Sémantique et composition des règles d'adaptation d'un système de raisonnement à partir de cas - Vers la construction d'une base de règles d'adaptation

Matthieu Tixier

Sémantique et composition des règles d'adaptation d'un système de raisonnement à partir de cas

—

Vers la construction d'une base de règles d'adaptation

Mémoire

Soutenu le 26 Juin 2007

pour l'obtention du

Master Sciences de la Cognition et Applications
(Traitement Automatique des Langues)
co-habilité Université Nancy 2 – Université Henri Poincaré

par

Matthieu Tixier

Composition du jury

Abdel Belaïd, Professeur, Université Nancy 2 (responsable du master SCA)
Patrick Blackburn, Directeur de recherche, INRIA (spécialité TAL),
Christine Bourjot, Maître de conférence, Université Nancy 2 (spécialité SC),
Samuel Cruz-Lara, Maître de conférence, Université Nancy 2 (spécialité TMN)

Encadrants : Jean Lieber, Maître de conférence, Université Henri Poincaré
Amedeo Napoli, Directeur de recherche, CNRS

Mis en page avec la classe thloria.

Table des matières

Introduction

Quatre grands paradigmes de raisonnement peuvent être distingués en intelligence artificielle : la déduction, l'induction, l'abduction et l'analogie [Chouraqui, 1986]. La déduction, mode privilégié des raisonnements mathématiques et logiques, constitue sans doute une des thématiques les plus développées en IA, toutefois les formes de raisonnement non-déductif, comme l'analogie, nourrissent de nombreux travaux et applications avec des résultats tout aussi probant.

L'analogie est un mode de raisonnement répandu et naturel, il consiste à inférer des éléments concernant une situation particulière à partir d'éléments déjà connus issus d'une situation jugée analogue. Un médecin peut se remémorer un précédent diagnostic pour proposer un traitement à un nouveau patient présentant des symptômes similaires, un développeur informatique utilise souvent des modèles d'applications et programmes déjà rencontrés lors de précédentes réalisations pour concevoir un nouveau logiciel, pour ne citer que deux exemples.

Le raisonnement à partir de cas (RàPC) s'appuie sur le raisonnement par analogie en proposant d'exploiter l'expérience du système, des problèmes déjà résolus, afin de répondre à de nouvelles situations. Le RàPC est souvent présenté comme une alternative efficace aux systèmes à base de règles, les cas étant réputés être une source de connaissances plus simple à acquérir.

Le projet de recherche Kasimir a pour thématique la gestion des connaissances décisionnelles en cancérologie. L'aide à la décision est l'une des grandes dimensions de ce projet, le principe étant de proposer une recommandation thérapeutique appropriée à partir de la description d'un patient en s'appuyant sur les référentiels de bonnes pratiques utilisés par les professionnels de santé.

L'intégration d'un module de RàPC est une perspective de développement intéressante notamment dans l'optique d'être à même de proposer une recommandation lorsqu'un patient ne rentre pas dans le cadre des référentiels. Pour répondre à de telles situations le système doit rapprocher le patient hors-référentiel d'un cas similaire et *adapter* la recommandation thérapeutique du standard en s'appuyant sur ses connaissances d'adaptation (ca). L'adaptation est une étape clé en RàPC qui est restée délicate pendant longtemps en l'absence de méthodologies générale d'acquisition et d'application des ca [Hanney, 1997].

S'appuyant sur les recherches menées dans le champs de l'acquisition de connaissances d'adaptation (aca) pour pallier à ce problème, CabamakA est l'application d'aca semi-automatique développée pour le projet Kasimir. CabamakA utilise des méthodes d'extraction des connaissances des base de données (ecbd) afin d'acquérir les règles d'adaptation qui constituent une des sources privilégiées de ca pour le module de RàPC du projet Kasimir.

Cependant comme dans beaucoup d'applications d'ecbd [Pasquier, 2000, Zaki and Hsiao, 2002, G. *et al.*, 2006] l'étape d'interprétation et de validation par l'analyste des résultats demeure problématique du fait de la quantité et de la pertinence des règles extraites. Proposer des moyens permettant de limiter le nombre et la redondance des résultats comme le permettent les bases pour les règles d'association est un enjeu stratégique pour une pleine exploitation du RàPC dans Kasimir.

Développant une des perspectives de recherches pour CabamakA proposée dans [d'Aquin *et al.*, 2007], notre travail vise à apporter des éléments de réponses au problème de l'aide à l'interprétation et à la validation des règles d'adaptation. En s'appuyant sur les travaux menés dans le cadre classique d'extraction des règles d'association et en étudiant en détail les spécificités de notre cadre, l'idée est de proposer une méthode permettant de réduire l'ensemble des règles d'adaptation issues de l'étape de fouille de données afin de limiter le coût en terme d'expertise humaine de la validation des résultats.

Ce mémoire est organisé comme suit. Dans un premier temps, le contexte scientifique de cette recherche sera présenté et l'on introduira les notions clés pour notre propos issues des champs du RàPC et de l'ecbd. Puis, ces concepts seront resitués dans le contexte applicatif du projet Kasimir et du système d'acquisition de connaissances d'adaptation CabamakA. Ensuite, les spécificités de notre cadre seront mises en évidence au travers de l'étude de la sémantique des règles d'adaptation et du travail qu'à nécessité sa définition. Enfin, la composition de règles d'adaptation sera proposée comme élément central pour la construction d'une base pour les règles d'adaptation. Pour finir nous évoquerons les apports et les limites de cette recherche ainsi que les perspectives qu'elle laisse ouvertes.

Chapitre 1

Contexte scientifique

Le contexte scientifique de cette recherche est l'acquisition de connaissances d'adaptation (ACA). Cette thématique est au coeur du développement des systèmes de raisonnement à partir de cas (RÀPC) et s'appuie, notamment, sur des principes et des techniques d'extraction de connaissances dans des bases de données (ECBD).
À un niveau plus général, la représentation des connaissances, jusque dans sa dimension opérationnelle, se pose comme une problématique transversale à ces deux domaines d'études. Ce chapitre vise à présenter le RÀPC et les méthodes mises en œuvre pour l'ACA dans notre cadre. Il s'agit ici d'introduire les notions clés sur lesquelles nous nous appuierons pour proposer des éléments de réponses à la problématique de l'aide à l'interprétation et à la validation des connaissances d'adaptation.

1.1 Le raisonnement à partir de cas - RàPC

Le RÀPC repose sur un mode de raisonnement naturel, le raisonnement par analogie. Il s'agit de résoudre de nouveaux problèmes en s'appuyant sur des problèmes déjà résolus disponibles en mémoire en exploitant les similarités ainsi que les différences entre problèmes pour proposer une nouvelle solution sur la base des solutions existantes. Nous faisons souvent appel à notre expérience pour trouver des solutions à des problèmes inédits. Un médecin peut s'appuyer sur le diagnostic d'un patient déjà rencontré pour en proposer un pour un nouveau patient. Un programmeur réexploite souvent des parties entière de programmes déjà codées pour une nouvelle application.

A la base du RÀPC on trouve la théorie des scripts qui est un modèle de fonctionnement cognitif développé notamment par R. C. Schank [Riesbeck and Schank, 1989]. Un script est une suite ordonnée de procédures qu'un individu utilise lorsqu'il est confronté à des situations typiques, l'exemple classique étant le script du restaurant : entrer dans le restaurant, trouver une place où s'asseoir, commander... En s'appuyant sur ses connaissances un individu peut adapter ces scripts pour répondre à de nouvelles situations s'éloignant plus ou moins des prototypes. On retrouve l'idée centrale en RÀPC d'utiliser l'expérience pour résoudre de nouveaux problèmes que celle-ci soit directement exploitable ou qu'elle nécessite des adaptations pour pouvoir proposer une solution adéquate.

Les systèmes de RÀPC sont présentés comme une alternative intéressante aux systèmes experts notamment pour le problème d'acquisition des connaissances. Il est souvent plus facile d'acquérir des cas, des problèmes accompagnés de leur solution que de modéliser les connaissances nécessaires à la résolution de ces problèmes. La disponibilité croissante de sources de données de grand volume dans bien des domaines rend la perspective d'exploiter la connaissance implicite portée par ces épisodes de résolution de problèmes particulièrement intéressante. Cependant ces systèmes de RÀPC doivent également être capable de proposer des solutions à de nouveaux problèmes non encore rencontrés, même si des problèmes analogues sont connus. C'est là tout l'enjeu de l'adaptation, étape clé du cycle du RÀPC et question centrale pour l'ACA.

1.1.1 Le processus du RàPC

Un système de RàPC s'appuie sur deux espaces propres au domaine d'application, `Problèmes` l'espace des problèmes et `Solutions` l'espace des solutions. Les problèmes sont associés aux solutions par une relation « a pour solution » `Problèmes` × `Solutions` qui est en général incomplètement connue. Un cas est un épisode de résolution de problème représenté par un couple (`pb`,`Sol(pb)`), associant un problème `pb` ∈ `Problèmes` à une solution `Sol(pb)` ∈ `Solutions`. Les choix de représentation des problèmes et solutions sont presque aussi nombreux que les domaines d'application des systèmes de RàPC. La description des problèmes doit être suffisamment expressive pour pouvoir juger de l'application d'un cas à un autre problème [Hanney, 1997]. La solution peut être exprimée de différentes façons : une liste de procédures (des recettes de cuisine pour CHEF (Hammond 1989)) ou une estimation de valeur (CARMA (Hastings 1995)), entres autres.

Une session de RàPC vise à proposer une solution `Sol(cible)` à un problème `cible` donné en s'appuyant sur un problème source, noté `srce`, déjà résolu, et sur sa solution `Sol(srce)`. Les couples (`srce`,`Sol(srce)`) sont appelés *cas sources* et l'ensemble des cas sources est appelé base de cas. La base de cas est une source de connaissances essentielle à tout système de RàPC et les deux étapes principales du raisonnement que sont la *remémoration* et l'*adaptation* s'appuient dessus.

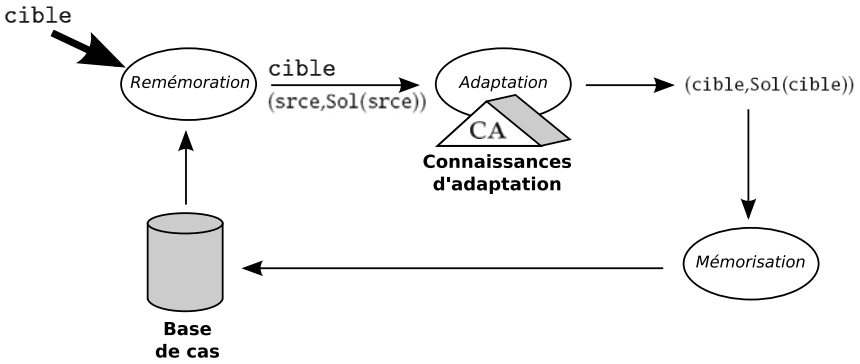

FIG. 1.1 – Schéma général d'un système de RàPC.

Le système commence par remémorer un cas (`srce`,`Sol(srce)`) sur la base d'une mesure de similarité entre les problèmes `srce` et `cible`. À ce niveau deux cas de figure peuvent se produire :

- soit il existe `srce` = `cible` et il suffit de proposer la solution `Sol(srce)` connue,
- soit on doit construire une solution `Sol(cible)` adaptée sur la base de `cible`, du cas remémoré et des connaissances d'adaptation du système.

La remémoration est un processus bien maîtrisé, il s'agit de définir un critère de similarité entre problèmes, lequel dépend des choix de représentation de ceux-ci dans l'application, et ainsi de retrouver dans la base de cas le problème `srce` le plus proche de `cible`. L'adaptation est plus délicate à mener et fait appel aux connaissances d'adaptation (CA) du système.

1.1.2 L'adaptation

Lorsqu'aucun cas source ne correspond exactement au problème `cible` proposé, le système doit construire une solution adaptée à partir de la solution `Sol(srce)` remémorée, des problèmes `srce` et `cible`, et des CA du système. Cette étape est réputée difficile et consiste souvent en un ensemble de méthodes propres à un système de RàPC et à son domaine d'application. Ces méthodes et connaissances sont donc difficilement transférables d'une système à un autre.

Les CA représentent les changements à opérer sur le cas source compte tenu du problème `cible` pour construire `Sol(cible)`. Elles constituent donc la pièce maîtresse de l'adaptation et leur

acquisition est stratégique. Il est délicat pour un concepteur d'application de RàPC d'être à même d'envisager *a priori* toutes les adaptations possibles que réclameront les problèmes que le système aura à résoudre, la question de l'*acquisition* des CA est donc omniprésente. Trouver une réponse à ces difficultés est la problématique centrale de l'ACA.

1.1.3 Acquisition de Connaissances d'Adaptation - ACA

Dans [Lieber *et al.*, 2004] une étude comparative de plusieurs travaux en ACA est présentée qui souligne, notamment, que deux approches émergent autour de la dimension supervisée et semi-automatique de l'acquisition de connaissances d'adaptation.

- La perspective supervisée met l'accent sur l'acquisition auprès d'experts, en s'appuyant notamment sur des modèles généraux du processus d'adaptation pour guider la conception des système RàPC.
- L'approche semi-automatique s'appuie en général sur la base de cas et cherche à mettre en évidence des principes généraux propres à l'adaptation dans le cadre du domaine d'application.

Toutes deux s'appuient sur les travaux portants sur les types et stratégies générales d'adaptation [1].

Cette recherche s'inscrit dans l'approche d'acquisition semi-automatique. CabamakA, l'application d'ACA utilisée dans notre cadre, s'appuie sur la perspective de recherche proposée par [Hanney, 1997] d'utiliser les méthodes d'extraction des connaissances dans des bases de données pour acquérir les CA par fouille de la base de cas.

1.2 Extraction de Connaissances dans des Bases de Données - ECBD

Les développements de l'informatique ces dernière décennies et sa diffusion croissante ont conduit à la constitution de nombreuses bases de données avec des records en terme de volume sans cesse surpassés. La question des outils et techniques permettant de manipuler et de valoriser ces données est un enjeu stratégique de plus en plus prégnant de nos jours [Fayyad *et al.*, 1996]. En effet, un expert par l'interprétation des données qu'il propose, qu'il s'agisse de transactions dans un super-marché ou de valeurs d'activation de gènes sur des bio-puces, met en lumière les connaissances intéressantes et exploitables cachées dans celles-ci.

Les données prises en elle-mêmes « ligne par ligne » ont un intérêt limité, savoir que le client 411 a acheté des œufs et du chocolat a une faible portée dans le contexte de la réorganisation des rayonnages d'une grande surface. Par contre mettre en évidence qu'un client achetant des oeufs achète dans 90% des cas également du chocolat se révèle très intéressant. Ainsi, on peut par exemple volontairement éloigner le rayon des oeufs de celui du chocolat pour optimiser le parcours des clients dans les rayonnages et maximiser le taux d'achat au mètre carré.

La découverte de telles connaissances demande une analyse fine et les volumes de données actuellement manipulés rendent illusoire la perspective de faire reposer cette valorisation sur la seule expertise humaine. L'ECBD offre un cadre permettant de mettre en évidence les relations entre données afin de faciliter le travail d'analyse et de répondre au besoin de traitement de volumes de plus en plus importants.

1.2.1 Le processus d'ECBD

Le diagramme suivant présente les différentes étapes clés du processus d'ECBD des sources de données jusqu'aux connaissances que l'on cherche à extraire. Les flèches en pointillés soulignent le

[1] On peut consulter [Lieber, 2006] et [Hanney, 1997] pour une revue détaillée des différentes taxonomies proposées.

fait que les connaissances extraites peuvent amener les différentes étapes à évoluer du fait des nouvelles connaissances découvertes et que l'analyste doit pouvoir intervenir en tous points du processus.

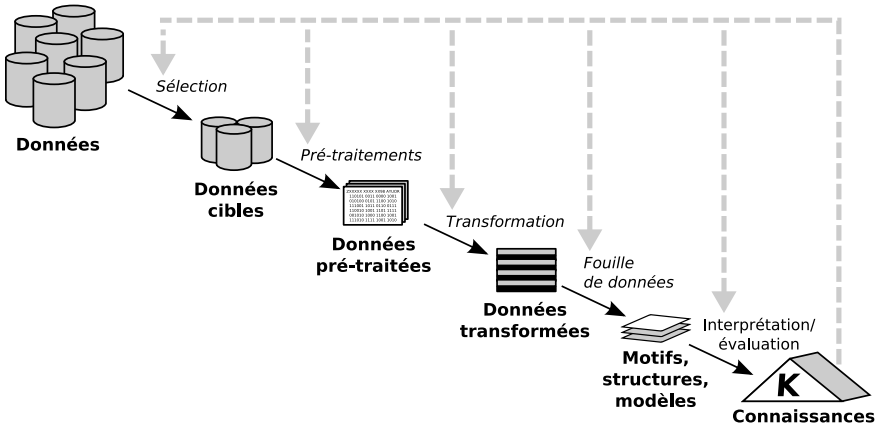

FIG. 1.2 – Le processus d'extraction de connaissances des bases de données. (d'après [Fayyad *et al.*, 1996])

L'ECBD est un processus d'extraction semi-automatique de connaissances, l'analyste peut intervenir pour piloter chaque étape. Les données pertinentes sont d'abord sélectionnées parmi toutes celles disponibles, elles subissent une première série de pré-traitements (élimination des enregistrements aberrants, corrections diverses, normalisation) visant à assurer leur compatibilité avec le module de formatage chargé de transformer les données dans un format adéquat pour l'étape de fouille.

La fouille de données peut-être assurée par plusieurs méthodes différentes. On distingue les techniques numériques (réseaux de neurones, analyse factorielle, entres autres), des techniques symboliques. Ce sont ces dernières qui nous intéresseront et qui sont exploitées par CABAMAKA. Il s'agira en s'appuyant sur les techniques d'extraction de motifs fermés fréquents et de règles d'association d'extraire des règles d'adaptation (*RA*) à destination d'une application de RÀPC. En bout de chaîne se trouve l'étape clé d'interprétation et de validation par l'analyste permettant de dériver des structures extraites les connaissances, objectif du processus dans son ensemble.

1.2.2 Extraction de règles d'association

Un contexte d'extraction est un triplet $(\mathcal{O}, \mathcal{I}, \mathcal{R})$, avec $\mathcal{O}$ un ensemble fini d'objets, $\mathcal{I}$ un ensemble fini de propriétés (que l'on appel aussi *items*) et $\mathcal{R}$, une relation binaire entre objets et propriétés ($\mathcal{R} \subseteq \mathcal{O} \times \mathcal{I}$). Un couple $(o, i) \in \mathcal{R}$, dénote le fait que l'objet $o \in \mathcal{O}$ à la propriété $i \in \mathcal{I}$ d'après $\mathcal{R}$. Un motif m est un ensemble de propriétés avec $m \subseteq \mathcal{I}$. On dit qu'un objet o contient un motif m si $\forall\, i \in m,\ \ (o, i) \in \mathcal{R}$.

Le support d'un motif m correspond au nombre d'objets de $\mathcal{O}$ contenant m.

$$support(m) = \frac{|\{o \in \mathcal{O} \mid \forall\, i \in m,\ \ (o, i) \in \mathcal{R}\}|}{|\{o \in \mathcal{O}\}|}$$

Etant donné un seuil minimal de support σ_{supp}, m est un *motif fréquent* si $support(m) \geqslant \sigma_{supp}$.

Un motif m est dit *fermé* s'il n'existe pas de motif n extrait de $\mathcal{D}$ tel que $n \supsetneq m$ et $support(n) = support(m)$.

Une règle d'association est une implication $m_1 \rightarrow m_2$ entre deux motifs $m_1, m_2 \subseteq \mathcal{P}$, telles que $m_1 \cap m_2 = \emptyset$. La confiance d'une règle d'association $r : m_1 \rightarrow m_2$ est la probabilité conditionnelle

qu'un objet contienne m_2 sachant qu'il contient déjà m_1. On peut également fixer un seuil minimal de confiance σ_{conf}. Le support d'une règle d'association $m_1 \rightarrow m_2$ est le support de $m_1 \cup m_2$.

$$confiance(r) = \frac{support(m_1 \cup m_2)}{support(m_1)}$$

$$support(r) = support(m_1 \cup m_2)$$

L'extraction des règles d'association d'une base de données d'objets $\mathcal{B}$ consiste à déterminer l'ensemble des règles de support et confiance supérieurs aux seuils minimaux définis σ_{supp} (pour le support) et σ_{conf} (pour la confiance). Ce processus se déroule en deux étapes :

- Déterminer l'ensemble des motifs fréquents de $\mathcal{B}$.
- Générer pour chaque motif fréquent m l'ensemble des règles d'association $r : m_2 \rightarrow m_1 \setminus m_2$ avec $m_1 \supsetneq m_2$ et $confiance(r) \geqslant \sigma_{conf}$.

1.2.3 Illustration

De nombreux algorithmes existent pour extraire les règles d'association [Pasquier, 2000]. À titre d'illustration on propose les motifs fréquents extraits à partir du contexte $\mathcal{D}$ et les règles d'association générées à partir d'eux.

objets / items	A	B	C	D	E
o_1	×		×	×	
o_2		×	×		×
o_3	×	×	×		×
o_4		×			×
o_5	×	×	×		×
o_6		×	×		×

FIG. 1.3 – Le contexte $\mathcal{D}$

Le tableau 1.4 montre les motifs fréquents extraits du contexte $\mathcal{D} = (\mathcal{O}, \mathcal{I}, \mathcal{R})$ pour le seuil de support $\sigma_{supp} = \frac{1}{2}$.

On souligne également en gras les *MFF* que l'on peut extraire avec un algorithme comme CHARM [Zaki and Hsiao, 2002]. Les *MFF* sont intéressants car ils sont la description la plus complète en terme de propriétés des ensembles d'objets pour un contexte.

m	$support(m)$		m	$support(m)$
$\{A\}$	3/6		$\{B, C\}$	4/6
$\{B\}$	5/6		$\{\mathbf{B}, \mathbf{E}\}$	**5/6**
$\{\mathbf{C}\}$	**5/6**		$\{C, E\}$	4/6
$\{E\}$	5/6		$\{\mathbf{B}, \mathbf{C}\,\mathbf{E}\}$	**4/6**
$\{\mathbf{A}, \mathbf{C}\}$	**3/6**			

FIG. 1.4 – Les motifs fréquents et *MFF* (en gras) extraits de $\mathcal{D}$ pour $\sigma_{supp} = \frac{1}{2}$.

À partir de l'ensemble des motifs fréquents on génère les règles d'associations avec $\sigma_{conf} = \frac{2}{5}$.

$\{A\}$, $\{B\}$, $\{C\}$ et $\{E\}$ sont fréquents mais n'ont pas de sous-motifs, on ne génère donc pas de règles d'association à partir de ceux-ci. Pour $\{A, C\}$ on peut construire les deux règles suivantes avec $\{A\} \subseteq \{A, C\}$ et $\{C\} \subseteq \{A, C\}$:

- $r_1 : \{A\} \rightarrow \{A, C\} \setminus \{A\}$, que l'on note $A \rightarrow C$, avec $confiance(r_1) = \frac{support(\{A\} \cup \{C\})}{\{A\}} = 1 \geqslant \sigma_{conf}$
  Lorsque $confiance(r) = 1$, on dit que r est une *règle exacte*. r_1 est une règle exacte.
- $r_2 : \{C\} \rightarrow \{A, C\} \setminus \{C\}$ que l'on note $C \rightarrow A$, avec $confiance(r_2) = \frac{support(\{C\} \cup \{A\})}{\{C\}} = \frac{3}{5} \geqslant \sigma_{conf}$
  Lorsque $confiance(r) < 1$, on dit que r est une *règle approximative*. r_2 est une règle approximative.

Ainsi on génère l'ensemble des règles d'adaptation.

r	$confiance(r)$		r	$confiance(r)$
$A \rightarrow C$	1		$BC \rightarrow E$	1
$C \rightarrow A$	3/5		$E \rightarrow BC$	4/5
$B \rightarrow C$	4/5		$CE \rightarrow B$	1
$C \rightarrow B$	4/5		$B \rightarrow CE$	4/5
$B \rightarrow E$	1		$BE \rightarrow C$	4/5
$E \rightarrow B$	1		$C \rightarrow BE$	4/5

FIG. 1.5 – Les règles d'association extraites de $\mathcal{D}$ pour $\sigma_{supp} = \frac{1}{2}$ et $\sigma_{conf} = \frac{2}{5}$.

1.2.4 Bases

Un problème majeur après l'extraction des règles d'association se situe au niveau de l'interprétation et de l'exploitation des résultats. Le nombre de règles générées est souvent trés important. Dans les résultats que nous présentons pour le contexte $\mathcal{D}$ nous obtenons 12 règles pour 5 objets et 6 items avec un seuil de support $\sigma_{supp} = 3/6$, le nombre de règles générées s'élève à 50 pour le même contexte avec $\sigma_{supp} = 2/6$. Des outils de visualisation et de navigation ont été développés afin de faciliter l'analyse et la validation des résultats. Cependant réduire le nombre de règles est un problème critique compte tenu de l'importance des volumes de données manipulés en cadre réel.

Une autre dimension de ce problème est l'intérêt des règles extraites, beaucoup de règles sont redondantes et pourraient être supprimées sans nuire à la pertinence des résultats. Dans notre exemple la règle $B \rightarrow C$ est déjà contenue dans la règle $B \rightarrow CE$, toutes deux sont de confiance 4/5, on pourrait sans perte d'information éliminer la première règle qui est redondante.

Afin de résoudre ce problème de nombreuses recherches [Pasquier, 2000, Luong, 2001, G. *et al.*, 2006] ont été menées en vue de construire des *bases* pour les règles d'association. Une base permet de résumer l'ensemble des règles d'association extraites par un ensemble plus restreint de règles. Le présent travail s'inscrit dans cette perspective, notre objectif est de proposer une base pour les *règles d'adaptation* en essayant de nous inspirer des travaux existants pour les *règles d'association* et en étudiant en profondeur les spécificités de notre cadre.

Chapitre 2

Contexte applicatif

Le RàPC a un champ d'application très vaste. Comme il a été souligné au chapitre précédent, l'adaptation entretient des liens étroits avec le contexte du système que l'on souhaite concevoir et l'ACA dans sa dimension semi-automatique tend à s'appuyer sur celui-ci pour développer des méthodes d'extraction des CA. Pour ces raisons, l'étude du contexte applicatif duquel la recherche d'une base pour les règles d'adaptation puise sa motivation est un point essentiel.

Notre travail s'inscrit dans la perspective d'intégrer un module de RàPC au sein du projet KASIMIR pour la gestion des connaissances et l'aide à la décision en cancérologie[2]. Notre propos sera donc ici de contextualiser les concepts clés du RàPC dans ce cadre.

De là, CABAMAKA, le système d'ACA conçu pour le projet KASIMIR sera présenté en détail, des sources de données utilisées en passant par la fouille jusqu'à la question de la validation et de la valorisation des résultats qui est au cœur de notre problématique.

2.1 Le projet de recherche Kasimir

KASIMIR est un projet pluridisciplinaire qui réunit depuis 1997 des experts en cancérologie du centre Alexis Vautrin (centre hautement spécialisé en oncologie), le réseau de santé Oncolor, des chercheurs en psycho-ergonomie du CNAM de Paris, des chercheurs en informatique au travers de l'équipe Orpailleur du LORIA et l'association HERMES par sa mission de promotion et de diffusion des TIC[3] en santé. L'objectif du projet KASIMIR est la gestion, la diffusion et l'évolution des connaissances utiles à la pratique de la cancérologie en Lorraine.

La source première de ces connaissances est formée par les guides de bonnes pratiques, appelés *référentiels*, gérés et diffusés par le réseau Oncolor. Rédigés par des experts reconnus des nombreuses disciplines que fédère la cancérologie, les référentiels ont pour vocation d'être des protocoles de décision standards et actualisés sur lesquels les professionnels de santé peuvent s'appuyer pour guider leur pratique quotidienne (diagnostic, traitement, surveillance post-thérapeutique...). Les référentiels sont une source documentaire en langue naturelle mais qui constituent déjà un premier effort vers la formalisation des connaissances au travers de l'élaboration de terminologies et de l'utilisation d'arbres de décisions. Chaque référentiel correspond à une pathologie ou à une dimension de pathologies complexes.

Le projet KASIMIR en proposant d'accompagner la gestion et la diffusion de ces référentiels participe à l'effort d'homogénéisation et d'amélioration des pratiques en cancérologie. KASIMIR est un projet riche de nombreuses dimensions et en constante évolution, [d'Aquin, 2005] présente le projet ainsi que plusieurs de ses perspectives de développements et de recherches.

[2] On parle également d'oncologie qui est un synonyme.
[3] Technologies de l'Information et de la Communication

2.1.1 Gestion des connaissances décisionnelles en Oncologie

Au travers d'outils d'aide à la décision et sur la base des référentiels, KASIMIR propose une approche globale pour la gestion des connaissances décisionnelles en cancérologie depuis l'*application*, et éventuellement l'*adaptation*, jusqu'à l'*évolution* des connaissances. On présente ici ces différentes étapes en insistant plus particulièrement sur l'adaptation qui est la thématique centrale de la présente recherche.

Application :
L'application des référentiels dans le cadre du projet KASIMIR passe par la mise à disposition des professionnels de santé d'un système à base de connaissances pour l'aide à la décision. Accessible depuis l'Internet[4], ce système s'appuie sur la modélisation dans une logique de description[5] des référentiels « papiers » disponibles sur le site de l'association Oncolor[6]. Ainsi le système peut exploiter un moteur d'inférence pour offrir un accès intelligent aux connaissances contenues dans les référentiels. Il suffit au médecin de décrire les caractéristiques de son patient dans le cadre du référentiel considéré pour obtenir les recommandations thérapeutiques associées.

Adaptation :
Cependant les référentiels ne sont pas en mesure de proposer une recommandation thérapeutique pour tous les cas de patients (contre-indications, interaction entre plusieurs pathologies...). Ainsi, le référentiel de traitement du cancer du sein qui fut le premier à être intégré au système ne couvre que 60 à 70% des cas. Par exemple, ce référentiel ne couvre pas le cas d'un homme atteint par un cancer du sein, ce qui bien que peu fréquent est un cas avéré. C'est à ce niveau que se pose le problème de l'adaptation du référentiel.

Dans cette situation une réunion de concertation pluridisciplinaires (RCP) est demandée afin de convenir des adaptations à apporter aux recommandations thérapeutiques du référentiel pour le patient « hors référentiel ». Ces experts en s'appuyant sur leurs connaissances et leur pratique de la cancérologie vont adapter la recommandation en se fondant notamment sur le cas le plus proche existant dans le référentiel. On retrouve là le type de raisonnement que permettent de conduire les système de RÀPC.

Permettre au projet KASIMIR de réaliser de telles adaptations est tout l'enjeu de l'intégration d'un module de RÀPC au système. L'idée générale étant d'essayer d'abstraire des différents cas d'adaptation recensés les règles sur lesquelles se sont fondés les experts pour proposer une recommandation thérapeutique adaptée. Ceci permettrait au système d'aide à la décision d'être à même de proposer une solution dans le cadre de patients « hors référentiel ».

Évolution :
Les référentiels sont régulièrement actualisés du fait de l'évolution et de l'accroissement des connaissances en cancérologie, de leur confrontation avec la pratique concrète et de la mise en évidence de cas hors référentiels fréquents. KASIMIR a également pour objectif d'accompagner ce processus notamment en permettant de signaler les cas d'adaptation fréquents pour lesquels la question de l'intégration aux protocoles standards mérite d'être posée.

[4] `http ://kasimir.hermes.asso.fr/`
KASIMIR est diffusé par l'association HERMES qui assure une mission de promotion des TIC auprès des professionnels de santé.
`http ://www.hermes.asso.fr/`

[5] Les logiques de descriptions sont un fragment décidable de la logique du premier ordre. KASIMIR utilise la logique de description $\mathcal{SHOIN}(\mathcal{D})$ pour être précis [Baader *et al.*, 2003].

[6] `http ://www.oncolor.org/`

2.1.2 Kasimir et le RàPC

Dans KASIMIR un cas est un couple (pb,Sol(pb)), associant la description des caractéristiques d'un patient qui constitue le problème pb ∈ Problèmes et les recommandations thérapeutiques associées, c'est-à-dire la solution Sol(pb) ∈ Solutions du problème. Les cas pris en charge par les référentiels constituent l'ensemble des cas sources, (srce,Sol(srce)), de la base de cas de l'application.

Un prototype de module de RàPC pour KASIMIR existe [d'Aquin, 2005]. Ce module s'appuie sur le modèle des *reformulations* [Melis *et al.*, 1998]. Une reformulation est une unité élémentaire de modélisation des connaissances d'adaptation, elle est définie par un couple $(r, \mathcal{A}_r)$ avec, r une relation entre problèmes et $\mathcal{A}_r$ une fonction d'adaptation. Ainsi, une reformulation est une règle indiquant que si deux problèmes, $\texttt{pb}_1$ et $\texttt{pb}_2$ sont en relation par r, alors $\texttt{Sol}(\texttt{pb}_1)$ peut être adaptée suivant $\mathcal{A}_r$ en une solution $\texttt{Sol}(\texttt{pb}_2)$.

La remémoration dans ce cadre s'appuie sur l'adaptation, il s'agit de remémorer le ou les cas sources qui nécessitent le moins de reformulations par rapport au problème cible. On parle de chemins de similarité où chaque reformulation compte pour une étape. L'adaptation consiste alors à parcourir le chemin de similarité de srce à cible en appliquant la fonction d'adaptation $\mathcal{A}_{r_i}$ à chaque étape de reformulation.

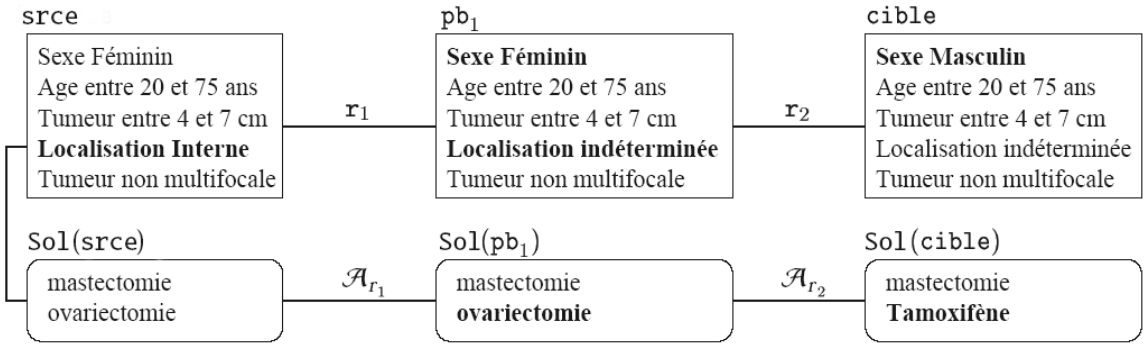

FIG. 2.1 – Exemple d'utilisation des reformulations (repris de [d'Aquin, 2005]).

La figure ci-dessus présente le principe des reformulations sur un cas volontairement simplifié dans le cadre projet KASIMIR. Le problème cible correspond à la description d'un patient non couvert par le référentiel, et ce pour deux raisons. Premièrement ce patient est de sexe masculin et le référentiel de traitement du cancer du sein est conçu pour les patients de sexes féminin qui sont la population la plus concernée par cette pathologie. Le second point est que la localisation de la tumeur est indéterminée ce qui n'est pas sans rapport avec la première raison. Les autres caractéristiques telles que l'âge du patient, la taille de la tumeur et le fait que celle-ci ne soit pas multifocale sont non problématiques.

Deux reformulations $(r_1, \mathcal{A}_{r_1})$ et $(r_2, \mathcal{A}_{r_2})$ sont appliquées pour proposer une solution au problème cible. La première dénote le fait que lorsque la localisation de la tumeur est indéterminée, on se base sur la situation la plus pessimiste c'est-à-dire celle d'une localisation interne. $\mathcal{A}_{r_1}$ consiste donc à recopier Sol(srce). La seconde reformulation indique que, comme le problème pb diffère de cible pour la valeur de la caractéristique sexe, l'adaptation consiste à remplacer l'ovariectomie, irréalisable chez un homme, par un traitement d'hormonothérapie de bénéfice thérapeutique équivalent, le tamoxifène.

L'adaptation est donc un processus central pour le RàPC dans le cadre du projet KASIMIR. De fait, la question de l'ACA est essentielle et a conduit à rechercher des méthodes permettant au système de disposer des connaissances d'adaptation nécessaires à son bon fonctionnement. C'est à cette fin qu'est développé CABAMAKA, l'application d'ACA semi-automatique utilisée pour le projet KASIMIR.

2.2 CabamakA (*Case Base Mining for Adaptation Knowledge Acquisition*)

CabamakA [d'Aquin *et al.*, 2007, Badra and Lieber, 2007] reprend les principes développés et vérifiés dans [Hanney, 1997] d'acquérir par des procédés semi-automatiques les CA à partir de la base de cas ainsi que sur la perspective de recherche d'utiliser les techniques d'ECBD à cette fin. L'idée est d'extraire les variations de caractéristiques entre les cas de la base de cas de l'application pour construire les CA. Cette méthode se présente comme moins coûteuse que la modélisation explicite de règles, les cas étant une source de connaissances plus disponible.

CabamakA est une application d'ECBD qui s'appuie les techniques de fouille de données pour acquérir des connaissances d'adaptation. Le processus d'ECBD va nous servir de trame pour présenter les différents traitements proposés depuis les données jusqu'aux résultats.

2.2.1 Sources de connaissances

Les données exploitées sont contenues dans la base de cas qui recense l'ensemble des cas pour un référentiel. C'est-à-dire l'ensemble des couples $(\texttt{srce}, \texttt{Sol}(\texttt{srce}))$ pour une certaine pathologie, avec srce décrivant les caractéristiques de patient et Sol(srce), les recommandations thérapeutiques associées. La sélection des données peut être plus fine en ne considérant qu'un aspect particulier d'une pathologie, par exemple le traitement du carcinome mammaire infiltrant pour le référentiel de traitement du cancer du sein.

Les référentiels ne sont pas des bases de données mais des bases de connaissances utilisant le formalisme des logiques de descriptions[7] (LD). Les descriptions de patients et leurs recommandations thérapeutiques associées ainsi représentées doivent donc être transformées dans un format compatible avec celui sur lequel opèrent traditionnellement les algorithmes de fouille de données, soit un contexte d'extraction $(\mathcal{O}, \mathcal{I}, \mathcal{R})$. De plus nous ne considérons pas les cas pris individuellement mais les variations de propriétés entre cas. Le formatage des données se déroule donc en deux étapes que nous allons présenter.

Formatage 1 :
Le principe[8] est de passer de descriptions de patients exprimées en LD vers des descriptions dans les termes de l'ensemble $\mathcal{P}$ de propriétés élémentaires de Problèmes et de Solutions. Un cas $(\texttt{pb}, \texttt{Sol}(\texttt{pb}))$ est donc, dans ce cadre, un ensemble $P \subseteq \mathcal{P}$ de propriétés de problèmes $p_{pb} \in \mathcal{P}_{\texttt{Problèmes}}$ et de solutions $p_{sol} \in \mathcal{P}_{\texttt{Solutions}}$.

À titre d'illustration on présente la description dans le formalisme de $2^{\mathcal{P}}$ (l'ensemble des parties de $\mathcal{P}$) d'un cas $(\texttt{pb}, \texttt{Sol}(\texttt{pb}))$, avec $\texttt{pb} \subseteq \mathcal{P}_{\texttt{Problèmes}}$ et $\texttt{Sol}(\texttt{pb}) \subseteq \mathcal{P}_{\texttt{Solutions}}$:

$$\begin{aligned}\texttt{pb} \cup \texttt{Sol}(\texttt{pb}) = \{&(\texttt{sexe} : \texttt{Feminin})_{pb}, (\texttt{age} \geqslant 20)_{pb}, (\texttt{age} < 75)_{pb}, (\texttt{taille-tumeur} \geqslant 4)_{pb},\\ &(\texttt{taille-tumeur} < 7)_{pb}, (\texttt{tumeur} : \texttt{localisation} : \texttt{LocalisationInterne})_{pb},\\ &(\texttt{tumeur} : \texttt{TumeurNonMultifocale})_{pb}, \texttt{Mastectomie}_{sol}, \texttt{Ovariectomie}_{sol}\}\end{aligned}$$

pb : Décrit un patient de sexe féminin et d'âge compris entre 20 et 74 ans. Cette patiente est atteinte d'une tumeur de taille comprise entre 4 et 7 cm (exclus). Cette tumeur est de localisation interne et de type non multifocale.

[7] CabamakA est dans le fait une application d'extraction de connaissances des bases de connaissances.

[8] Seuls les éléments indispensables à notre propos seront ici présentés, nous ne rentrons pas dans les détails de la traduction des formules en LD. On peut trouver une explication détaillée de ce formatage dans [Badra and Lieber, 2007] et [d'Aquin *et al.*, 2007].

Sol(pb) Décrit les recommandations thérapeutiques associées à pb. Il s'agit de pratiquer une mastectomie et une ovariectomie.

On annote les propriétés de problèmes par $_{pb}$ et les propriétés de solutions par $_{sol}$. Nous nous exprimerons dans le cadre de $2^{\mathcal{P}}$ pour ce qui relève des règles d'adaptation et de leur composition.

Formatage 2 :

CabamakA ne travaille pas sur les cas individuellement mais vise à extraire des variations de propriétés entre cas. Le contexte d'extraction est donc constitué de l'ensemble des couples de cas différents de la base de cas.

Un contexte d'extraction pour CabamakA est un triplet $(\mathcal{O}, \mathcal{I}, \mathcal{R})$, avec $\mathcal{O}$ l'ensemble des couples $(\texttt{cas-source}_i, \texttt{cas-source}_j)$ de la base de cas pour $i \neq j$, $\mathcal{I}$ l'ensemble des propriétés propres et partagées entre les cas et $\mathcal{R}$ une relation binaire telle que $\mathcal{R} \subseteq \mathcal{O} \times \mathcal{I}$.

$\mathcal{I}$ est construit à partir de l'ensemble $\mathcal{P}$ en considérant chaque $p \in \mathcal{P}$ selon quatre cas de figure :

- On annote par un $-$ les propriétés propres à $\texttt{cas-source}_i$;
- On annote par un $=$ les propriétés partagées $\texttt{cas-source}_i$ et $\texttt{cas-source}_j$;
- On annote par un $+$ les propriétés propres à $\texttt{cas-source}_j$;
- On ne considère pas les propriétés n'appartenant à aucun des deux cas.

Ainsi $\mathcal{I}$ contient « trois exemplaires » de $\mathcal{P}$ suivant les modalités d'appartenance des propriétés aux cas mentionnés dans les couples. Le schéma suivant explicite le résultat de ce second formatage.

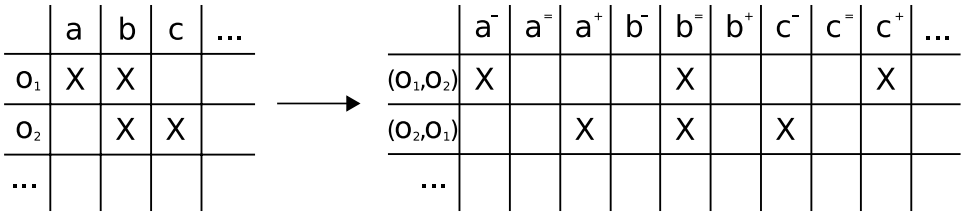

Fig. 2.2 – Construction du contexte d'extraction – Passage du formatage 1 au formatage 2.

De cette manière, on obtient un contexte binaire compatible avec les algorithmes de fouille de données et permettant d'extraire les variations de propriétés entre cas.

2.2.2 Fouille de données

CabamakA utilise l'algorithme charm [Zaki and Hsiao, 2002] pour l'étape de fouille de données. charm permet d'extraire les motifs fermés fréquents (*MFF*) du contexte. Un motif est dit *fermé* s'il n'existe pas dans le contexte de super-motif $m' \supset m$ de même support. Par exemple dans $\mathcal{D}$ (c.f. 1.2.3), {*B, C, E*} est un motif fermé de support 4/6 car il n'existe pas de motif de même support le contenant, son seul super-motif est {*A, B, C, E*} de support 2/6 (qui par ailleurs n'est pas fréquent relativement au seuil $\sigma_{supp} = 1/2$ que nous avons choisi). Cela signifie que tous les objets qui contiennent *B* et *C* contiennent également *E*, de fait le motif {*B, C*} bien qu'il soit fréquent n'est pas intéressant car il véhicule une information partielle et redondante par rapport au *MFF* {*B, C, E*}.

L'étape de fouille de données nous permet d'obtenir des généralisations sur les variations de propriétés entre cas. Généraliser est indispensable afin de trouver des règles intéressantes pouvant s'appliquer à de nombreux cas. Par ailleurs, les résultats extraits devant encore être interprétés et validés dans l'optique d'en dériver des ca, il ne serait pas envisageable de travailler directement sur

les couples de cas. Si n est la taille de la base de cas ($n = |\mathtt{BC}|$), le volume de couples de cas différents à analyser s'élève à $n(n-1)$. À titre d'illustration la partie traitement du carcinome mammaire infiltrant du référentiel de traitement du cancer du sein contient près de 647 cas, soit pour $n \simeq 650$ on a $n(n-1) \simeq 5.10^5$ couples à analyser. Cela met bien en évidence le besoin et l'avantage d'utiliser des techniques efficaces de fouille de données comme CHARM.

2.2.3 Exploitation des résultats

La dernière étape du processus est l'interprétation et la validation des résultats par l'analyste. Dans notre cadre les *MFF* sont interprétés comme des règles d'adaptation dont la cohérence et la validité doivent être évaluées par l'analyste afin de statuer sur leur intégration ou non aux CA du module de RÀPC de KASIMIR. L'ensemble des étapes vues jusqu'ici est implémenté et fonctionnel dans CABAMAKA. Cependant la tâche d'analyse des résultats reste délicate du fait de la quantité de règles extraites. À titre d'illustration on reprend les résultats d'un test réalisé avec une base de cas de 59 cas.

support	nombre de *MFF*
10%	591
5%	3057
1%	49651
0%	189690

FIG. 2.3 – Résultats de l'étape de fouille de données pour une base de cas test de 59 cas [Badra and Lieber, 2007].

Le volume des résultats à analyser devient rapidement conséquent, aussi CABAMAKA possède un système de requêtes permettant de ne travailler que sur une sous-partie des *MFF*, les motifs contenant la propriété `tumeur : localisation : LocalisationInterne` par exemple. Plusieurs recherches sont également en cours dans le cadre de la thèse de Fadi Badra afin de faciliter l'interprétation et la validation des résultats en proposant des outils de navigation parmi les règles, des métriques de qualité et de pertinence des règles, des modes de présentation plus intuitifs des *MFF* vers les règles d'adaptation.

Dans la même optique d'aide à l'interprétation et à l'évaluation des résultats, la recherche présentée ici vise à étudier la structure de l'ensemble des règles d'adaptation extraites afin de réduire, sur le modèle des bases pour les règles d'association, le nombre de règles à présenter en validation à l'analyste.

Chapitre 3

Sémantique des règles d'adaptation

Les connaissances d'adaptation (CA) exploitées dans notre cadre sont les règles d'adaptation (*RA*). CabamakA offre une solution d'acquisition semi-automatique de CA par fouille de la base de cas (BC). Les motifs fermés fréquents sont extraits de l'ensemble des couples de cas différents pouvant être constitués à partir de la base de cas. Ces motifs peuvent ensuite être lus comme des *RA* qui pourront, après validation par l'analyste, être ajoutées aux connaissances d'adaptation du système.

3.1 Connaissances d'adaptation

La question de la représentation des CA est incontournable afin d'être à même de les utiliser dans notre système et de définir les opérateurs qui les manipuleront. Nous nous appuierons sur la métaphore linguistique proposée par D. Kayser [Kayser, 1997] afin de bien définir notre cadre : « Les concepts s'expriment dans un langage, possédant une *syntaxe* et une *sémantique*. C'est cette métaphore qui, au moins depuis Leibnitz, guide les développements les plus féconds en logique, et c'est évidemment elle qui nous a dirigé – avec l'importante mise en garde de ne pas confondre langage de représentation et langage du sens. » Ainsi, après avoir présenté les principes généraux sur lesquels repose l'adaptation dans CabamakA, la syntaxe, puis la sémantique des *RA* seront définies.

3.1.1 Le schéma d'adaptation

Le RàPC propose d'exploiter les cas, c'est-à-dire les couples (`srce`,`Sol(srce)`), déjà connus du système pour résoudre de nouveaux problèmes. L'adaptation dans Kasimir s'appuie sur un schéma général mettant en relation un cas (`srce`, `Sol(srce)`) *remémoré* à partir des caractéristiques d'un nouveau problème, `cible`, pour lequel on cherche à proposer une solution, `Sol(cible)` par *adaptation*.

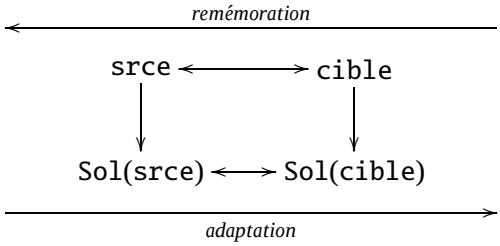

Fig. 3.1 – Le schéma d'adaptation.

Ainsi le but est de construire Sol(cible) à l'aide des CA et ce, partant du cas source (srce, Sol(srce)) et de la description du problème cible. On parle alors de *problème d'adaptation*. Un problème d'adaptation est la donnée d'un cas source (srce,Sol(srce)) et d'un problème cible dénoté par le triplet (srce, Sol(srce), cible).

3.1.2 Adaptation - Le modèle transformationnel

L'adaptation dans KASIMIR est fondée sur le modèle *transformationnel* [Carbonell, 1983], les variations entre les caractéristiques de problèmes, Δpb, induisent des variations dans les caractéristiques de leurs solutions, Δsol. Ainsi, connaissant ces variations on est en mesure de proposer une solution adaptée à de nouveaux problèmes. Les *RA* expriment le procédé, les ajouts et substitutions à opérer, pour construire de telles solutions en s'appuyant sur (Δpb, Δsol).

Le modèle d'adaptation transformationnelle :

1. (srce, cible) $\mapsto$ Δpb,
   où Δpb encode les similarités et dissimilarités entre des problèmes srce et cible.
2. (Δpb, CA) $\mapsto$ Δsol,
   où CA est un ensemble de connaissances d'adaptation et Δsol encode les similarités et dissimilarités entre Sol(srce) et la solution Sol(cible) à construire pour cible.
3. (Sol(srce), Δsol) $\mapsto$ Sol(cible),
   Sol(srce) est modifié en Sol(cible) selon Δsol.

À partir de l'ensemble des couples de cas (srce_i, $\text{Sol}(\text{srce}_i)$) et ($\text{srce}_j$, $\text{Sol}(\text{srce}_j)$), avec $i \neq j$, de la base de cas on peut dériver le couple (Δpb, Δsol) décrivant les variations de caractéristiques entre ces cas et ainsi en inférer des CA potentielles.

1. $(\text{srce}_i, \text{srce}_j) \mapsto \Delta pb_{ij}$
2. $(\text{Sol}(\text{srce}_i), \text{Sol}(\text{srce}_j)) \mapsto \Delta sol_{ij}$
3. $\left\{(\Delta pb_{ij}, \Delta sol_{ij})\right\} \mapsto$ CA

De cette manière on peut déterminer la règle d'adaptation permettant de construire la solution du problème cible, $\text{Sol}(\text{srce}_j)$ connaissant srce_i, $\text{Sol}(\text{srce}_i)$ et srce_j.

$$(\text{srce}_i, \text{Sol}(\text{srce}_i), \text{srce}_j) \mapsto \text{Sol}(\text{srce}_j)$$

Que l'on peut généraliser en une règle d'adaptation permettant de résoudre d'autres *problèmes d'adaptation* :

$$(\text{srce}, \text{Sol}(\text{srce}), \text{cible}) \mapsto \text{Sol}(\text{cible})$$

Lors d'une session de RÀPC on cherche à résoudre un problème d'adaptation en s'appuyant sur les *RA* pour inférer la solution Sol(cible). Par ailleurs, la généralisation opérée par l'extraction de motifs fermés fréquents augmente le potentiel d'application des *RA* aux problèmes d'adaptation.

3.2 Syntaxe des règles d'adaptation issues de CABAMAKA

De façon générale, une règle d'adaptation est une application qui permet de résoudre une classe de problèmes d'adaptation (srce, Sol(srce), cible). On peut donc la voir comme une application *partielle* de l'ensemble des problèmes d'adaptation dans l'ensemble des solutions. Dans cette section,

nous nous intéressons aux règles d'adaptations issues de CabamakA : étant donné un *MFF m*, on définit une règle d'adaptation *RA*(*m*).

Les variations de caractéristiques entre cas sont l'information essentielle pour l'adaptation dans notre cadre. Ces similarités et dissimilarités sont mises en évidence à l'aide d'annotations (-, =, +) lors du second formatage du contexte par CabamakA. Puis elles sont conservées tout au long du processus de fouille jusqu'à obtention des motifs fermés fréquents. Les *MFF* offrent une syntaxe en terme d'ensembles de propriétés sur $\mathcal{P}$ l'ensemble des propriétés des domaines `Problèmes` et `Solutions`.

Une règle d'adaptation est dénoté par *RA*(*m*), où *m* est un ensemble de propriétés composé de six sous-ensembles disjoints deux à deux et éventuellement vides, $P^{-}_{\mathtt{pb}}$, $P^{=}_{\mathtt{pb}}$, $P^{+}_{\mathtt{pb}}$, $P^{-}_{\mathtt{sol}}$, $P^{=}_{\mathtt{sol}}$, $P^{+}_{\mathtt{sol}}$
Ces ensembles correspondent aux propriétés propres et partagées entre les cas source et cible.

$$m = P^{-}_{\mathtt{pb}} \cup P^{=}_{\mathtt{pb}} \cup P^{+}_{\mathtt{pb}} \cup P^{-}_{\mathtt{sol}} \cup P^{=}_{\mathtt{sol}} \cup P^{+}_{\mathtt{sol}}$$

Chaque propriété de *m* peut être étiquetée ce qui permet d'identifier auquel des six ensembles elle appartient. Ainsi, si $a^{=}_{pb} \in m$, alors $a^{=}_{pb} \in P^{=}_{\mathtt{pb}}$

Notation alternative :
On propose une notation équivalente exprimant de manière synthétique les différents ensembles de propriétés composant un motif :

$$m = P^{-}_{\mathtt{pb}} \cup P^{=}_{\mathtt{pb}} \cup P^{+}_{\mathtt{pb}} \cup P^{-}_{\mathtt{sol}} \cup P^{=}_{\mathtt{sol}} \cup P^{+}_{\mathtt{sol}} = \left| \begin{array}{c|c|c} - & = & + \\ P^{-}_{\mathtt{pb}} & P^{=}_{\mathtt{pb}} & P^{+}_{\mathtt{pb}} \\ P^{-}_{\mathtt{sol}} & P^{=}_{\mathtt{sol}} & P^{+}_{\mathtt{sol}} \end{array} \right|$$

Cette représentation plus facile à lire sera très utile lorsque l'on abordera la composition de *RA*.

3.3 Sémantique des règles d'adaptation issues de CabamakA

L'ECBD opère à partir de données ayant une sémantique bien déterminée : des transactions réalisées par des clients dans un supermarché, des valeurs d'activation de gènes par exemple et, dans notre cadre des caractéristiques de patients et des recommandations thérapeutiques qui leurs sont associées. La sémantique que nous proposons ici pour nos ensembles de propriétés ne s'inscrit pas dans la perspective de construire un modèle. Bien que cet aspect reste pertinent compte tenu du fait que nos connaissances sont exprimées en premier lieu (avant formatage) en logique de descriptions, cette question dépasse le cadre du présent travail et figure plus au titre des perspectives de recherche. Notre propos est ici de définir clairement un cadre permettant à l'analyste d'interpréter les motifs sous forme de *RA*. En bout de chaîne les motifs doivent être interprétés et validés par un analyste. C'est cette étape d'interprétation qui vient donner une sémantique aux ensembles de propriétés formelles manipulées tout au long du processus.

3.3.1 Lecture et interprétation des règles d'adaptation

Une sémantique pour les *RA* est ici présentée en vue de la définition d'une opération de composition des règles d'adaptation. Elle s'appuie pour l'essentiel sur les définitions proposées jusqu'ici ([Badra and Lieber, 2007, d'Aquin *et al.*, 2007]) et a été motivée par le souci de trouver un compromis entre expressivité et possibilités d'inférence, problème classique en représentation des connaissances ([Kayser, 1997]).

Définition 3.1 :

La règle d'adaptation $RA(m)$ issue du motif $m = \left| \begin{array}{c|c|c} - & = & + \\ P^{-}_{\texttt{pb}} & P^{=}_{\texttt{pb}} & P^{+}_{\texttt{pb}} \\ P^{-}_{\texttt{sol}} & P^{=}_{\texttt{sol}} & P^{+}_{\texttt{sol}} \end{array} \right|$ permet d'effectuer le calcul suivant :

$$(\texttt{srce}, \texttt{Sol(srce)}, \texttt{cible}) \mapsto \texttt{Sol(cible)}$$

Si

(i) $P^{-}_{\texttt{pb}}$ est *inclus* dans l'ensemble des propriétés propres à `srce`

(ii) $P^{=}_{\texttt{pb}}$ est *inclus* dans l'ensemble des propriétés partagées par `srce` et `cible`

(iii) $P^{+}_{\texttt{pb}}$ est *inclus* dans l'ensemble des propriétés propres à `cible`

(iv) $P^{-}_{\texttt{sol}}$ et $P^{=}_{\texttt{sol}}$ sont *inclus* dans l'ensemble des propriétés de `Sol(srce)` et $P^{+}_{\texttt{sol}}$ n'est pas inclus dans les propriétés de `Sol(srce)`.

et

(v) $\texttt{Sol(cible)} = (\texttt{Sol(srce)} \setminus P^{-}_{\texttt{sol}}) \cup P^{+}_{\texttt{sol}}$

En s'appuyant sur la sémantique des *RA*, l'analyste est en mesure de valider, rejeter ou modifier ces règles en vue de les intégrer aux CA du système de RàPC.

3.3.2 Illustration

Soit $m_{ex} = \left\{ a^{-}_{pb}, b^{=}_{pb}, c^{=}_{pb}, d^{+}_{pb}, A^{-}_{sol}, B^{=}_{sol}, C^{+}_{sol} \right\} = \left| \begin{array}{c|c|c} - & = & + \\ \text{a} & \text{b, c} & \text{d} \\ \text{A} & \text{B} & \text{C} \end{array} \right|$

On considère le problème d'adaptation suivant :

- `srce` = $\{a, b, c, \alpha_1, \alpha_2\}$
- `Sol(srce)` = $\{A, B, \Gamma_1, \Gamma_2\}$
- `cible` = $\{b, c, d, \beta_1, \beta_2\}$
- Les $\alpha_i, \beta_i, \Gamma_i$ sont des propriétés de $\mathcal{P}$ propres au problème d'adaptation considéré et non explicitées dans $RA(m_{ex})$

$RA(m_{ex})$ permet de résoudre ce problème d'adaptation, car :

- `srce` $\supseteq \{a, b, c\}$
- `cible` $\supseteq \{b, c, d\}$
- `Sol(srce)` $\supseteq \{A, B\}$

L'adaptation par $RA(m_{ex})$ donne :
`Sol(cible)` $= (\texttt{Sol(srce)} \setminus \{A\}) \cup \{C\} = \{B, C, \Gamma_1, \Gamma_2\}$

3.4 Illustration issue de KASIMIR

Afin d'illustrer les règles d'adaptation et leur interprétation nous présentons ici un exemple concret de *RA* dans le contexte du projet KASIMIR.

3.4.1 Motif fermé fréquent issu de CABAMAKA

m_{ex} est un *MFF* extrait par CABAMAKA lors de la fouille d'une base de cas constituée à partir du référentiel de traitement du cancer du sein.

m_{ex} = { $(\texttt{taille-tumeur} < 4)^{-}_{\texttt{pb}}$, $(\texttt{age} < 70)^{=}_{\texttt{pb}}$, $(\texttt{taille-tumeur} \geqslant 4)^{+}_{\texttt{pb}}$,
$\texttt{Mastectomie partielle}^{-}_{\texttt{sol}}$, $\texttt{Curage}^{=}_{\texttt{sol}}$, $\texttt{Mastectomie}^{=}_{\texttt{sol}}$, $\texttt{Mastectomie totale}^{+}_{\texttt{sol}}$ }

Ce motif traduit les variations de propriétés de Problèmes et de Solutions entre deux descriptions de patients (srce,Sol(srce)) et (cible,Sol(cible)) que l'on peut facilement expliciter :

- srce est un patient d'âge < 70 ans, atteint d'une tumeur de taille < 4 cm. Sol(srce) : la recommandation thérapeutique associée est d'effectuer un Curage et une Mastectomie partielle

- cible est un patient d'âge < 70 ans, atteint d'une tumeur de taille ⩾ 4 cm. Sol(cible) : la recommandation thérapeutique associée est d'effectuer un Curage et une Mastectomie totale.

3.4.2 Interprétation

La règle d'adaptation $RA(m_{ex})$ est lue par l'analyste comme suit :

Si
Le patient décrit par srce est atteint d'une tumeur de taille < 4 cm.
Les patients décrits par srce et cible ont tous deux un âge < 70 ans
Le patient décrit par cible est atteint d'une tumeur de taille ⩾ 4 cm.
Un Curage et une Mastectomie partielle, qui est un type de Mastectomie,
sont les recommandations thérapeutiques proposées pour Sol(srce) et
Mastectomie totale n'est pas une recommandation de Sol(srce).
alors
Sol(cible) = (Sol(srce) \ {Mastectomie partielle}) ∪ {Mastectomie totale}
c'est-à-dire,
les recommandations thérapeutiques proposées pour Sol(cible) sont un Curage
et une Mastectomie totale (remplaçant la Mastectomie partielle de Sol(srce)).

L'analyste peut sur cette base valider, modifier ou refuser cette règle en vue de son intégration aux CA de l'application. $RA(m_{ex})$ traduit le fait que la taille de la tumeur a une incidence sur l'importance de l'acte chirurgical à pratiquer, ce qui semble intuitivement être une règle intéressante à ajouter aux CA bien que l'analyste doivent bien entendu statuer sur ce sujet.

Chapitre 4

Composition et base pour les règles d'adaptation

La sémantique des *RA* définie, il est maintenant possible d'envisager leur composition. Comme il a été mentionné plus tôt, ce travail s'inscrit dans le cadre de l'aide à l'interprétation des résultats de CabamakA. L'ensemble des modules de préparation et de fouille sont actuellement implémentés et fonctionnels[9]. Toutefois l'étape de valorisation des résultats sous forme de connaissances reste critique. Sur ce point les développements pour CabamakA s'orientent autour de deux grandes axes :

- La mise à disposition d'outils de navigation et de manipulation des *MFF* extraits afin de permettre à l'analyste d'accéder plus facilement aux règles intéressantes et de faciliter leur validation (indices de qualité, hiérarchie, classification thématique).
- L'étude de la structure algébrique de l'ensemble des *RA* afin d'éliminer les redondances et de limiter le nombre de règles à proposer en validation à l'analyste.

Ce travail s'intéresse à la deuxième problématique et cherche à définir une opération de composition des *RA* afin de construire une base pour l'ensemble des *RA*.

4.1 Composition de règles d'adaptation

Nous nous appuyons sur le principe de la composition de *RA* proposé dans [d'Aquin *et al.*, 2007] comme perspective de recherches pour l'aide à l'interprétation et à la validation des résultats de CabamakA. RA_c est la composition des règles d'adaptation RA_1 et RA_2 si adapter (`srce`,`Sol(srce)`) conformément à RA_c pour résoudre `cible` est équivalent à :

1. Mettre en évidence un *problème intermédiaire* `pb` entre `srce` et `cible`.
2. Résoudre `pb` conformément à RA_1 à partir de (`srce`,`Sol(srce)`).
3. Résoudre `cible` conformément à RA_2 à partir de (`pb`,`Sol(pb)`).

L'idée étant de construire une famille génératrice minimale *B* telle que sa clôture avec l'opération de composition comprenne l'ensemble *E* des *RA*.

4.1.1 Schéma de composition

Le principe de la composition de *RA* est illustré par le schéma de la figure 4.1.

[9] Bien entendu, de nombreuses voies de recherches restent ouvertes et continues d'être explorées sur ces étapes. On mentionnera notamment la mise en évidence de dépendances qualitatives entre variables [Badra and Lieber, 2007].

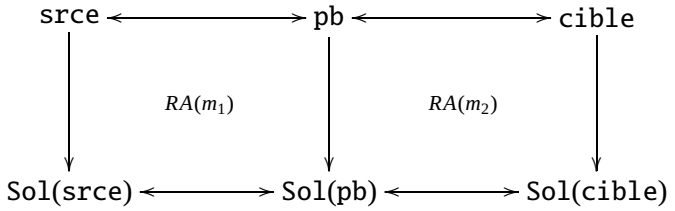

FIG. 4.1 – Principe de la composition de *RA*.

L'intuition derrière cette composition est le passage par un problème intermédiaire pb pour résoudre un problème cible en s'appuyant donc, non plus sur une, mais sur deux adaptations. Le résultat de la composition étant la *RA* permettant de résoudre directement cible à partir de (srce,Sol(srce)). Ainsi, il n'est plus nécessaire de spécifier explicitement une règle d'adaptation dès lors que l'on peut la reconstruire par une ou plusieurs compositions de règles d'adaptations. La question étant quelle définition donner à cette composition et à quelles conditions peut-elle être appliquée, notamment au niveau des propriétés du problème intermédiaire.

4.1.2 Définition de la composition de règles d'adaptation

Définition 4.1 :
L'opérateur de *composition* des *RA* est noté « ; ».

$$RA_c = RA_1 \ ; \ RA_2$$

Si pour deux cas (srce,Sol(srce)) et (cible,Sol(cible)) on a l'équivalence entre les deux affirmations suivantes :

1. RA_c permet d'adapter (srce,Sol(srce)) en la solution Sol(cible) de cible.
2. Il existe un problème pb tel que :
   - RA_1 permet d'adapter (srce,Sol(srce)) en *une* solution Sol(pb) de pb.
   - RA_2 permet d'adapter (pb,Sol(pb)) en *la* solution Sol(cible) de cible.

Dans la suite, nous allons nous intéresser à la composition de règles d'adaptation issues de *MFF*— $RA_1 = RA(m_1)$, $RA_2 = RA(m_2)$ — dont le résultat peut aussi s'écrire sous la forme d'un motif — $RA_c = RA(m_c)$. La proposition suivante donne une condition *suffisante* pour qu'on est $RA(m_1) \ ; \ RA(m_2) = RA(m_c)$.

Proposition 1 :

$$\textbf{Soit} \quad m_1 = \left| \begin{array}{c|c|c} - & = & + \\ P^-_{\text{pb}} & P^=_{\text{pb}} & P^+_{\text{pb}} \\ P^-_{\text{sol}} & P^=_{\text{sol}} & P^+_{\text{sol}} \end{array} \right| \quad \textbf{et } m_2 = \left| \begin{array}{c|c|c} - & = & + \\ Q^-_{\text{pb}} & Q^=_{\text{pb}} & Q^+_{\text{pb}} \\ Q^-_{\text{sol}} & Q^=_{\text{sol}} & Q^+_{\text{sol}} \end{array} \right|$$

Une condition *suffisante* pour que $RA(m_1) \ ; \ RA(m_2)$ puisse s'écrire sous la forme de $RA(m_c)$ est d'avoir $P^=_{\text{sol}} \cup P^+_{\text{sol}} = Q^-_{\text{sol}} \cup Q^=_{\text{sol}}$.
Dans ce cas, on a :

$$m_c = \left| \begin{array}{c|c|c} - & = & + \\ (P^-_{\text{pb}} \cup P^=_{\text{pb}}) \setminus (Q^=_{\text{pb}} \cup Q^+_{\text{pb}}) & (P^-_{\text{pb}} \cup P^=_{\text{pb}}) \cap (Q^=_{\text{pb}} \cup Q^+_{\text{pb}}) & (Q^=_{\text{pb}} \cup Q^+_{\text{pb}}) \setminus (P^-_{\text{pb}} \cup P^=_{\text{pb}}) \\ (P^-_{\text{sol}} \cup P^=_{\text{sol}}) \setminus (Q^=_{\text{sol}} \cup Q^+_{\text{sol}}) & (P^-_{\text{sol}} \cup P^=_{\text{sol}}) \cap (Q^=_{\text{sol}} \cup Q^+_{\text{sol}}) & (Q^=_{\text{sol}} \cup Q^+_{\text{sol}}) \setminus (P^-_{\text{sol}} \cup P^=_{\text{sol}}) \end{array} \right| \tag{4.1}$$

Remarque :

En fait, les différents sous-ensembles qui composent m_c sont construits de façon analogue aux ensembles de propriétés issues des couples de cas lors de l'étape de formatage du contexte. En d'autres termes :

- La « colonne − » de m_c correspond à l'ensemble des propriétés de problèmes ($_{\text{pb}}$) (respectivement de solutions ($_{\text{sol}}$)) qui sont contenues dans la partie `srce` de m_1 et qui ne sont pas contenues dans la partie `cible` de m_2.
- La « colonne = » de m_c correspond à l'ensemble des propriétés de problèmes ($_{\text{pb}}$) (respectivement de solutions ($_{\text{sol}}$)) qui sont contenues dans la partie `srce` de m_1 et dans la partie `cible` de m_2.
- La « colonne + » de m_c correspond à l'ensemble des propriétés de problèmes ($_{\text{pb}}$) (respectivement de solutions ($_{\text{sol}}$)) qui sont contenues dans la partie `cible` de m_2 et qui ne sont pas contenues dans la partie `srce` de m_1.

Preuve :

On considère m_1 et m_2 tels que définis dans la proposition 1.2 et on suppose :

$$P^=_{\text{sol}} \cup P^+_{\text{sol}} = Q^-_{\text{sol}} \cup Q^=_{\text{sol}} \tag{4.2}$$

Soit m_c définis par (1). Montrons que $RA(m_1)$; $RA(m_2) = RA(m_c)$.
Soit (`srce`, `Sol(srce)`, `cible`), un problème d'adaptation.

Supposons que $RA(m_1)$; $RA(m_2)$ permette de résoudre ce problème d'adaptation et donne comme résultat `Sol(cible)`).

Dans ce cas :

- `srce` $\supseteq P^-_{\text{pb}} \cup P^=_{\text{pb}}$ donc s'écrit `srce` $= P^-_{\text{pb}} \cup P^=_{\text{pb}} \cup \alpha$, où α est disjoint de $P^-_{\text{pb}} \cup P^=_{\text{pb}}$;
- `cible` $\supseteq Q^=_{\text{pb}} \cup Q^+_{\text{pb}}$ donc s'écrit `cible` $= Q^-_{\text{pb}} \cup Q^=_{\text{pb}} \cup \beta$, où β est disjoint de $Q^=_{\text{pb}} \cup Q^+_{\text{pb}}$;
- `Sol(srce)` $\supseteq P^-_{\text{sol}} \cup P^=_{\text{sol}}$ donc s'écrit `Sol(srce)` $= P^-_{\text{sol}} \cup P^=_{\text{sol}} \cup \Gamma$, où Γ est disjoint de $P^-_{\text{sol}} \cup P^=_{\text{sol}}$.

Soit `pb` $= P^=_{\text{pb}} \cup P^+_{\text{pb}} \cup Q^-_{\text{pb}} \cup Q^=_{\text{pb}}$.
$RA(m_1)$ permet de résoudre le problème d'adaptation (`srce`, `Sol(srce)`, `pb`)
en une solution `Sol(pb)`) = (`Sol(srce)` $\setminus P^-_{\text{sol}}) \cup P^+_{\text{sol}}$.
On a `Sol(pb)` $\supseteq P^=_{\text{sol}} \cup P^+_{\text{sol}}$: il contient $P^=_{\text{sol}}$ car `Sol(srce)` $\supseteq P^=_{\text{sol}}$ et $P^-_{\text{sol}} \cap P^=_{\text{sol}} = \emptyset$ (c.f. section 3.2).
D'après (4.2) on a donc : `Sol(pb)` $\supseteq Q^-_{\text{sol}} \cup Q^=_{\text{sol}}$
Comme `pb` $\supseteq Q^-_{\text{pb}} \cup Q^=_{\text{pb}}$ et `cible` $\supseteq Q^=_{\text{pb}} \cup Q^+_{\text{pb}}$, $RA(m_2)$ permet de résoudre le problème d'adaptation (`pb`, `Sol(pb)`, `cible`) en une solution `Sol(cible)`.

$$\begin{aligned} \texttt{Sol(cible)} &= (\texttt{Sol(pb)} \setminus Q^-_{\text{sol}}) \cup Q^+_{\text{sol}} \\ &= (((\texttt{Sol(srce)} \setminus P^-_{\text{sol}}) \cup P^+_{\text{sol}}) \setminus Q^-_{\text{sol}}) \cup Q^+_{\text{sol}} \\ &= ((((P^-_{\text{sol}} \cup P^=_{\text{sol}} \cup \Gamma) \setminus P^-_{\text{sol}}) \cup P^+_{\text{sol}}) \setminus Q^-_{\text{sol}}) \cup Q^+_{\text{sol}} \end{aligned}$$

or, Γ, P^-_{sol}, $P^=_{\text{sol}}$ sont disjoints deux à deux, d'où $(P^-_{\text{sol}} \cup P^=_{\text{sol}} \cup \Gamma) \setminus P^-_{\text{sol}} = P^=_{\text{sol}} \cup \Gamma$
donc `Sol(cible)` $= ((P^=_{\text{sol}} \cup P^+_{\text{sol}} \cup \Gamma) \setminus Q^-_{\text{sol}}) \cup Q^+_{\text{sol}}$
D'après (4.2), on a donc :
`Sol(cible)` $= \Gamma \cup Q^=_{\text{sol}} \cup Q^+_{\text{sol}}$

Nous allons montrer à présent que $RA(m_c)$ permet de résoudre (src, Sol(src), cible) en une solution Sol'(cible) et que Sol'(cible) = Sol(cible).

On a bien :

- $\texttt{src} \supseteq \left[(P^-_{pb} \cup P^=_{pb}) \setminus (Q^=_{pb} \cup Q^+_{pb})\right] \cup \left[(P^-_{pb} \cup P^=_{pb}) \cap (Q^=_{pb} \cup Q^+_{pb})\right]$ car $\texttt{src} = P^-_{pb} \cup P^=_{pb} \cup \alpha$, où α est disjoint de $P^-_{pb} \cup P^=_{pb}$;
- $\texttt{cible} \supseteq \left[(Q^=_{pb} \cup Q^+_{pb}) \setminus (P^-_{pb} \cup P^=_{pb})\right] \cup \left[(P^-_{pb} \cup P^=_{pb}) \cap (Q^=_{pb} \cup Q^+_{pb})\right]$ car $\texttt{src} = Q^-_{pb} \cup Q^=_{pb} \cup \beta$, où α est disjoint de $Q^=_{pb} \cup Q^+_{pb}$;
- $\texttt{Sol(src)} \supseteq \left[(P^-_{sol} \cup P^=_{sol}) \setminus (Q^=_{sol} \cup Q^+_{sol})\right] \cup \left[(P^-_{sol} \cup P^=_{sol}) \cap (Q^=_{sol} \cup Q^+_{sol})\right]$ car $\texttt{src} = P^-_{sol} \cup P^=_{sol} \cup \Gamma$, où Γ est disjoint de $P^-_{sol} \cup P^=_{sol}$.

D'où :

$$\texttt{Sol'(cible)} = \left\{\texttt{Sol(src)} \setminus \left[P^-_{pb} \cup P^=_{pb}) \setminus (Q^=_{pb} \cup Q^+_{pb})\right]\right\} \cup \left[(Q^=_{pb} \cup Q^+_{pb}) \setminus (P^-_{pb} \cup P^=_{pb})\right]$$

$$\texttt{Sol'(cible)} = \left\{(P^-_{sol} \cup P^=_{sol} \cup \Gamma) \setminus \left[P^-_{pb} \cup P^=_{pb}) \setminus (Q^=_{pb} \cup Q^+_{pb})\right]\right\} \cup \left[(Q^=_{pb} \cup Q^+_{pb}) \setminus (P^-_{pb} \cup P^=_{pb})\right]$$

Or, pour trois ensembles A, B et C on a :

$$A \setminus (B \setminus C) = A \cap (\overline{B \setminus C}) = A \cap (\overline{B} \cap C) = (A \cap \overline{B}) \cup (A \cap C) = (A \setminus B) \cup (A \cap C)$$

Donc,

$$\texttt{Sol'(cible)} = \left[(P^-_{sol} \cup P^=_{sol} \cup \Gamma) \setminus (P^-_{pb} \cup P^=_{pb})\right] \cup \left[(P^-_{sol} \cup P^=_{sol} \cup \Gamma) \cap (Q^=_{pb} \cup Q^+_{pb})\right] \cup \left[(Qpb^= \cup Q^+_{pb}) \setminus (P^-_{pb} \cup P^=_{pb})\right]$$

$$\texttt{Sol'(cible)} = \Gamma \cup \left\{\left[(Q^=_{pb} \cup Q^+_{pb}) \cap (P^-_{sol} \cup P^=_{sol} \cup \Gamma)\right] \cup \left[(Qpb^= \cup Q^+_{pb}) \setminus (P^-_{pb} \cup P^=_{pb})\right]\right\}$$

$$\texttt{Sol'(cible)} = \Gamma \cup Q^=_{pb} \cup Q^+_{pb}$$

$$\texttt{Sol'(cible)} = \texttt{Sol(cible)}$$

□

Donc, on a montré que si on pouvait résoudre (src, Sol(src), cible) par $RA(m_1)$; $RA(m_2)$, on peut le résoudre par $RA(m_c)$ et les deux adaptations donnent le même résultat.

La réciproque se montre de façon analogue.
On a donc $RA(m_1)$; $RA(m_2) = RA(m_c)$

4.1.3 Composition faible de règles d'adaptation

La composition de règles d'adaptation définie en 4.1 propose un cadre trop large pour pouvoir être exploitée. La proposition 1 introduit une condition suffisante pour la composition de *RA* sur laquelle il est possible de s'appuyer pour définir un cadre plus restreint nous permettant de travailler sur la composition de *RA*. De plus, il a été prouvé dans 4.1.2 que sous cette condition $RA(m_1)$; $RA(m_2) = RA(m_c)$. L'idée est donc de proposer une composition faible de règles d'adaptation notée « $\diamond$ » en intégrant cette condition suffisante 4.2 à ce nouveau cadre.

Définition 4.2 :
Suivant la définition 4.1
On considère m_1 et m_2 tels que définis dans la proposition 1

Comme il a été prouvé en 4.1.2 :
Pour, $RA(m_1)$; $RA(m_2) = RA(m_c)$, il *suffit* que 4.2, c'est-à-dire $P^=_{sol} \cup P^+_{sol} = Q^-_{sol} \cup Q^=_{sol}$.
On définit $\diamond$ pour :
Si $RA(m_1) \diamond RA(m_2) = RA(m_c)$ **où** m_c est défini par l'équation 4.1.

4.2 Propriétés de la composition faible

La composition faible de *RA* définie, nous disposons d'un cadre suffisamment clair pour travailler sur la composition de règles d'adaptation. Il est intéressant d'étudier ses différentes propriétés. Dans cette section nous allons étudier si la loi interne $\diamond$ sur l'ensemble des *RA* vérifie ou non différentes propriétés.

Remarque : Pour tout ce qui suit, on pose $m_i = P^{-}_{\mathtt{pb}_i} \cup P^{=}_{\mathtt{pb}_i} \cup P^{+}_{\mathtt{pb}_i} \cup P^{-}_{\mathtt{sol}i} \cup P^{=}_{\mathtt{sol}i} \cup P^{+}_{\mathtt{sol}i}$ conformément à 3.2

Afin de simplifier les notations et la lecture, on se donne les correspondances suivantes pour m_i : $a_i = P^{-}_{\mathtt{pb}_i}$ $b_i = P^{=}_{\mathtt{pb}_i}$ $c_i = P^{+}_{\mathtt{pb}_i}$ $A_i = P^{-}_{\mathtt{sol}i}$ $B_i = P^{=}_{\mathtt{sol}i}$ $C_i = P^{+}_{\mathtt{sol}i}$

Ainsi, $m_i = \left|\begin{array}{c|c|c} - & = & + \\ a_i & b_i & c_i \\ A_i & B_i & C_i \end{array}\right| = \left|\begin{array}{c|c|c} - & = & + \\ P^{-}_{\mathtt{pb}_i} & P^{=}_{\mathtt{pb}_i} & P^{+}_{\mathtt{pb}_i} \\ P^{-}_{\mathtt{sol}i} & P^{=}_{\mathtt{sol}i} & P^{+}_{\mathtt{sol}i} \end{array}\right|$

4.2.1 Associativité

Définition 4.3 :
Une loi de composition sur un ensemble E est associative si,

$$\forall x, y, z \in E, \quad x * (y * z) = (x * y) * z$$

On écrit alors $x * y * z$.

On veut prouver pour tout m_1, m_2 et m_3 appartenant à un ensemble de *MFF* extraits par CabamakA que l'on a : $(RA(m_1) \diamond RA(m_2)) \diamond RA(m_3) = RA(m_1) \diamond (RA(m_2) \diamond RA(m_3))$

Si $RA(m_i) \diamond RA(m_j) = RA(m_{ij})$ **alors** d'après 4.1 on a,

$$m_{ij} = \left|\begin{array}{c|c|c} - & = & + \\ a_{ij} & b_{ij} & c_{ij} \\ A_{ij} & B_{ij} & C_{ij} \end{array}\right| = \left|\begin{array}{c|c|c} - & = & + \\ (a_i \cup b_i) \setminus (b_j \cup c_j) & (a_i \cup b_i) \cap (b_j \cup c_j) & (b_j \cup c_j) \setminus (a_i \cup b_i) \\ (A_i \cup B_i) \setminus (B_j \cup C_j) & (A_i \cup B_i) \cap (B_j \cup C_j) & (B_j \cup C_j) \setminus (A_i \cup B_i) \end{array}\right|$$

1er cas de figure :
On suppose la condition suffisante 4.2 respectée, on a $B_1 \cup C_1 = A_2 \cup B_2$ et $B_2 \cup C_2 = A_3 \cup B_3$.
On montre que,

$$\begin{array}{ll} A_{12} = (A_1 \cup B_1) \setminus (B_2 \cup C_2) & A_{23} = (A_2 \cup B_2) \setminus (B_3 \cup C_3) \\ B_{12} = (A_1 \cup B_1) \cap (B_2 \cup C_2) & B_{23} = (A_2 \cup B_2) \cap (B_3 \cup C_3) \\ C_{12} = (B_2 \cup C_2) \setminus (A_1 \cup B_1) & C_{23} = (B_3 \cup C_3) \setminus (A_2 \cup B_2) \end{array}$$

On rappelle : $(A \setminus B) \cup (A \cap B) = A$

La condition de la composition faible est satisfaite pour $RA(m_1) \diamond RA(m_{23})$ car :
$A_{23} \cup B_{23} = (A_2 \cup B_2) \setminus (B_3 \cup C_3) \cup (A_2 \cup B_2) \cap (B_3 \cup C_3) = A_2 \cup B_2 = B_1 \cup C_1$
La condition de la composition faible est satisfaite pour $RA(m_{12}) \diamond RA(m_3)$ car :
$B_{12} \cup C_{12} = C_{12} \cup B_{12} = (B_2 \cup C_2) \setminus (A_1 \cup B_1) \cup (A_1 \cup B_1) \cap (B_2 \cup C_2) = B_2 \cup C_2 = A_3 \cup B_3$.

Ainsi,

$A_{(12)3} = \underbrace{(A_{12} \cup B_{12})} \setminus (B_3 \cup C_3)$
$= (A_1 \cup B_1) \setminus (B_3 \cup C_3)$
$= A_{1(23)}$

$A_{1(23)} = (A_1 \cup B_1) \setminus \underbrace{(B_{23} \cup C_{23})}$
$= (A_1 \cup B_1) \setminus (B_3 \cup C_3)$
$= A_{(12)3}$

et

$B_{(12)3} = \underbrace{(A_{12} \cup B_{12})} \cap (B_3 \cup C_3)$
$= (A_1 \cup B_1) \cap (B_3 \cup C_3)$
$= B_{1(23)}$

$B_{1(23)} = (A_1 \cup B_1) \cap \underbrace{(B_{23} \cup C_{23})}$
$= (A_1 \cup B_1) \cap (B_3 \cup C_3)$
$= B_{(12)3}$

et

$C_{(12)3} = (B_3 \cup C_3) \setminus \underbrace{(A_{12} \cup B_{12})}$
$= (B_3 \cup C_3) \setminus (A_1 \cup B_1)$
$= C_{1(23)}$

$C_{1(23)} = \underbrace{(B_{23} \cup C_{23})} \setminus (A_1 \cup B_1)$
$= (B_3 \cup C_3) \setminus (A_1 \cup B_1)$
$= C_{(12)3}$

2e cas de figure :

On suppose la condition suffisante 4.2 respectée pour $RA(m_2) \diamond RA(m_3)$ avec $B_2 \cup C_2 = A_3 \cup B_3$, mais pas pour $RA(m_1) \diamond RA(m_2)$ et on a $B_1 \cup C_1 \neq A_2 \cup B_2$.

On montre que,

$A_{12} =$ **échec de la composition faible** $\quad A_{23} = (A_2 \cup B_2) \setminus (B_3 \cup C_3)$
$B_{12} =$ **échec de la composition faible** $\quad B_{23} = (A_2 \cup B_2) \cap (B_3 \cup C_3)$
$C_{12} =$ **échec de la composition faible** $\quad C_{23} = (B_3 \cup C_3) \setminus (A_2 \cup B_2)$

L' échec de la composition faible se propage pour $RA(m_{12}) \diamond RA(m_3)$

$A_{(12)3} = B_{(12)3} = C_{(12)3} =$ **échec de la composition faible**

La condition de la composition faible n'est pas satisfaite pour $RA(m_1) \diamond RA(m_{23})$, car $(B_1 \cup C_1) \neq (A_2 \cup B_2)$ et on a, $(A_{23} \cup B_{23}) = (A_2 \cup B_2) \setminus (B_3 \cup C_3) \cup (A_2 \cup B_2) \cap (B_3 \cup C_3) = (A_2 \cup B_2)$

donc $A_{1(23)} = B_{1(23)} = C_{1(23)} =$ **échec de la composition faible**

3e cas de figure :

On suppose la condition suffisante 4.2 respectée pour $RA(m_1) \diamond RA(m_2)$ avec $B_1 \cup C_1 = A_2 \cup B_2$, mais pas pour $RA(m_2) \diamond RA(m_3)$ et on a $B_2 \cup C_2 \neq A_3 \cup B_3$.

On montre que,

$A_{12} = (A_1 \cup B_1) \setminus (B_2 \cup C_2)$ $\quad A_{23} =$ **échec de la composition faible**
$B_{12} = (A_1 \cup B_1) \cap (B_2 \cup C_2)$ $\quad B_{23} =$ **échec de la composition faible**
$C_{12} = (B_2 \cup C_2) \setminus (A_1 \cup B_1)$ $\quad C_{23} =$ **échec de la composition faible**

L' échec de la composition faible se propage pour $RA(m_1) \diamond RA(m_{23})$

$A_{1(23)} = B_{1(23)} = C_{1(23)} =$ **échec de la composition faible**

La condition de la composition faible n'est pas satisfaite pour $RA(m_{12}) \diamond RA(m_3)$, car $(B_2 \cup C_2) \neq (A_3 \cup B_3)$ et on a, $(B_{12} \cup C_{12}) = (B_2 \cup C_2) \setminus (A_1 \cup B_1) \cup (A_1 \cup B_1) \cap (B_2 \cup C_2) = (B_2 \cup C_2)$

donc $A_{(12)3} = B_{(12)3} = C_{(12)3} =$ **échec de la composition faible**

Selon des méthodes analogues on montre les égalités suivantes :
$a_{(12)3} = a_{1(23)}$, $b_{(12)3} = b_{1(23)}$, $c_{(12)3} = c_{1(23)}$
Ainsi, $\theta_{(12)3} = \theta_{1(23)}$ **pour** $\theta \in \{a, b, c, A, B, C\}$
Donc, la composition faible de RA, $\diamond$ est associative.

4.2.2 Non commutativité de $\diamond$

Définition 4.4 :
Une loi de composition sur un ensemble E est commutative si :

$$\forall x, y \in E, x * y = y * x.$$

$\diamond$ n'est pas commutative. Le contre-exemple suivant le prouve :

Soit, $m_1 = \left|\begin{array}{c|c|c} - & = & + \\ \emptyset & \emptyset & \emptyset \\ \emptyset & \{p\} & \emptyset \end{array}\right| \diamond \left|\begin{array}{c|c|c} - & = & + \\ \emptyset & \emptyset & \emptyset \\ \{p\} & \emptyset & \emptyset \end{array}\right|$ où p est une propriété quelconque.

$RA(m_1) \diamond RA(m_2) \neq$ **échec de la composition faible** car $\{p\} \cup \emptyset = \emptyset \cup \{p\}$
En revanche, $RA(m_1) \diamond RA(m_2) \neq$ **échec de la composition faible** : $\emptyset \cup \emptyset \neq \emptyset \cup \{p\}$

4.2.3 Pas d'élément neutre pour $\diamond$

Définition 4.5 :
Soit E un ensemble muni d'une loi de composition. On dit que $e \in E$ est un élément neutre si

$$\forall x \in E, \quad e * x = x * e = x$$

Il n'existe pas d'élément neutre pour $\diamond$. Nous allons le montrer par l'absurde.

Soit $e = \left|\begin{array}{c|c|c} - & = & + \\ a_e & b_e & c_e \\ A_e & B_e & C_e \end{array}\right|$ tel que RA(e) est élément neutre de $\diamond$.

Cela entraîne que pour tout *MFF* m,$RA(e) \diamond RA(m) = RA(m)$, ce qui est absurde :

il suffit en effet de choisir $m = \left|\begin{array}{c|c|c} - & = & + \\ a & b & c \\ A & B & C \end{array}\right|$ tel que $B_e \cup C_e \neq A \cup B$

pour que cette égalité ne soit pas respectée. On a alors $RA(e) \diamond RA(m) =$ **échec de la composition faible**

4.2.4 Condition pour que $RA(m) \diamond RA(m) \neq$ échec de la composition faible

Soit $m = \left|\begin{array}{c|c|c} - & = & + \\ a & b & c \\ A & B & C \end{array}\right|$

Pour que $RA(m) \diamond RA(m) \neq$ **échec de la composition faible**, il faut (4.2) $B \cup C = A \cup B$
Comme A, B et C sont disjoints deux à deux (3.2), $A = C = \emptyset$

4.2.5 Règles inverses

En l'absence d'un élément neutre il est délicat de parler d'éléments inverses, cependant notre cadre présente une propriété intéressante.

Soit $m = \left|\begin{array}{c|c|c} - & = & + \\ a & b & c \\ A & B & C \end{array}\right|$. On note $-m$ le *MFF* $-m = \left|\begin{array}{c|c|c} - & = & + \\ c & b & a \\ C & B & A \end{array}\right|$

(on échange les colonnes « + » et « – ».)

Il est important de noter que si m est un *MFF* extrait de CabamakA alors $-m$ sera également un *MFF* extrait de CabamakA. En effet, CabamakA considère tous les couples de cas différents appartenant à la base de cas, de fait pour tous couples ($\texttt{cas-source}_i$, $\texttt{cas-source}_j$) construits il existe un couple ($\texttt{cas-source}_j$, $\texttt{cas-source}_i$) dans le contexte de fouille. Cette propriété est conservée pour les *MFF* extraient lors de la fouille.

Notons aussi que $-(-m) = m$.
$RA(m) \diamond RA(-m)$ est défini car $B \cup C = C \cup B$. De plus,

$$RA(m) \diamond RA(-m) = RA(\left|\begin{array}{c|c|c} - & = & + \\ \emptyset & a \cup b & \emptyset \\ \emptyset & A \cup B & \emptyset \end{array}\right|)$$

Par ailleurs $RA(m) \diamond RA(-m) \diamond RA(m)$ est défini, et $RA(m) \diamond RA(-m) \diamond RA(m) = RA(m)$.

Remarque :
Ceci nous rapproche de la façon dont l'inverse est définie pour les semi-groupes inversifs qui ne disposent justement pas d'élément neutre mais pour lesquels tout élément à un élément inverse unique, au sens où $x * y * x = x$ et $y * x * y = y$.

4.3 Famille génératrice – Base

La composition faible de *RA* étant définie, on peut s'attacher à la construction de la base B s'appuyant sur la famille génératrice qui clot E_{MFF}, l'ensemble des *RA* dérivées des *MFF* issus de la fouille de données.

Définition 4.6 :
On appelle clôture pour $\diamond$ d'un ensemble E_{MFF} de *RA*, le plus petit ensemble de *RA* dénoté par $Clôture_\diamond(E_{MFF})$ et défini par :
- $E_{MFF} \subseteq Clôture_\diamond(E_{MFF})$
- Si $RA(m_1)$, $RA(m_2) \in Clôture_\diamond(E_{MFF})$ et que $RA(m_1) \diamond RA(m_2) \neq$ **échec de la composition faible**, donc $RA(m_1)$; $RA(m_2) \in Clôture_\diamond(RA)$

Définition 4.7 :
G est une *famille génératrice* pour E_{MFF} si $Clôture_\diamond(G) \supseteq E_{MFF}$.
Par exemple E_{MFF} est une famille génératrice finie pour E_{MFF}.

Définition 4.8 :
B est une base pour l'ensemble E_{MFF} des *RA* dérivées des *MFF* extraits par CabamakA si *B* est une famille génératrice de cardinal minimum pour E_{MFF}.
Notons qu'une base est nécessairement un ensemble fini, puisque E_{MFF} est une famille génératrice finie.

4.3.1 Algorithme de construction de *G*

On propose ici un algorithme permettant de construire une famille génératrice *G* à partir de E_{MFF} l'ensemble des *RA* dérivées des *MFF* extraits par CabamakA.

$F := E$
$G := \emptyset$ La famille génératrice que l'on cherche à construire.
$C := \emptyset$ La clôture de *G*.

tant que $C \not\supseteq E$ **faire**
Soit $RA' \in F$
$F := F \setminus RA$
$C := C \cup \{RA\} \cup \{RA \diamond RA' \mid RA' \in C\} \cup \{RA' \diamond RA \mid RA' \in C\}$
$G := G \cup \{RA\}$
fin tant que
résultat : *G*

Les principes sur lesquels s'appuie cet algorithme sont généraux et pourrait être utilisés dans d'autres cadres de composition que la composition faible $\diamond$ que nous avons définis dans cette recherche.

4.3.2 Résultats préliminaires

Une première implémentation de l'algorithme permettant de construire notre base a été réalisée en Python. Celle-ci nécessiterait encore de nombreuses optimisations pour être pleinement exploitable, notamment en terme de temps d'exécution. Toutefois, une première validation a été réalisée à partir des *MFF* extraits par CabamakA, avec un seuil de support $\sigma_{supp} = 2\%$, pour la partie traitements complémentaires suite à une chirurgie non conservatrice du référentiel de traitement du cancer du sein. La base a été construite pour le sous-ensemble de *RA* exprimant une relation entre la localisation d'une tumeur et un traitement par radiothérapie, soit 2605 *MFF* au total.

Sur ce sous-ensemble l'algorithme a permis d'obtenir une famille génératrice de 1932 *RA* desquels on peut reconstruire l'ensemble initial par l'opération de composition des *RA*. L'ensemble des *MFF* sortie de fouille a donc été réduit de plus de 25%. Ces résultats sont relativement encourageant et confirme que notre composition permet bien de réduire l'ensemble des règles extraites. Ils le sont d'autant plus que les *RA* sont passées en revue aléatoirement, aussi, il est possible que nous ayons été amené à éliminer des *RA* permettant des compositions plus intéressantes. La mise en évidence de critères d'ordonnancement permettant d'optimiser notre algorithme constitue une perspective de recherche prometteuse pour améliorer notre base.

Conclusions et perspectives

La présente recherche a permis d'apporter des éléments de réponses à la problématique de l'aide à l'interprétation et à la validation des règles d'adaptation en explorant une perspective de recherche proposée dans [d'Aquin *et al.*, 2007]. Après avoir présenté le contexte scientifique de notre travail que constituent le RàPC, l'ECBD et l'ACA, nous avons resitué l'ensemble des notions clés sur lesquels nous nous appuyons au sein du projet KASIMIR et de son système acquisition semi-automatique de connaissances d'adaptation CABAMAKA. De là, une étude de la sémantique des règles d'adaptation a été proposée dans l'optique d'offrir un cadre clair permettant de définir la composition de règles d'adaptation. Cette composition représentant un cadre trop large trop large pour être exploité dans la présente recherche, nous avons été amené à définir la composition faible de règles d'adaptation et à étudier ses différentes propriétés. À l'appui de ces éléments nous avons été en mesure de proposer une base pour les règles d'adaptation et de valider cette approche à la lumière des résultats obtenus sur des données réelles extraites par CABAMAKA à partir du référentiel de traitement du cancer du sein.

Bien sûr de nombreuses questions restent ouvertes pour le développement de notre base et de nombreuses perspectives de recherche restent à explorer. La plus immédiate vise à améliorer l'algorithme permettant de construire notre base en travaillant notamment à la découverte de critères permettant d'ordonner l'ensemble des règles d'adaptation afin d'exploiter pleinement les possibilités offertes par notre composition faible des règles d'adaptation. Travailler à optimiser notre implémentation sera une étape importante afin de pouvoir mener à bien les tests et observations nécessaires à une validation complète de notre approche permettant de souligner ses bénéfices et ses limites.
Une caractérisation mathématique plus fine que celle que nous avons esquissée pour l'ensemble des règles d'adaptation muni de l'opération de composition faible sera essentielle afin d'exploiter tout le potentiel de l'approche. Ce travail de recherche conséquent permettra certainement d'apporter des éléments en vue de participer à la recherche d'indices de qualité pour les règles d'adaptation espérant y trouver un bénéfice équivalent à celui qui en est fait dans le cadre des bases pour les règles d'associations. Par ailleurs il sera important de chercher si d'autres compositions intéressantes, plus ou moins restreintes que celle définie dans cette recherche, peuvent être proposées pour les règles d'adaptation. Une étude plus poussée de la structure algébrique des règles d'adaptation permettra de développer des bases pour les règles d'adaptation dans des perspectives proches de celles proposées pour les règles d'association [Pasquier, 2000, Guigues and Duquenne, 1986].

Bibliographie

[Baader *et al.*, 2003] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider, editors. *The Description Logic Handbook*. Cambridge University Press, cambridge, UK, 2003.

[Badra and Lieber, 2007] F. Badra and J. Lieber. Extraction de connaissances d'adaptation par l'analyse de la base de cas. In *Extraction et gestion des connaissances (EGC'2007), Actes des septièmes journées Extraction et Gestion des Connaissances, Namur, Belgique, 23-26 janvier 2007, 2 Volumes*, Revue des Nouvelles Technologies de l'Information, pages 751–760, 2007.

[Carbonell, 1983] J. G. Carbonell. Learning by analogy : Formulating and generalizing plans from past experience. In R. S. Michalski and J. G. Carbonell and T. M. Mitchell, editor, *Machine Learning, An Artificial Intelligence Approach*, Chapitre 5, pages 137–161. Morgan Kaufmann, Inc., 1983.

[Chouraqui, 1986] E. Chouraqui. Le raisonnement analogique : sa problématique, ses applications. In *Actes des Journées Nationales sur l'Intelligence Artificielle, Aix-les-Bains*, pages 107–117. CEPADUES-Editions, Toulouse, 1986.

[d'Aquin, 2005] M. d'Aquin. *Un portail sémantique pour la gestion des connaissances en cancérologie*. Thèse d'université, Université Henri Poincaré Nancy 1, soutenue le 15 décembre 2005, 2005.

[d'Aquin *et al.*, 2007] M. d'Aquin, F. Badra, S. Lafrogne, J. Lieber, A. Napoli, and L. Szathmary. Case Base Mining for Adaptation Knowledge Acquisition. In M. M. Veloso, editor, *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI'07)*, pages 750–755. Morgan Kaufmann, Inc., 2007.

[Fayyad *et al.*, 1996] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From data mining to knowledge discovery in databases. *Ai Magazine*, 17 :37–54, 1996.

[G. *et al.*, 2006] Gasmi G., Ben Yahia S., and Mephu Nguifo E.and Slimani Y.. IGB : une nouvelle base générique informative des règles d'association. *Revue I3 (Information-Interaction-Intelligence)*, 6(1) :pp 31–67, 2006.

[Guigues and Duquenne, 1986] J.L. Guigues and V. Duquenne. Familles minimales d'implications informatives résultant d'un tableau de données binaires. *Mathématiques et Sciences Humaines*, 95 :5–18, 1986.

[Hanney, 1997] K. Hanney. Learning Adaptation Rules from Cases. Master's thesis, Trinity College, Dublin, 1997.

[Kayser, 1997] D. Kayser. *La représentation des connaissances*. Hermès, 1997.

[Lieber, 2006] J. Lieber. A Definition and a Formalization of Conservative Adaptation for Knowledge-Intensive Case-Based Reasoning – Application to Decision Support in Oncology (A Preliminary Report). Rapport de recherche, LORIA, 2006.

[Lieber *et al.*, 2004] J. Lieber, M. d'Aquin, S. Brachais, and A. Napoli. Une étude comparative de quelques travaux sur l'acquisition des connaissances d'adaptation pour le raisonnement à partir de cas. In R. Kanawati, S. Salotti, and F. Zehraoui, editors, *Actes du douzième atelier raisonnement à partir de cas, RàPC'04*, pages 53–60, 2004.

[Luong, 2001] V. Phan LUONG. Raisonnement sur les règles d'association. In CÉPADUÈS, editor, *17eme Journées Bases de Données Avancées BDA'2001*, pages 299–310, 2001.

[Melis *et al.*, 1998] E. MELIS, J. LIEBER, and A. NAPOLI. Reformulation in Case-Based Reasoning. In B. SMYTH and P. CUNNINGHAM, editors, *Fourth European Workshop on Case-Based Reasoning, EWCBR-98*, Lecture Notes in Artificial Intelligence 1488, pages 172–183. Springer, 1998.

[Pasquier, 2000] N. PASQUIER. Data Mining : Algorithmes d'extraction et de réduction des règles d'association dans les bases de données. Thèse de Doctorat d'État, Université Clermont-Ferrand II, 2000.

[Riesbeck and Schank, 1989] C. K. RIESBECK and R. C. SCHANK. *Inside Case-Based Reasoning*. Lawrence Erlbaum Associates, Inc., Hillsdale, New Jersey, 1989.

[Zaki and Hsiao, 2002] M. J. ZAKI and C.-J. HSIAO. CHARM : An Efficient Algorithm for Closed Itemset Mining. In *SIAM International Conference on Data Mining SDM'02*, pages 33–43, Apr 2002.

Résumé

L'acquisition de connaissances d'adaptation (ACA) , notamment dans sa dimension automatique, ouvre de grandes perspectives pour l'étape critique en raisonnement à partir de cas (RàPC) que constitue l'adaptation. CabamakA s'appuie sur les méthodes symboliques de fouille de données pour extraire des généralisations sur les variations de propriétés entre cas.

Cette recherche propose d'étudier la structure de l'ensemble des règles d'adaptation dans l'optique de construire une base. Notre base est un ensemble minimal qui clos l'ensemble des règles extraites sous l'opération de composition.

Réduire le nombre de règles d'adaptation à soumettre en validation à l'analyste est stratégique afin de réduire le coût de développement des systèmes de RàPC. L'expertise humaine est essentielle pour permettre l'exploitation de ces connaissances d'adaptation dans le cadre projet Kasimir pour la gestion des connaissances décisionnelles en oncologie.

Ce travail présente la sémantique des règles d'adaptation et leur composition en vue de construire une base. Plusieurs perspectives pour l'implémentation et l'amélioration de cette base sont proposées.

Mots-clés: Règles d'adaptation, Base, Acquisition de Connaissances d'Adaptation (ACA), Extraction de Connaissances de Bases de Données (ECBD), Raisonnement à Partir de Cas (RàPC)

Abstract

Automatic Adaptation Knowledge Acquisition (AKA) offers great perspectives for improving the critical step of adaptation on Case Based Reasoning (CBR) systems. CabamakA uses symbolic data mining methods to extract generalisation about properties variations between cases.

This work proposes to study the structure of adaptation rules in order to construct a base. Our base is a minimal set which closes the complete set of extracted rules under the composition operation.

Reducing the number of adaptation rules an human analyst needs to assess is strategic to reduce development costs of CBR systems. Expert assessment is essential to allow for exploitation of these adaptation knowledge in our knowledge based decision support system in oncology : Kasimir.

This master's thesis presents the semantic definition of adaptation rules and their composition for base building. Some perspectives for implementation and improvement of this base are outlined.

Keywords: Adaptation rules, Base, Adaptation Knowledge Acquisition (AKA), Knowledge Discovery in Databases (KDD), Case-Based Reasoning (CBR)