de générer une nouvelle intersection à partir de chaque concept (I, E) déjà présent dans le treillis en calculant l'intersection des ensembles d'attributs I et $\rho(e)$. Pour ce faire, les sommets sont d'abord classés par cardinalité croissante de leur intension. Cela permet de s'assurer que le premier sommet rencontré produisant une nouvelle intersection est un générateur.

Dans notre cas, l'intension d'un concept n'est pas un ensemble d'attributs mais la borne inférieure des valeurs représentant les entités de son extension. Afin de trouver les générateurs, nous classons les sommets du treillis $\widetilde{\mathcal{G}}$ par niveau croissant de spécificité et nous parcourons le treillis depuis les concepts les plus généraux vers les concepts les plus spécifiques. La définition suivantes des niveaux de spécificité nous assure que les concepts appartenant à un niveau donné forment une anti-chaîne, ainsi en parcourant le treillis, le premier sommet rencontré qui produit une nouvelle intension -dont le degré de généralité est inférieur au seuil fixé- est le plus général possible, c'est bien le générateur.

DÉFINITION 3.4 *Dans le treillis $\widetilde{\mathcal{G}}$, un sommet appartient à un niveau de spécificité n si et seulement si la chaîne la plus longue entre le plus petit élément du treillis $(*, \mathcal{E})$ et le sommet C est de longueur n.*

3.2.6 Algorithme de construction du treillis $\widetilde{\mathcal{G}}$

Au départ, $\widetilde{\mathcal{G}}$ est constitué du concept le plus général, dont l'intension est totalement indéterminée et dont l'extension contient toutes les entités possibles, et du concept le plus spécifique dont l'extension est vide.

La construction de $\widetilde{\mathcal{G}}$ se fait par des mises à jour successives au fur et à mesure que les descriptions $\rho(e)$ de nouvelles entités e sont disponibles. Cette mise à jour s'effectue selon l'algorithme suivant :

Procedure AjouteSeuil(e, $\rho(e)$, $\widetilde{\mathcal{G}}$, *seuil*)
Si le degré de généralité de $\rho(e)$ est supérieur à *seuil* alors
Ajouter e dans l'extension du concept $(*, \mathcal{E})$
FIN de AjouteSeuil
Finsi
Classer les sommets de $\widetilde{\mathcal{G}}$ par niveau de spécificité
$M \leftarrow \emptyset$ {contient les concepts modifiés ou nouvellement créés au cours de la mise à jour}
Parcourir $\widetilde{\mathcal{G}}$ par niveau de spécificité croissant
Pour chaque concept $C = (I, E)$ du niveau courant
Si I est plus générale que la description $\rho(e)$ alors
Ajouter e dans E
Ajouter C dans la liste M
Si $I = \rho(e)$ alors FIN de AjouteSeuil Finsi
Sinon
Si le degré de généralité de $I \wedge \rho(e)$ est inférieur à *seuil*
et s'il n'existe pas de concept dans M dont l'intension est $I \wedge \rho(e)$ alors
Créer le nouveau concept $C_N = (I \wedge \rho(e), E \cup \{e\})$
Mettre à jour les arcs du treillis
Ajouter C_N dans M
Si $I \wedge \rho(e) = \rho(e)$ alors FIN de AjouteSeuil Finsi
Finsi
Finsi
Finpour
FIN de Ajouteseuil

3.3 Extraction d'un graphe de concepts

L'introduction d'un seuil de généralité dans le calcul du treillis de Galois permet effectivement d'en élaguer les concepts trop généraux, cependant le treillis obtenu peut contenir encore beaucoup de concepts. Afin d'optimiser la recherche de concepts intéressants, nous proposons un algorithme permettant d'extraire un graphe hiérarchique de concepts à partir d'un treillis de concepts. La construction d'une telle hiérarchie – au sens large – permet d'obtenir un « résumé » du treillis des concept dont la taille est au maximum le double du nombre d'observations prises en compte.

3.3.1 Mesure de similarité entre intensions de concepts

L'extraction des concepts est faite à l'aide d'une mesure de similarité sur les ASN, dont on souhaite qu'elle remplisse les objectifs suivants :

1. I_1 et I_2 étant les intensions de deux concepts dont on veut mesurer la similarité, si I_1 est plus générale que I_2 ou inversement, alors la similarité entre ces deux concepts

doit être maximale, cela afin que la relation de généralisation/spécialisation existant sur les intensions des concepts soit vérifiée dans la hiérarchie extraite.

2. Elle doit être cohérente avec la sémantique des treillis de nuances, c'est à dire que pour un attribut simple donné, la similarité entre deux valeurs de cet attribut doit être d'autant plus faible qu'elles contiennent de nuances incomparables pour des modalités identiques.
3. Elle doit tenir compte de la structure arborescente des descriptions.

Pour respecter la cohérence avec la sémantique des treillis de nuances, nous définissons une mesure de similarité locale au niveau des attributs simples.

DÉFINITION 3.5 *Étant donné un attribut simple* $(D, V, [\![A]\!])$, *de domaine* $D = \{d_1, \ldots, d_q\}$ *et deux valeurs qualifiées* $\mu = ((d_1, v_1) \ldots (d_q, v_q))$ *et* $\mu' = ((d_1, v'_1) \ldots (d_q, v'_q))$ *avec* $v_i \in V$ *et* $v'_i \in V$, *on note :*

- $n_=$ *le nombre de modalités telles que leurs nuances dans* μ *et* μ' *sont égales :*
  $n_= = |\{d_i \in D | v_i = v'_i\}|$
- $n_<$ *le nombre de modalités dont la nuance dans* μ *est inférieure (au sens de la relation d'ordre du treillis* V*) à la nuance dans* μ' *:*
  $n_< = |\{d_i \in D | v_i < v'_i\}|$
- $n_>$ *le nombre de modalités dont la nuance dans* μ *est supérieure à la nuance dans* μ' *(i.e. dont la nuance dans* μ' *est inférieure à la nuance dans* μ*) :*
  $n_> = |\{d_i \in D | v_i > v'_i\}|$
- $n_\neq$ *le nombre de modalités dont la nuance dans* μ *est incomparable à la nuance dans* μ' *:*
  $n_\neq = |\{d_i \in D | v_i \not\leq v'_i \text{ et } v_i \not> v'_i\}| = |D| - n_= - n_< - n_>$

PROPOSITION 3.5 *La fonction de ressemblance* $\sigma : V^D \times V^D \to [0, 1]$ *définie par :*

$$\forall \mu_A, \mu'_A \in V^D \ : \ \sigma(\mu_A, \mu'_A) = \frac{n_= + max(n_<, n_>)}{|D|}$$

est une mesure de similarité.

On peut facilement vérifier que cette mesure satisfait les deux premiers objectifs fixés : elle est maximale lorsque les valeurs sont identiques ou lorsque l'une est plus générale que l'autre ; elle est nulle si et seulement si les valeurs sont complétement incomparables et plus ces valeurs contiennent de nuances incomparables, plus leur similarité est faible.

Pour la mesure de similarité entre ASN, on tient compte de la structure arborescente de celles-ci en affectant des poids aux attributs qui les composent ; un attribut structuré ayant un poids égal à la somme des poids des attributs qui le forment. Pour le choix de ces poids, on peut par exemple suivre ce qu'on fait instinctivement lorsqu'on compare deux descriptions structurées, c'est à dire qu'on donne une importance plus grande aux parties les plus génériques. Cela revient à supposer que toutes les sous-parties d'une même partie sont aussi significatives les unes que les autres. On pourrait aussi affecter ces poids en selon l'importance accordée aux différents attributs selon le point de vue de l'utilisateur.

PROPOSITION 3.6 *Étant donné* A *un attribut (simple ou structuré) et* $\mu_A, \mu'_A \in [\![A]\!]$ *deux valeurs de* A, *la fonction* $S : [\![A]\!] \times [\![A]\!] \to [0, 1]$ *définie ci-dessous est une mesure de similarité :*

1) si $A = (D, V, T)$ *est un attribut simple*

$$S(\mu_A, \mu'_A) = \sigma(\mu_A, \mu'_A)$$

2) si $A = \langle A_1, \ldots, A_k \rangle$ *est un attribut structuré, on a* $\mu_A = \langle \mu_{A_1}, \ldots, \mu_{A_k} \rangle$ $\mu'_A = \langle \mu'_{A_1}, \ldots, \mu'_{A_k} \rangle$ *et*

$$S(\mu_A, \mu'_A) = \frac{1}{ps(A)} \cdot \sum_{i=1}^{k} ps(A_i).S(\mu_{A_i}, \mu'_{A_i})$$

où $ps(A_i)$ *désigne le poids de l'attribut* A_i.

3.3.2 Principe d'extraction du graphe de concepts

Le principe d'extraction d'un graphe à partir des concepts les plus spécifiques du treillis de Galois et d'utiliser une méthode similaire à celle de la construction d'une hiérarchie binaire.

Entrée : Un treillis de concepts $\mathcal{G}$
Sortie : Un graphe hiérarchique de concepts $\mathcal{H}$
Initialisations : Au départ $\mathcal{H}$ est constitué de l'ensemble des concepts les plus sécifique de $\mathcal{G}$: $Spec(\mathcal{G})$ et du concept le plus général de $\mathcal{G}$, c'est à dire $(*, \mathcal{E})$ qui est le parent de tous les autres.

La construction se fait en maintenant une liste de concepts actifs –initialisée avec $Spec(\mathcal{G})$– parmi lesquels on sélectionne à chaque étape les deux concepts C_1 et C_2 les plus similaires.

Si la similarité entre C_1 et C_2 est égale à 1, cela peut signifier que l'un des concepts est plus général que l'autre, par exemple C_1. Dans ce cas, le parent commun à C_1 et C_2 est C_1. On conserve ce dernier dans la liste des concepts actifs et on désactive C_2.

Si C_1 et C_2 ne sont pas comparables, alors on on active le concept $C_1 \wedge C_2$ et on désactive C_1 et C_2.

Enfin, à chaque étape, il faut mettre à jour les liens d'héritages entre les deux concepts sélectionnés et leur parent, ainsi qu'entre ce parent et les autres concepts déjà présents dans $\mathcal{H}$.

L'algorithme se termine lorsque la liste des concepts actifs est vide ou bien s'il ne reste qu'un concept actif, que l'on place alors dans la hiérarchie.

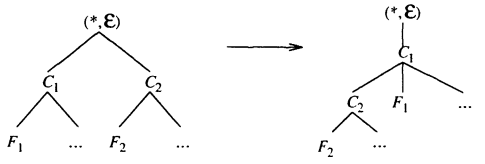
Lors de la selection des concepts les plus similaires, en cas d'égalité, si la similarité pour chaque couple est de 1, alors on choisit le couple tel que l'un des concepts est plus général que l'autre. Si plusieurs couples sont dans ce cas, on choisit le couple tel que le degré de généralité (c.f définition 3.2) de leur parent commun est le moins élevé. Et s'il y a encore le choix entre plusieurs couples, on sélectionne le premier rencontré.

Dans tous les autres cas d'égalité (similarité différente de 1), on choisit le couple dont le parent commun a le degré de généralité le moins élevé et s'il reste plusieurs couples possibles alors on choisit le premier rencontré.

À chaque étape de la construction, l'ensemble des concepts actifs est l'ensemble des concepts qui ont comme unique parent dans la hiérarchie le concept $(*, \mathcal{E})$. Dès qu'un nouveau concept C est introduit, il faut mettre à jour les liens le concernant ainsi que ceux des deux concepts C_1 et C_2 qui ont été sélectionnés et dont le parent devient C. Plusieurs cas de figure se présentent pour la mise à jour de ces liens.

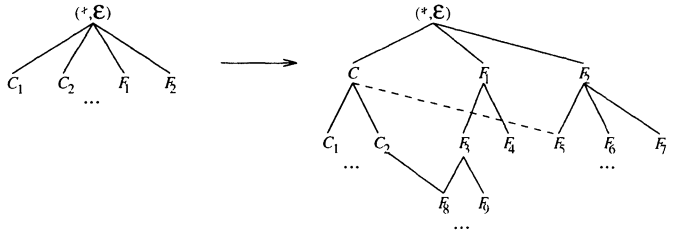
Premier cas : Le parent commun à C_1 et C_2 est le concept $(*, \mathcal{E})$. Il n'y a alors aucun lien à modifier, car C_1 et C_2 étant des concepts actifs, ils ont déjà comme parent $(*, \mathcal{E})$. On désactive C_1 et C_2 et on passe à l'étape suivante.

Deuxième cas :



C_1 et C_2 ont une similarité égale à 1 et l'un des deux (par exemple C_1) est plus général que l'autre qui devient son fils. On supprime le lien $(*, \mathcal{E}) \rightarrow C_2$, on ajoute le lien $C_1 \rightarrow C_2$, on désactive C_2 et on laisse C_1 actif.

Troisième cas :



Le parent C de C_1 et C_2 est nouveau dans la hiérarchie. C_1 et C_2 deviennent inactifs, C devient actif, on supprime les liens $(*, \mathcal{E}) \rightarrow C_1$ et $(*, \mathcal{E}) \rightarrow C_2$, on ajoute les liens $C \leftarrow C_1$, $C \leftarrow C_2$ et $(*, \mathcal{E}) \rightarrow C$.

C étant nouveau, il faut alors ajouter les liens hiérarchiques qui peuvent exister entre C et les autres concepts inactifs de la hiérarchie. On doit donc rechercher parmi les descendants des frères de C les concepts qui sont plus spécifiques que C et qui n'ont pas déjà un parent plus spécifique que C.

Par exemple, sur la figure ci-dessus, l'introduction de C dans la hiérarchie entraîne l'ajout du lien $C \rightarrow F_5$ mais pas du lien $C \rightarrow F_8$.

4 CLASSIFICATION ET APPROCHE ORIENTÉE OBJETS

Les algorithmes présentés sont implantés en C++ et ont servi à l'analyse de données aussi bien tabulaires que structurées (les éponges du modèle Hyalomena). Il sont en cours d'intégration dans le système IKBS† [7]. Les apports de l'approche orientée objets ont concerné :

1. La représentation de l'information. Le formalisme des ASN est inspiré des modèles de représentation de données en Base de données orientées objets [1]. L'attribut dit structuré correspond à un attribut de type *tuple*.
2. L'utilisation intensive du *polymorphisme*. On désigne par ce terme la possibilité à une fonction (appelée méthode dans un langage orienté objet), d'avoir des comportements dépendant de l'objet receveur. Ceci est particulièrement intéressant dans notre cas où les opérateurs $\wedge, \vee$, les différentes mesures de similarités sont calculés de manière différente selon les objets auquels ils s'appliquent. Par exemple, pour calculer $a \wedge b$, on écrira $a \rightarrow inf(b)$ et le système choisira automatiquement le code adapté selon que a et b sont des valeurs de treillis de nuance, des valeurs qualifiés d'un attribut simple ou structuré (le calcul est alors récursif) ou une ASN. L'écriture du code est élégant et concis car on n'a pas à distinguer ces différents cas.

† Iterative Knowledge Base System, `www.univ-reunion.fr/~ikbs`

4.1 Application à des données tabulaires

Dans ce premier exemple, les données sont issues d'une enquête portant sur un ensemble de questions visant à caractériser les individus qui préfèrent certains parfums féminin et masculin.

Les individus sont décrits à partir d'un ensemble de 22 variables qualitatives. Les deux premières indiquent le parfum masculin et le parfum féminin préféré. Les quatre suivantes : *sexe*, *age*, *situation familiale* et *profession* caractérisent les individus, et les 16 autres prennent en compte la position des individus sur des questions d'ordre général.

Les 16 questions d'ordre général sont très intéressantes car elles ont toutes le même ensemble de valeurs : {*tout à fait d'accord, plutôt d'accord, ne se prononce pas, plutôt contre, tout à fait contre*}. Dans les données originales, les variables correspondantes sont des variables qualitatives dont le domaine n'a pas de structure. Avec le modèle des ASN, nous transposons facilement ces données en utilisant le treillis de nuances V_4 (cf figure 2). L'intérêt est de pouvoir représenter le fait qu'accord et désaccord ne sont pas des valeurs comparables, information qui n'existe pas si l'on choisit une chaîne ou un domaine non structuré par exemple.

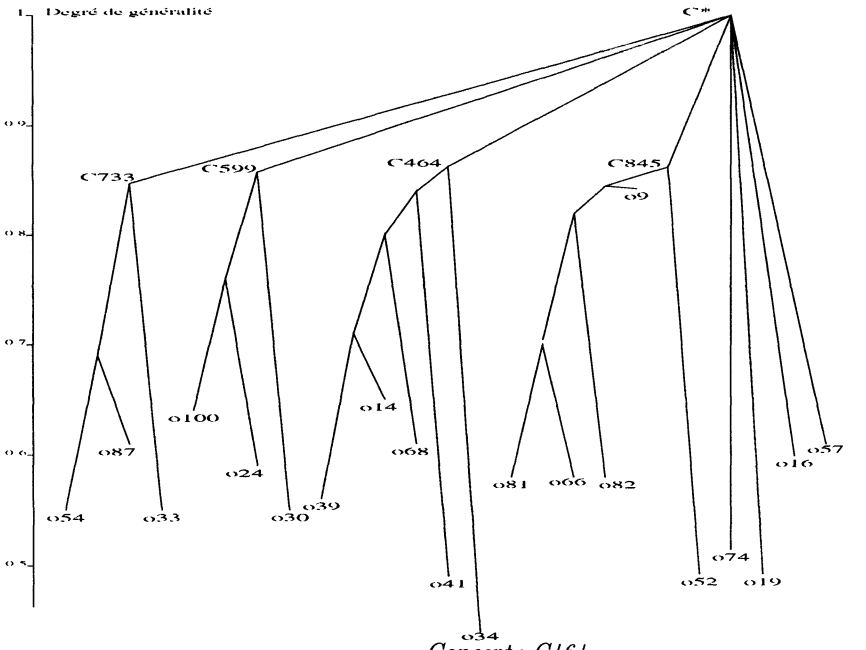
Les tableaux qui suivent donnent la description des 22 attributs simples utilisés pour représenter les données de l'enquête. Pour les six premiers, le treillis de nuances associé est le treillis V_2, pour les autres, le treillis V_4.

Attribut	Identificateur	Domaine
homm	Parfum masculin	{brut for men, habit rouge, drakkar noir, jacomo, jules}
femm	Parfum feminin	{cristalle, diorescence, shalimar, arpège, eau de campagne}
sexe	Sexe	{masculin, féminin}
age	Age	{[15-24], [25-34], [35-44], [45-54], 55 et plus}
situ	Situation de famille	{célibataire, autre, marié}
prof	Profession	{cadre sup/professeur/..., cadre moyen/technicien, artisan/ouvrier/employé, étudiant/sans profession}

Attribut	Identificateur	Domaine
avor	AVOR	{Il faut libéraliser l'avortement}
dieu	DIEU	{Dieu existe}
reve	REVE	{Il faut égaliser les revenus}
heri	HERI	{Il faut limiter les héritages}
pub	PUB	{La publicité est indispensable}
fran	FRAN	{Il faut acheter français}
pein	PEIN	{Il ne fallait pas supprimer la peine de mort}
cord	CORD	{Il fallait construire le concorde}
jeun	JEUN	{Il faut tout faire pour la jeunesse}
nata	NATA	{Il faut encourager la natalité}
cons	CONS	{On doit défendre le consommateur}
fami	FAMI	{La famille est une bonne chose}
noce	NOCE	{Je suis contre le mariage}
conv	CONV	{Il faut respecter les convenances}
retr	RETR	{Il faut donner la retraite plus jeune}
libe	LIBE	{Il faut encourager la libération de la femme}

Pour un échantillons aléatoire de 20 individus, et l'élagage du treillis de Galois a un seuil de généralité à 0.86. Le treillis obtenu est constitué de 978 concepts. La figure 7 montre le graphe hiérarchique de concepts extrait de ce treillis, qui dans le cas présent est une arborescence. Nous avons placé les différents concepts du graphe selon leur degré de généralité.

Remarquons d'abord que les individus pris en compte ont tous un degré de généralité compris entre 0.46 et 0.6 (hormis l'individus *o9* dont le degré vaut 0.85). Cela signifie que



Concept : C845
Extension : {*o52 o66 o81 o82 o9*}
Intension :
⟨*Description :*
⟨*Parfum masculin : (habit rouge, 0)*
(drakkar noir, 0) (jules, 0)⟩
⟨*Parfum feminin : (diorescence, 0)*⟩
⟨*Sexe : (masculin, 0)(feminin, 1)*⟩
⟨*Age : ([35-44], 0)([45-54], 0)*⟩
⟨*Profession : (cadre sup professeur..., 0)*
(cadre moyen technicien, 0)⟩
⟨*AVOR : (Il faut liberaliser l'avortement, +)*⟩
⟨*HERI : (Il faut limiter les heritages, -)*⟩
⟨*JEUN : (Il faut tout faire pour la jeunesse, +)*⟩
⟨*FAMI : (La famille est une bonne chose, +)*⟩
⟨*LIBE : (Encourager la lib. de la femme, +)*⟩⟩

Concept : C599
Extension : {*o100 o24 o30*}
Intension :
⟨*Description :*
⟨*Parfum masculin : (jacomo, 0)(jules, 0)*⟩
⟨*Parfum feminin : (diorescence, 0)(arpege, 0)*
(eau de campagne, 0)⟩
⟨*Sexe : (masculin, 1)(feminin, 0)*⟩
⟨*Age : ([35-44], 0)(55 et +, 0)*⟩
⟨*Situation de famille : (celibataire, 1)(autres, 0)*
(marie, 0)⟩
⟨*Profession : (cadre sup professeur..., 0)*⟩
⟨*AVOR : (Il faut liberaliser l'avortement, +)*⟩
⟨*PUB : (La plublicite est indispensable, -)*⟩
⟨*RETR : (Donner la retraite plus jeune, +)*⟩
⟨*LIBE : (Encourager la lib. de la femme, +)*⟩⟩

Concept : C464
Extension : {*o14 o34 o39 o41 o68*}
Intension :
⟨*Description :*
⟨*Parfum masculin : (jacomo, 0)(jules, 0)*⟩
⟨*Parfum feminin : (shalimar, 0)*
(eau de campagne, 0)⟩
⟨*Age : ([15-24], 0)([45-54], 0)*⟩
⟨*Situation de famille : (celibataire, 0)(autres, 0)*
(marie, 1)⟩
⟨*Profession : (artisan ouvrier employe, 0)*
(etudiant sans prof, 0)⟩
⟨*REVE : (Egaliser les revenus, -)*⟩
⟨*HERI : (Il faut limiter les heritages, -)*⟩
⟨*PEIN : (Pas supprimer la peine cap., +)*⟩
⟨*CONS : (Doit defendre le consommateur, +)*⟩⟩

Concept : C733
Extension : {*o33 o54 o87*}
Intension :
⟨*Description :*
⟨*Parfum masculin : (brut for men, 0)*
(drakkar noir, 0) (jules, 0)⟩
⟨*Parfum feminin : (shalimar, 0)*
(eau de campagne, 0)⟩
⟨*Sexe : (masculin, 1)(feminin, 0)*⟩
⟨*Age : ([15-24], 0)([45-54], 0)(55 et +, 0)*⟩
⟨*Situation de famille : (celibataire, 0)(autres, 0)*
(marie, 1)⟩
⟨*Profession : (cadre moyen technicien, 0)*
(etudiant sans prof, 0)⟩
⟨*HERI : (Il faut limiter les heritages, -)*⟩
⟨*PEIN : (Pas supprimer la peine cap., -)*⟩
⟨*FAMI : (La famille est une bonne chose, +)*⟩⟩

FIG. 7 – *Données d'enquête – Graphe hiérarchique de concepts*

pour ces individus, environ la moitié des attributs sont indéterminés, c'est à dire qu'ils ont répondu « sans opinion » à la moitié des questions de l'enquête.

Le graphe hiérarchique obtenu permet de dégager quatre concepts – disjoints – qui semblent intéressants.

Précisons la manière d'interpréter l'intension d'un concept : celle-ci généralise toutes les descriptions des individus présents dans l'extension du concept, et du fait de la structure des treillis de nuances, si la valeur d'un attribut A de l'intension est qualifiée par la nuance n (par exemple « plutôt d'accord »), cela signifie que, pour un individu de ce concept, la valeur de l'attribut A peut être qualifiée par la même nuance n ou par n'importe quelle nuance plus spécifique (par exemple « plutôt d'accord » ou « tout à fait d'accord »).

En observant les descriptions de ces concepts, on peut facilement les interpréter :

C845 : Il s'agit de femmes de moins de 35 ans ou plus de 54 ans, qui ne sont pas cadres, qui n'aiment pas le parfum féminin Diorescence ni les parfums masculins Habit Rouge, Drakkar Noir et Jules. Ces femmes sont pour libéraliser l'avortement et pour encourager la libération de la femme ; elles sont contre la limitation des héritages, pensent qu'il faut tout faire pour la jeunesse et que la famille est une bonne chose.

C464 : Il s'agit d'hommes et de femmes mariés, qui ont entre 25 et 45 ans ou plus de 54 ans et qui sont cadres. Ils n'aiment pas les parfums masculins Jacomo et Jules, ni les parfums féminins Shalimar et Eau de Campagne. Ces personnes sont contre l'égalisation des revenus et la limitation des héritages et pensent qu'on doit défendre le consommateur et qu'il ne fallait pas supprimer la peine de mort.

C599 : Ce sont des hommes célibataires de moins de 35 ans, qui ne sont pas cadres supérieurs. Ils n'aiment pas les parfums masculins Jacomo et Jules, ni les parfums féminins Diorescence, Arpège et Eau de Campagne. Il sont pour encourager la libération de la femme, pour la libéralisation de l'avortement et pour donner la retraite plus jeune. Ils ne pensent pas que la publicité est indispensable.

C733 : Ce sont des hommes mariés, âgés de 25 à 45 ans, qui ne sont ni cadres moyens, ni étudiants, ni sans profession. Ils n'aiment pas les parfums Brut for men, Drakkar Noir, Jules, Shalimar et Eau de Campagne. Ils pensent que la famille est une bonne chose, qu'il ne faut pas limiter les héritages et sont contre la peine de mort.

Dans [19], on peut trouver les classes supérieures obtenues par une classification pyramidale [10] et une classification ascendante hiérarchique à partir des mêmes individus décrits dans le formalisme des objets symboliques [11].

En ce qui concerne la hiérarchie, on observe seulement la formation de deux classes dont une non-intéressante car montrant des préférences pour tous les parfums.

La pyramide laisse apparaître la formation de trois classes :

Classe 77 $\{o74\ o82\ o16\ o66\ o81\ o30\}$: il s'agit d'une classe formée d'hommes et de femmes qui ne sont pas cadres moyens, ont entre 15 et 34 ans ou plus de 44 ans, ne préfèrent pas le parfum Drakkar Noir pour homme ; ne préfèrent pas le parfum Diorescence pour femme ; sont contre l'avortement ; sont plutôt d'accord avec le fait qu'il faut tout faire pour la jeunesse ; sont en faveur de la libération de la femme ; croient que la famille est une bonne chose.

Classe 83 $\{o30\ o54\ o33\ o34\ o41\ o52\ o19\}$: il s'agit d'une classe formées d'hommes et de femmes qui ont plus de 25 ans ; sont célibataires ou mariés, ne préfèrent pas le parfum Eau de Campagne pour femme ; sont contre la limitation des héritages ; ne sont pas contre le fait que la famille est une bonne chose.

Classe 84 $\{o19\ o57\ o9\ o87\ o14\ o39\ o68\ o24\ o11\}$: il s'agit d'une classe formées d'hommes et de femmes, qui sont âgés de moins de 44 ans ou de plus de 55 ans ; sont célibataires ou mariés ; ne préfèrent pas Eau de Campagne pour femme, ne sont pas contre la libération de la femme ; ne sont pas contre le fait qu'on doit défendre le consommateur ; n'ont pas un position extrême sur le fait qu'il faut respecter les convenances.

Les classes issues de la pyramide n'ont rien à voir avec les concepts extraits du treillis de Galois. On peut remarquer cependant que les intensions des concepts sont plus spécifiques que les descriptions des classes. En effet les intensions des concepts montrent moins d'indétermination quant aux préférences sur les parfums : celles-ci sont connues pour au moins quatre des parfums sur les dix proposés alors que les descriptions des classes extraites de la pyramide ne permettent de connaître l'opinion des individus qui les constituent que pour un ou deux parfums.

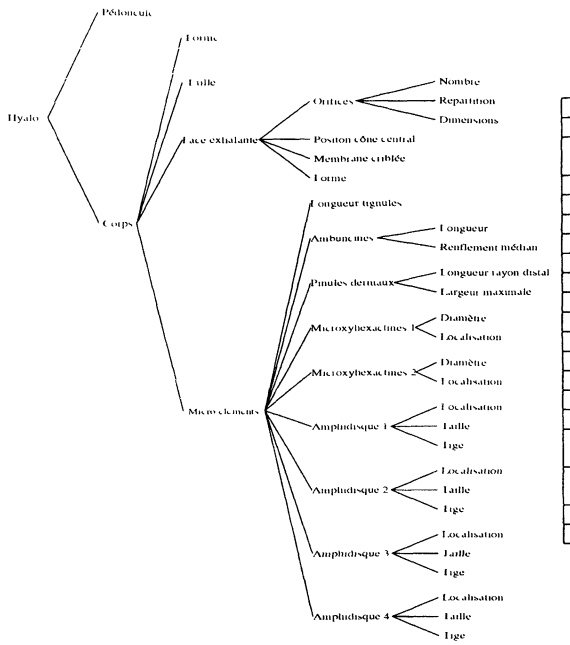
On remarque aussi que les trois classes sont constituées à la fois d'hommes et de femmes et ne permettent pas de mettre à jour un lien entre le sexe et les parfums. En revanche, les concepts *C599* et *C733* ne contiennent que des hommes et *C845* que des femmes, laissant voir des préférences différentes selon le sexe.

Sur un même ensemble d'observations, l'extraction d'une hiérarchie à partir d'un treillis de Galois permet d'obtenir des concepts qui sont à la fois plus spécifiques et plus discriminants que les classes obtenues par une construction pyramidale ou une hiérarchie binaire.

La spécificité résulte de la manière d'agréger les concepts, puisqu'en recherchant les concepts parents les plus proches on favorise l'inclusion et on minimise la variation interclasse. Et si les intensions sont plus discriminantes c'est qu'en utilisant un treillis de nuances dans lequel accord et désaccord sont incomparables, on évite d'agréger deux concepts dont la généralisation des intensions est trop indéterminée.

4.2 Application à des données structurées

Cet exemple traite de données structurées décrivant des éponges du genre *Hyalonema*. Les descriptions initiales sont des instanciations d'un modèle défini par Noël Conruyt[6].



Attribut	Domaine
Pédoncule	{*torsadé*}
Forme (Corps)	{*subcylindrique, renflée, évasée, aplatie, mal définie*}
Taille (Corps)	{*petit, moyen, grand*}
Nombre (Orifices)	{*unique, quatre, multiple*}
Répartition (Orifices)	{*homogène, en plage*}
Dimensions (Orifices)	{*homogène, hétérogène*}
Position cône central	{*saillant*}
Membrane criblée	{*distincte*}
Forme (Face exhalante)	{*plane, déprimée, invaginée*}
Longueur tignules	{*court, moyens, long*}
Longueur (Ambuncines)	{*court, moyens, long*}
Renflement médian	{*présent*}
Longueur rayon distal	{*court, moyen, long*}
Largeur maximale	{*étroit, moyen , large*}
Diamètre (Microxyhexactines)	{*petit, moyen, grand*}
Localisation (Microxyhexactines)	{*en paquets, éparse, périphériques*}
Localisation (Amphidisques)	{*toute région, périphérique, à l'intérieur*}
Taille (Amphidisques)	{*petit, moyen, grand*}
Tige	{*tuberculée, oblique*}

FIG. 8 – *Description de la structure des données Hyalonema*

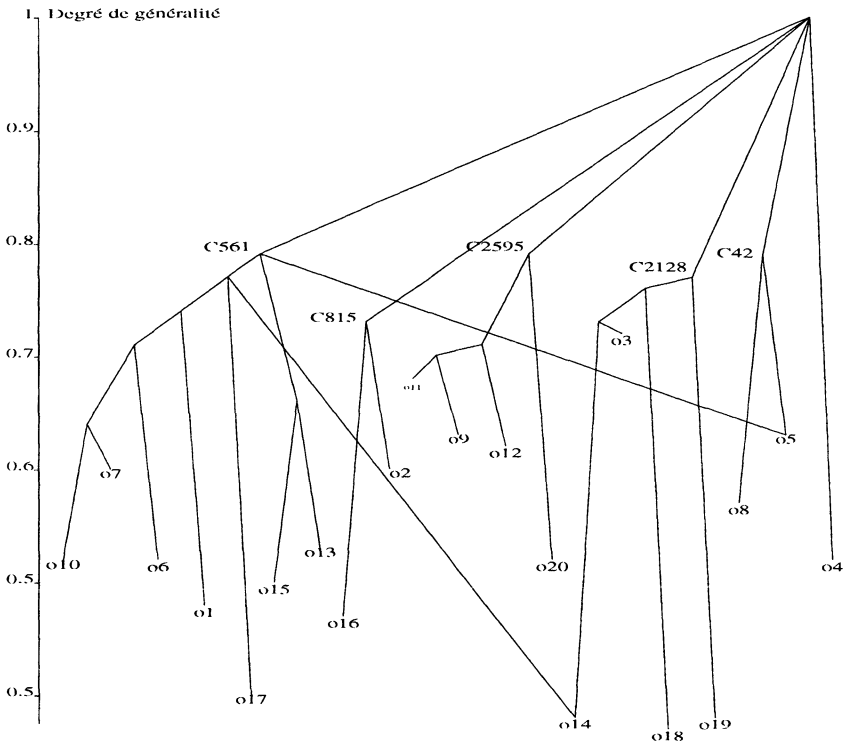


FIG. 9 – *Données Hyalonema – Graphe hiérachique de concepts*

À chacun de ces attributs est associé le treillis de nuances V_2 qui permet de représenter la présence ou l'absence d'une caractéristique et de prendre en compte les valeurs inconnues et inapplicables.

Remarquons que si l'on veut traiter ces données par des méthodes classiques d'analyse des données, par exemple la construction d'une hiérarchie binaire, on est amené à coder les 30 attributs à l'aide de 49 variables correspondant aux modalités. Se pose alors le problème de l'inapplicabilité des attributs (certaines parties d'éponge peuvent ne pas exister sur certains spécimen) qui introduit des valeurs manquantes dans le tableau des données ; valeurs qu'on ne sait pas traiter. En outre un tel codage ne permet pas de tenir compte des contraintes qui existent entre les valeurs des attributs.

Pour un échantillons aléatoire de 20 descriptions d'éponges, avec un seuil de généralité égal à 0.8, le treillis obtenu est formé de 277 concepts. La figure 9 montre le graphe hiérarchique de concepts que l'on extrait du treillis. On a étiqueté sur ce graphe les concepts dont l'intension ne contient qu'une observation par l'identificateur de cette observation. Sur la figure 10 on peut voir l'ASN de l'intension du concept C2595 qui est constitué d'éponges n'ayant que trois types d'amphidisques ; l'attribut Amphidisque4 est donc inapplicable.

En observant le graphe de concepts obtenu, on peut partitionner l'ensemble des observations en 5 classes, correspondant aux concepts les plus généraux : C561, C815, C2595, C2128 et C42. Remarquons que ces concepts ne sont pas tous disjoints. En effet, l'observation $o5$ appartient à la fois au concept C42 et au concept C561.

5 CONCLUSION

Dans cet article, nous avons d'abord proposé un modèle de représentation de données qui présente les avantages suivant :

- il offre un cadre unique pour la représentation de données structurées ou tabulaires.

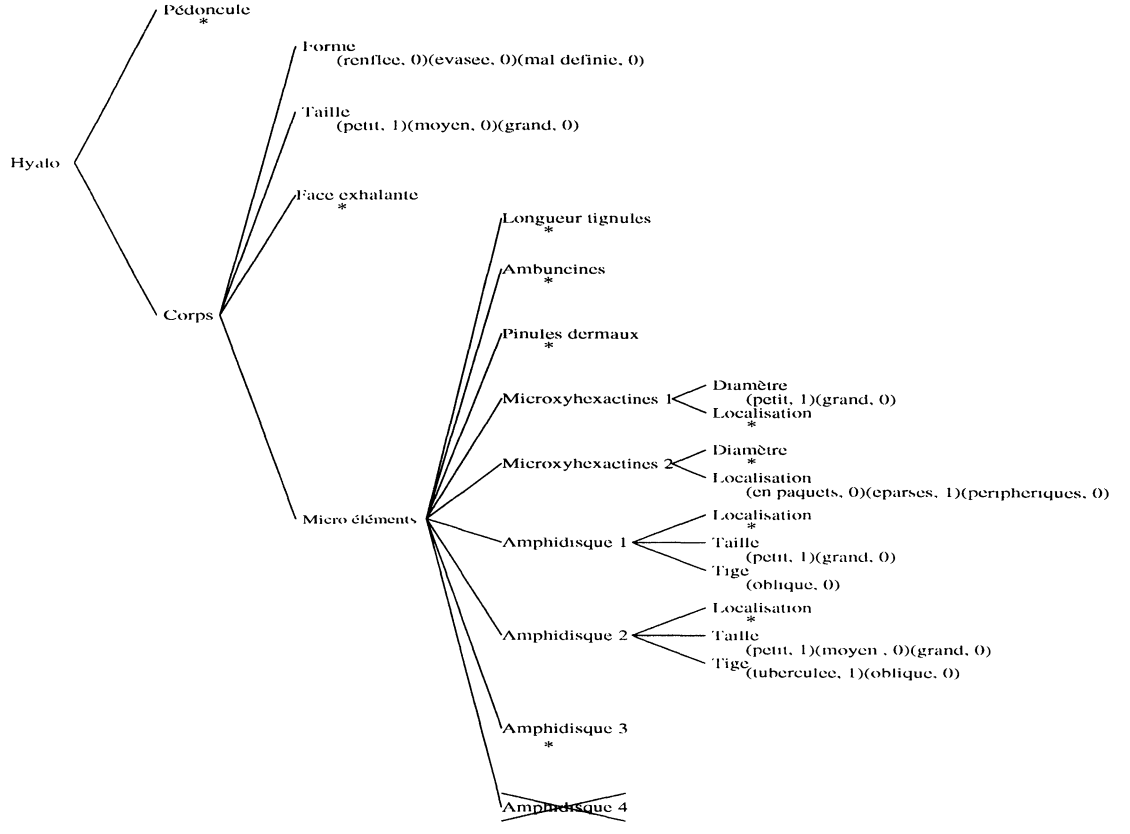


FIG. 10 – *Données Hyalonema – Intension du concepts C2595*

- il permet de traiter aussi bien des données précises qu'imprécises au travers d'un choix adéquat de treillis de nuance. Ces treillis autorisent les valeurs « inconnue », « absente » ou « impossible ».
- il prend en compte les contraintes éventuelles reliant les valeurs des attributs des observations.

L'espace des observations a été muni d'une structure de treillis et une correspondance de Galois a été définie. Un algorithme incrémental a été proposé pour construire le treillis des concepts. Les concepts issus d'un tel treillis sont intéressants pour l'utilisateur car faciles d'interprétation et leurs instances respectent les contraintes initiales des données.

Cependant leur détermination se heurte à la taille du treillis qui est généralement importante pour des applications réelles. Des algorithmes permettant d'extraire un treillis de concepts plus petit ou un graphe conceptuel ont été fournis rendant opérationnel le système de classification. Les prolongements de ce travail sont nombreux. En restant dans le cadre de la classification conceptuelle, une direction possible est la recherche des concepts « imprécis » c'est-à-dire dont les éléments appartiennent au concept avec un certain degré. On pourrait aussi considérer d'autres structures de classification telles que les hiérarchies ou les partitions ou aborder d'autres problématiques comme la discrimination pour l'étude de données représentées par une arborescence nuancée.

BIBLIOGRAPHIE

[1] ABITEBOUL S., HULL R. et VIANU V., *Foundations of Databases*, New-York, Addison-Wesley Publishing Company Inc, 1995.

[2] BARBUT M. et MONTJARDET B., *Ordre et classification : Algèbre et Combinatoire – volume II*, Paris, Hachette, 1970.

[3] BEZDEK J.C., *Pattern Recognition with Fuzzy Objective Function Algorithms*, New-York, Plenum, 1981.

[4] BEZDEK J.C., « A rewiew of probabilistic, fuzzy, and neural models for pattern recognition », *J. intell. fuzzy syt.*, 1(1993), 1–25.

[5] BRISSAC O. et LIQUIERE M., « Gabels : Un système d'apprentissage construit sur un modèle d'hypergraphes », *Actes des 9e Journées Acquisition, Validation, Apprentissage*, 1995, 89–102.

[6] CONRUYT N., *Amélioration de la robustesse des systèmes d'aide à la description, à la classification et à la détermination des objets biologiques*, thèse, Université de Paris IX, 1994.

[7] CONRUYT N., GROSSER D. et RALAMBONDRAINY H., « IKBS : an interactive knowledge base system for improving description, classification and identification of biological objects », *Indo-French Workshop on Symbolic Data Analysis and its Applications*, Université de Paris IX – LISE – CEREMADE, 1997, 212–224.

[8] DANIEL-VATONNE M.C., *Les termes : un modèle de repésentation et structuration de données symboliques*, thèse, Université Montpellier II, 1993.

[9] DANIEL-VATONNE M.C. et DE LA HIGUERA C., « Les termes : un modèle de représentation et structuration de données symboliques », *Math. inform. sci. hum.*, 122(1993), 41–63.

[10] DIDAY E., « Une représentation visuelle des classes empiétantes : les pyramides », *R.A.I.R.O.*, 20(1986), 475–526.

[11] DIDAY E., « Introduction à l'approche symbolique en analyse des données », *Actes des journées Symboliques-Numériques pour l'Apprentissage de Connaissances à partir d'Observations*, 1987, 21–26.

[12] GANTER B., *Two basic algorithm in concept analysis*, Technical Report 831, Darmstadt, Technische Hochschule, 1984.

[13] GANTER B. et WILLE R., *Conceptual scaling*, Technical Report 1174, Darmstadt, Technische Hochschule, 1988.

[14] GINSBERG M.L., « Multivalued logics : a uniform approach to inference in artificial intelligence », *Computat. intell.*, 4(1988), 265–316.

[15] GIRARD R., *Classification conceptuelle sur des données arborescentes et imprécises*, thèse, Université de La Réunion, 1997.

[16] GODIN R., MISSAOUI R. et ALAOUI H., « Incremental concept formation algorithms based on Galois (concept) lattices », *Computat. intell.*, 11(1995), 246–267.

[17] JAPPY P., DANIEL-VATONNE M.C, DE LA HIGUERA C. et GASCUEL O., « Learning from recursive, tree structured examples », *Proceedings of the seventh European Conference on Machine Learning, LNAI 784*, Berlin, Springer, 1994, 367–370.

[18] LEBBE J., *Représentation des Concepts en Biologie et en Médecine*, thèse, Université Pierre et Marie Curie Paris IV, 1991.

[19] DE PINHO M.P., *Analyse de données symboliques - Pyramides d'héritage*, thèse, Université de Paris XI Dauphine, 1991.

[20] QUINLAN J.R., « Learning logical definitions from examples », *Machine Learning*, Los Altos, Morgan Kaufman, 5(1994), 239–266.

[21] RALAMBONDRAINY H., « Apprentissage dans le contexte d'un schéma de base de données », *Induction Symbolique et Numérique à Partir de Données*, Paris, Dunod, 1991, 241–255.

[22] SALLANTIN J., QUINQUETON J., BARBOUX C. et AUBERT J.P., « Les théorie semi-empririques: éléments de formalisation », *Rev. intell. artif.*, 5(1991), 93–107.

[23] WILLE R., « Restructuring lattice theory: an approach based on hierarchies of concepts », *Ordered Sets*, I.Rival Ed., 1982, 445–470.

[24] WINSTON P.H., « Learning structural descriptions from examples », *The psychology of Computer Vision*, New-York, MacGraw Hill, 1975, chapitre 5.