Aspects théoriques et pratiques de l'apprentissage en cas d'incomplétude des données : Etendre l'apprentissage par satisfiabilité

Dominique BOUTHINON[1,2] Henry SOLDANO[1,2] Veronique VENTOS[3] (*)
[1] Atelier de BioInformatique (ABI)
[2] Laboratoire d'Informatique Paris Nord (LIPN)
[3]Laboratoire de Recherche en 'Informatique Paris SUD (LRI)

Atelier de BioInformatique 12, rue Cuvier 75005 PARIS (FRANCE)
email : Dominique.Bouthinon@snv.jussieu.fr
phone : (33) 01 44 24 65 82
fax : (33) 01 44 27 63 12

Abstract
This work originated from a concept learning problem in Molecular Biology in which the data displays an extreme form of incompleteness : we search for a substructure common to a set of RNA sequences, whereas the structure of each RNA sequence is unknown. here incompleteness arises because each description only contains indirect information from which assumptions can be derived about the object. The main purpose of this paper is to discuss theoretical issues concerning incomplete data. We first investigate how concept learning by satisfiability has to be modified in order to cope with incompleteness in negative examples. We propose then a framework that generalizes leaning from satisfiability in order to cope both with generalized examples (each representing a set of examples) and incomplete examples (each representing what we know about the example, i.e. the corresponding set of possible complete examples). Finally we show that this framework, though designed to cope with incomplete data, has again to be extended when dealing with single and multiple predicate learning. The reason is that in predicate learning a piece of data is an observation that is related to several requirements concerning positive and negative instances of the target predicates. So when the observation is incomplete, these requirements can no longer be considered independently: in order to determine what hypotheses satisfy our requirements concerning the instances, we have to consider each possible complete observation. Finally we briefly discuss various practical issues concerning our RNA problem, and particularly the necessity of considering only maximal sets of assumptions, thus performing a form of default reasoning.

Keywords :
Concept-learning, inductive logic programming, abduction, incomplete data, molecular biology

1 Introduction

Un problème de Biologie Moléculaire est à l'origine de ce travail : il s'agissait de trouver une sous-structure secondaire commune à un ensemble de séquences d'ARN [1,2,3]. Une caractéristique importante de ce problème est que seule une représentation ambigue de chaque exemple est accessible. Plus précisément à chaque séquence correspond un ensemble de structures possibles, l'une d'entre elles correspondant au vrai repliement. Pour chaque exemple nous avons donc un forme extrême de connaissance incomplète : aucun élément structural n'est "certain" mais nous avons des contraintes (certains éléments structuraux sont incompatibles) et des connaissances générales. Dans [4, 5] nous rendions compte de cette application en présentant un cadre de travail (les problèmes de Classe-A) permettant de traiter ce genre de situation. Ce cadre utilisait la theta-subsomption pour tester une complétude faible : une hypothèse couvre faiblement un exemple, représenté par un état de connaissances incomplètes, si elle couvre une réalité compatible avec cet état de connaissances. Un ensemble de contraintes et une théorie du domaine étaient utilisés pour définir ces réalités compatibles avec ce que nous

(*) The authors have equally contributed to this work and appears here according to the alphabetical order.

savons de l'exemple. Nous ne discutions pas dans ce travail de la manière dont des contre-exemples incomplets du concept-cible peuvent être traités, ni de l'utilisation de connaissances dans le test de complétude faible. Un des buts du présent travail est de tenir compte de ces deux points, mais pour cela nous allons étudier en détail les conséquences de l'incomplétude des données sur la manière dont on pose les problèmes d'apprentissage.
Le paragraphe 2 est une discussion de la notion d'apprentissage de concepts en situation d'incomplétude. Nous discutons séparément les cadres issus de l'apprentissage de concepts au sens de l'apprentissage par interprétations, et les cadres issus de l'apprentissage de prédicats couramment pratiqué en programmation logique inductive. Nous montrons que les extensions nécessaires pour tenir compte de l'incomplétude dans les premiers cadres ne sont pas suffisantes dans les seconds. Le paragraphe 3 discute brièvement certaines particularités rencontrées lors de l'application au RNA.

1 Apprentissage de concepts à partir d'exemples.

Une vision classique de l'apprentissage d'un concept à partir d'exemples considère un exemple (ou exemple positif) d'un concept comme une situation particulière, associée à une description en principe complète, pour laquelle le concept est satisfait. Une telle description, émise dans un certain langage, doit donc être "couverte" par un énoncé, émis dans un autre langage, que l'on associera à ce concept. Les cadres d'apprentissage fondés sur la logique sont particulièrement intéressants puisqu'ils fournissent à la fois une syntaxe (langage formel) pour la représentation des connaissances et une sémantique permettant de raisonner sur ces dernières. Dans ces cadres les exemples sont des interprétations, un exemple positif e+ est un modèle d'un énoncé, c'est à dire une formule, candidat à représenter le concept. De même un contre-exemple ne doit pas être un modèle d'un tel énoncé. Cette vision est explicitement ou implicitement celle de toutes les méthodes d'apprentissage de concepts utilisant des représentations proches de la logique propositionnelle. Lorsque les situations à décrire font intervenir des relations entre constituants, l'usage de restrictions de la logique des prédicats s'impose. Cependant, pour différentes raisons, et en schématisant un peu, deux voies ont été explorées dans ce domaine. La première, plus récente, étend la conception présentée ci-dessus à cette représentation plus riche et a donné lieu à ce que nous appellerons l'apprentissage par interprétations. La seconde, issue de la programmation logique et qui a été très étudiée dans le cadre de la programmation logique inductive (ou ILP), associe aux concepts-cibles des prédicats. Nous lui donnerons ici le nom générique d'apprentissage de prédicats [11]. En apprentissage de prédicats on dispose d'une théorie du domaine, sous la forme d'une théorie clausale, et d'exemples (positifs et négatifs) constitués d'instances (faits) des prédicats associés aux concepts-cibles. Le but est d'enrichir cette théorie avec un ensemble de clauses (l'hypothèse) afin que chaque exemple positif soit une conséquence logique de cette nouvelle théorie et que chaque exemple négatif ne le soit pas.
Un article récent et très éclairant du à L. Deraedt [12] décrit les liens entre les différents cadres issus de ces deux formes d'apprentissage de concepts. En particulier Deraedt propose un cadre d'apprentissage, l'apprentissage par satisfiabilité, auquel les autres cadres, apprentissage de concepts ou apprentissage de prédicats, se "réduisent". Par "réduction" DeRaedt entend une transformation grâce à laquelle une méthode permettant de résoudre un problème dans un cadre A, permet mécaniquement de résoudre un problème dans le cadre B. Ce dernier est dit alors se "réduire" au cadre A.
Une des motivations de DeRaedt était de suggérer que ce cadre de satisfiabilité, auquel les autres cadres se réduisent sans que la converse soit vraie, était particulièrement intéressant en présence de données incomplètes. Nous montrerons ici qu'en toute généralité le traitement de données incomplètes nécessite une extension du cadre de satisfiabilité. Nous commençons ci-dessous par traiter de l'apprentissage de concepts au sens du paradigme d'apprentissage par interprétations.

2.1 Apprentissage de concepts à partir d'interprétations

Comme DeRaedt nous chercherons à caractériser l'ensemble des solutions, c'est à dire l'ensemble des formules h du langage de représentation des concepts, satisfaisant les conditions requises, c'est à dire la couverture des exemples positifs et le rejet des exemples négatifs. Il faut remarquer à ce propos que nous adoptons ici le point de vue de l'Espace des Versions de T. Mitchell, étendu par la suite par H. Hirsh [9]: l'ensemble des solutions courantes est l'ensemble des formules du langage qui sont "consistantes" avec les informations disponibles à un instant donné. Plus précisément, c'est l'ensemble des formules qui n'ayant pas été éliminées à cet instant, sont encore candidates pour représenter le concept. Cette manière de considérer

comme appartenant à l'espace des solutions les formules non encore rejetées, est centrale dans la prise en compte des exemples incomplets.

2.1.1 Exemples incomplets : cadre d'incomplétude de Hirsh

Lorsque la description des exemples (positifs ou négatifs) est incomplète, il n'est pas toujours possible de savoir si un exemple est couvert ou non par une formule candidate. Dans ce cas, et suivant le principe énoncé par H. Hirsh, nous considérons qu'une formule est solution si elle est consistante avec les exemples positifs et négatifs, c'est à dire précisement s'il n'est pas exclu que la formule candidate couvre les exemples positfs complets et ne couvre pas les exemples négatifs complets. Autrement dit une formule candidate est solution si elle ne peut être éliminée.

Dans notre présentation un exemple positif ou négatif incomplet est représenté, comme suggéré par DeRaedt [12], par une théorie clausale dont les modèles sont les exemples "possibles" complets. En conséquence de quoi nous aboutissons à la formulation suivante:

Un élément h du langage est consistant avec un exemple positif incomplet e+, si il existe un modèle de e+ qui soit aussi un modèle de h, ce que nous écrirons:

h et $e^+ |\neq \square$ (1)

en utilisant l'opérateur de conséquence logique |=.
Notons que la relation (1) est équivalente à la notion d'explication faible de Flach[8].

De même h est consistant avec un exemple négatif incomplet e-, si il existe un modèle de e- qui ne soit pas un modèle de h, ce que nous écrirons:

$e^- |\neq h$ (2)

Exemple[1]:
Le langage est constitué des prédicats "oiseau(), poisson(), même-Espèce(,), leger(), dense(), rouge(), bleu() " et des constantes "a,b".

e+={oiseau(a); oiseau(b); <--poisson(a); <--poisson(b); leger(a) <-- leger(b) ; leger(b) <-- leger(a) ; <--dense(a); <--dense(b); rouge(a), <--rouge(b); <--bleu(a); bleu(b); même-Espèce(a,b)}

e+ a deux modèles: dans l'un a et b sont tous deux légers, dans l'autre ni l'un ni l'autre ne l'est.

h = leger(x)<--oiseau(x) est consistante avec e+ car l'un des modèles de e+ (celui où a et b sont tous deux légers) est un modèle de h.
h= <--leger(x) et oiseau(x) est consistante avec e+ car l'un des modèles de e+ (celui où ni a ni b ne sont légers) est un modèle de h.

Nous référons par la suite à (1) et (2) comme étant le cadre d'incomplétude de Hirsh.

[1] Dans le documents les clauses unitaires positives instanciées seront simplement représentés par l'atome instancié correspondant. Les clauses unitaires négatives seront représentées par l'atome instancié correspondant précédé d'une flèche

2.1.2 Exemples généralisés.

Une situation fréquente en apprentissage est celle où la description d'un exemple est volontairement incomplète. Dans ce cas il s'agit d'une manière de définir un exemple "généralisé" c'est à dire qu'une partie de la description de l'exemple est inutile au sens suivant : tous les exemples "virtuels" qui complètent l'exemple généralisé exercent la même contrainte sur les solutions. L'exemple généralisé représente donc un ensemble d'exemples tous positifs ou tous négatifs.
Comme dans le cas des exemples incomplets, un exemple généralisé est représenté par une théorie clausale, dont les modèles sont les exemples "virtuels" complets. Nous aboutissons alors à la formulation suivante:

Un élément h du langage est consistant avec un exemple positif généralisé e+, si tous les modèles de e+ sont aussi des modèles de h:

$\mathbf{e}^+ \models h$ (3)

De même h est consistant avec un exemple négatif généralisé e-, si aucun des modèles de e- n'est modèle de h.
h et $e^- \models \square$ (4)
On note que la relation (4) correspond, à une transformation près, à la notion d'explication forte de Flach [8] ($h \wedge B \models e$, B théorie du domaine)[2].

Exemple:
Nous reprenons l'exemple e+ précédent en supposant cette fois qu'il s'agit d'un exemple positif généralisé. Cette fois:

h1 = leger(x)<--oiseau(x), comme h2= <--leger(x) $\wedge$ oiseau(x), ne sont pas consistantes avec e+ (au sens de (3)) car dans les deux cas seul l'un des modèles de e+ est un modèle de la formule candidate.

On constatera qu'aussi bien pour les exemples incomplets que pour les exemples généralisés, la relation de consistance des exemples positifs et négatifs ne peut être ramenée à une seule relation de couverture et à sa négation. La plupart des cadres d'apprentissage étant articulés autour de la relation de couverture et sa négation ne peuvent rendre compte de manière satisfaisante des exemples incomplets et généralisés. Une exception importante est le travail de Hirsh sur la fusion des espaces de versions, qui d'une manière plus informelle, contient déjà la conception présentée ci-dessus. Une autre exception se trouve dans le travail de Kakas [7, 10] sur l'intégration de mécanismes abductifs et inductifs. Celui-ci traite plus précisément de l'incomplétude dans I-ACL [10] et utilise alors une relation particulière de couverture et traite des exemples négatifs par l'usage de prédicats particuliers not_P. Nous discuterons I-ACL en §2.2.2.

Pour être plus précis le cadre d'apprentissage par satisfiabilité proposé par DeRaedt utilise la relation (1) pour définir la couverture d'un exemple incomplet. En conséquence les exemples négatifs ne devant pas être couverts, c'est la relation (4) qu'utilise alors Deraedt, celle permettant de traiter les exemples négatifs généralisés. Remarquons que le propos central de DeRaedt, la construction de théorie, passe essentiellement par l'apprentissage à partir d'exemples positifs seulement.

Bien que formulées d'une manière lègèrement différentes les relations (1) et (4) sont aussi celles qui fondent l'apprentissage multi-instances tel qu'il a été défini par T. Dietterich [6], puis étudié par différents auteurs dont Zucker [17]. Plus précisément Dietterich considère un exemple composé de plusieurs instances. Il ne s'agit pas à proprement parlé d'incertitude ici : l'exemple correspond à plusieurs réalités différentes mais qui ne s'excluent pas. A l'origine de ce travail un problème de chimie : l'activité d'une molécule. Cette molécule peut prendre plusieurs conformations (structures physiques

[2] Considérons par exemple les données suivantes : h = {vole(X) $\leftarrow$ oiseau(X)}, B = {oiseau(a)} , e = {vole(a)}, qui vérifient $h \wedge B \models e$, relation équivalente à (4) par la transformation $e^- = B \wedge \neg e$.

tri-dimensionnelles) et on cherche des caractéristiques structurales associées à l'activité chimique. La relation (1) dans ce cas exprime qu'il suffit qu'une des ces conformations permette l'activité pour que celle-ci soit observée. Inversement la relation (4) exprime qu'aucune des conformations d'une molécule inactive ne doit permettre l'activité. Remarquons que dans le cas de Dietterich, les représentations ne sont pas de type relationnel. De plus chaque conformation est explicitement décrite indépendamment des autres ce qui est à l'opposé de la représentation de l'exemple incomplet par une théorie clausale, qui est une représentation implicite des possibles à partir d'un état, partiel, de connaissances.
Enfin Hirsh, qui a inspiré la présentation de ce paragraphe, considère plusieurs situations. La situation principale décrite par Hirsh est le cas des exemples incomplets pour lesquels il décrit, informellement, l'usage des relations (1) et (2).

Nous présentons ci-dessous deux relations, une pour les exemples positifs et l'autre pour les exemples négatifs, qui permettent d'exprimer aussi bien les cas d'incomplétude que les exemples généralisés, et permettent également d'exprimer ces deux situations simultanément. Nous présentons dabord les relations générales, puis nous montrons comment elles permettent d'exprimer les cas particuliers.

2.1.3 Exemples incomplets et généralisés

L'idée essentielle est de rendre compte de l'incertitude par de multiples "possibles", chaque possible étant écrit sous la forme d'une théorie clausale.
Ainsi un exemple positif e+, n'est plus représenté par une théorie clausale mais par un ensemble de théories clausales {p1+, p2+, ... , pm+} chaque pi+ correspondant à un possible. Un exemple positif est alors couvert par une formule h ssi il existe pi+ tel que tous les modèles de pi+ sont des modèles de h :

$\exists pi+ \in e+ \, , \, pi^+ \models h$ (1-3)

De même un exemple négatif e-, représenté par un ensemble de possibles {p1-, p2-, ... , pn-}, n'est pas couverte par une formule h ssi il existe pi- tel qu'aucun modèle de pi- n'est un modèle de h.

$\exists pi- \in e-, h \wedge pi^- \models \square$ (2-4)

On peut noter que la relation (1-3) est une situation qui reprend la contrainte faible (incertitude) de la relation (1) dans le sens où il y a une simple condition d'existence sur pi+, et la contrainte forte (généralisation) de la relation (3) du fait que pi+ est un exemple positif généralisé.
De même la relation (2-4) est une situation qui reprend la contrainte faible de la relation (2) et la contrainte forte de la relation (4).
Par ailleurs on suppose implicitement que chaque possible est consistant (possède au moins un modèle).

Ce cadre, que nous appellerons *apprentissage par possibilité*, subsume l'apprentissage par satisfiabilité[3]. En effet dans ce dernier, un exemple positif e^+_s est représenté par une théorie clausale . Il en va de même pour tout exemple négatif e^-_s. Les propriétés à satisfaire pour toute hypothèse solution h, également représentée par une théorie calusale, sont alors :

(S1) $\forall e^+_s \in E^+, h \wedge e^+_s \not\models \square$ (h et e^+_s possède un modèle commun)
(S2) $\forall e^-_s \in E^-, h \wedge e^-_s \models \square$ (h et e^-_s n'ont aucun modèle commun)

Ce système s'exprime aisément en apprentissage par possibilité. A chaque exemple positif e^+_s du cadre de satisfiabilité nous faisons correspondre un exemple $e^+ = \{p+_1, p+_2, ..., p+_i, ..., p+_k\}$ décrivant l'ensemble des modèles de $e+_s$, chaque $p+_i$ étant plus

[3] et par transitivité l'apprentissage par entailment et par interprétation.

précisémment la théorie clausale asociée à un modèle donné de $e+_s$. Ainsi la relation (1-3) exprime bien qu'un modèle de e_s est aussi modèle de h.
D'une façon dissymétrique à chaque exemple négatif e^-_s du cadre de satisfiabilité nous faisons correspondre un exemple $e^- = \{e^-_s\}$. La relation (2-4) signifiant bien qu'aucun modèle de e^-_s n'est modèle de h.

L'apprentissage par possibilité subsume aussi le cadre d'incomplétude de Hirsh (conditions (1) et (2) de la section 2.1.1). La démonstration vient d'être faite pour la relation (1) de Hirsh qui est identique à (S1) du cadre de satisfiabilité. Pour la relation (2) stipulant qu'un modèle d'un exemple négatif e^- du cadre d'incomplétude de Hirsh ne doit pas être un modèle de h, il suffit de considérer $e- = \{p^-_1,p^-_2,...,p^-_i,...,p^-_k\}$, où chaque pi- décrit la théorie clausale associée à un modèle donné de e^-.

Nous montrons maintenant comment l'apprentissage par possibilité permet d'exprimer différentes situations d'exemples incomplets et d'exemple généralisés.

a) Exemples positifs ou négatifs provenant de l'apprentissage par Explication (Hirsh)

Hirsh considère un exemple positif ou négatif complet et une théorie permettant de conclure sur le concept-cible ou sa négation. Une explication est alors constituée d'une interprétation partielle dont l'exemple complet est une extension, et qui est suffisante pour conclure. Chaque explication associée à un exemple correspond donc à un exemple généralisé. Les relations précédentes nous permettent d'exprimer cette situation en considérant que l'exemple, après recherche d'explications, est constitué de l'ensemble de ses explications possibles. Pour retrouver les relations (1-3) et (2-4) il suffit de transformer pour chaque explication l' interprétation partielle correspondante en une théorie clausale.
Dans ce cas la relation (1-3) exprime que pour au moins une explication positive pi+ toutes les extensions de l'interprétation partielle correspondante doivent être des modèles de l'hypothèse h. De même la relation (1-4) exprime qu'au moins une explication négative pi- n'a aucun modèle commun avec h.

Pour illustrer cette situation considérons l'exemple positif complet représenté par l'interprétation suivante:

{oiseau(a), oiseau(b), leger(a), leger(b), rouge(a), bleu(b), même-Espèce(a,b)}

Supposons que le concept-cible soit " cage-amicale " et qu'une théorie T permette de conclure sur le concept cible à partir de l'ensemble de faits suivant:
{oiseau(a), oiseau(b), même-Espèce(a,b)}
que nous considérons ici comme une interprétation partielle (qui en l'occurence ne spécifie que des littéraux positifs} et nous lui associons la théorie clausale suivante:
p1+ ={oiseau(a) ; oiseau(b) ; même-Espèce(a,b)}

Par ailleurs la même théorie permet de conclure sur " cage-amicale "à partir de :
{oiseau(a), oiseau(b), leger(a), leger(b)}
à qui nous associons la théorie clausale :
p2+ ={oiseau(a) ; oiseau(b) ; leger(a) ; leger(b)}

La relation (1-3) assure que
h1= Il existe X,Y oiseau(X) & oiseau(Y) & leger(X)
est une solution car tous les modèles de p2+ sont des modèles de h1 : c'est ici la première explication qui supporte l'hypothèse h1.

b) Incomplétude et données manquantes non pertinentes

Il s'agit essentiellement d'une situation où un exemple généralisé (positif ou négatif) est partiellement défini (incomplet), sachant qu'on dispose d'un ensemble de connaissances supplémentaires qui pourront être utilisées afin de le compléter. Il faut noter que ces connaissances complémentaires sont données a priori et représentent le seul sous-ensemble du langage de description des exemples qui soit jugé pertinent pour effectuer la complétion précitée.

Plus formellement, soit HB la base de Herbrand du langage, l'exemple généralisé partiel est représenté par une théorie clausale eg et la connaissance complémentaire par un ensemble d'atomes instanciés A qui est un sous-ensemble de HB représentant les atomes dont la valeur de vérité est considérée comme pertinente relativement au concept-cible. L'idée est de construire, pour chaque exemple eg, une nouvelle théorie clausale eg $\wedge \overline{A}$ représentant un exemple généralisé complet (par rapport à A) possible. $\overline{A}$ est la théorie clausale associée à une interprétation partielle de A[4]. Il est aisé de voir qu'il y a autant d'exemples généralisés complets possibles que d'interprétations partielles sur la sous-base de A.

Ainsi nous pouvons définir qu'une hypothèse h couvre l'exemple généralisé partiel eg (ici positif) si tous les modèles d'un des exemples généralisés complets possibles sont des modèles de h :
(5) h couvre eg $\Leftrightarrow \exists \overline{A}$ tel que eg $\wedge \overline{A}$ |= h et eg $\wedge \overline{A}$ |≠ ❐

Pour le cas où eg est un exemple négatif la relation devient :
(6) h ne couvre pas eg $\Leftrightarrow \exists \overline{A}$ tel que eg $\wedge \overline{A} \wedge$ h |= ❐ et eg $\wedge \overline{A}$ |≠ ❐
On retrouve les relations (1-3) et (2-4) de l'apprentissage par possibilité si l'on considère que (eg $\wedge \overline{A}$) est un "possible" pi (de ce fait implicitement eg $\wedge \overline{A}$ |≠ ❐), et que l'exemple généralisé partiel eg (positif ou negatif) est en réalité représenté par l'ensemble des exemples généralisés complets possibles e = {p1, p2, ...,pi,...,pk}.

Si nous reprenons l'exemple positif précédent :
eg={oiseau(a)←; oiseau(b)← ; ←poisson(a); ←poisson(b); leger(a) <-- leger(b) ; leger(b) ← leger(a) ; ←dense(a); ←dense(b); rouge(a)←, ←rouge(b); bleu(a)←; bleu(b)←; même-Espèce(a,b)←}
L'ensemble des faits permettant de compléter un exemple est
A={oiseau(a); oiseau(b); poisson(a), poisson(b), leger(a); leger(b); dense(a); dense(b), même-Espèce(a,b) }
(les atomes correspondant à des prédicats de couleur n'interviennent pas).
Ainsi h= leger(x)<--oiseau(x) est solution, car il existe $\overline{A}$ = {oiseau(a) ←, oiseau(b) ←, ←poisson(a), ←poisson(b), leger(a)←, leger(b)← , ←dense(a), ←dense(b), meme-espece(a,b) ←} forçant la valeur de vérité des atomes de A tel que :
eg $\wedge \overline{A}$ |= h et eg $\wedge \overline{A}$ |≠ ❐.
Remarquons que dans $\overline{A}$ les seules réelles suppositions sont leger(a) et leger(b), les autres littéraux apparaissant déja dans eg.

Le cadre des exemples généralisés incomplets subsume aussi le cadre de l'apprentissage par satisfiabilité[5] et le cadre d'incomplétude de Hirsh[6].

---

[4] une interprétation partielle formée à partir de la sous-base de Herbrand A détermine les valeurs de vérité de chaque atome de A, et ne statue pas sur les atomes de la base de Herbrand qui n'apparaissent pas dans A. Elle est donc partielle sur HB et totale sur A.

[5] Il suffit de considérer que pour chaque exemple eg+ de la relation (5) A =HB. De ce fait tout $\overline{A}$ vérifiant la condition implicite eg+ $\wedge \overline{A}$ |≠ ❐, est un modèle de eg+, la condtion (5) devient équivalente à la condition (S1) de la satisfiabilité. De même pour chaque exemple négatif eg- de la relation (6) on associe A = ∅, $\overline{A}$ est alors la théorie clausale universelle {}, la relation (6) devient équivalente à la relation (S2).

[6] La démonstration pour la relation (1) du cadre d'incomplétude de Hirsh est la même que pour la relation (S1) de la satisfiabilité. Pour la relation (2) de Hirsh, on montre qu'il suffit d'associer A = HB à tout exemple négatif eg- de la relation (6), $\overline{A}$, qui vérifie la condition implicite, eg- $\wedge \overline{A}$ |≠ ❐, est la théorie clausale associé à un modèle de eg-, et la condition (6) devient équivalente à la condition (2).

2.2 Apprentissage de prédicats.

Une grande partie de la programmation logique inductive s'intéresse à l'apprentissage de prédicats, où le problème est de trouver une hypothèse permettant de déduire de ce que l'on sait, les instanciations d'un ou de plusieurs prédicats-cibles. Pour traiter de la notion d'incomplétude dans ce cadre il est nécessaire de comprendre quels types de problèmes d'apprentissage sont abordés par l'apprentissage de prédicats.

Nous commençons par donner sur un exemple trois écritures différentes équivalentes, chacune portant une conception différente d'un problème d'apprentissage :

a) $h \wedge oiseau(a) \wedge \leftarrow vole(a) \models \square$

b) $h \wedge oiseau(a) \models vole(a)$

c) $h \models vole(a) \leftarrow oiseau(a)$

Les trois écritures étant équivalentes, (a), (b) et (c) partagent les mêmes solutions, par exemple:
$h = vole(X) \leftarrow oiseau(X)$

a) relève de l'apprentissage par satisfiabilité, c'est à dire de l'apprentissage de concepts suivant le paradigme de l'apprentissage par interprétations: l'hypothèse h doit être contradictoire avec l'affirmation que l'oiseau a ne vole pas, de ce point de vue $oiseau(a) \wedge \leftarrow vole(a)$ est un exemple négatif.

b) relève de l'apprentissage par prédicats. De h et de l'observation oiseau(a) on veut inférer vole(a). Ici vole(a) est une instance positive de vole.

c) relève d'une forme d'apprentissage, l'apprentissage par dérivation, où on essaie de généraliser une connaissance, il s'agit donc ici d'induction : on cherche h plus générale que $vole(a) \leftarrow oiseau(a)$.

Cette équivalence d'écriture illustre les réductions de l'apprentissage de prédicats et de l'apprentissage par dérivation à l'apprentissage par satisfiabilité proposée par Deraedt.

2.2.1 L'apprentissage par observations. Cas uni-observationnel.

En général l'apprentissage de prédicats, dans le cas d'un prédicat-cible unique, est présenté comme suit:

E+ ({e1+, e2+,....}) et E- ({e1-, e2-,...}) sont des ensembles d'instanciations du prédicat-cible. On dispose d'une théorie B qui contient les connaissances générales et les faits connus. L'ensemble des solutions, dans un certain langage, (en général une solution est une théorie clausale) est défini comme les hypothèses h du langage satisfaisant les conditions suivantes:

$\forall e+ \in E+ , h \wedge B \models e+$ (i)
$\forall e- \in E-, h \wedge B \not\models e-$ (ii)

E+ contient donc les instances positives, et E- contient des instances négatives au sens où elles ne doivent pas être conséquences de B et de h.

Dans cette forme d'apprentissage on a donc essentiellement dans B une seule observation décrivant une réalité, plus des connaissances générales. La relation (ii) pose un problème connu illustré par l'exemple suivant:

E- ={vole(c) ; vole(d)}, B=∅.

h = {vole(c) ∨ vole(d)} est solution, bien que cela soit contraire à l'intuition, car h a pour conséquence que l'une des deux instances est vraie.

Ce problème (qui ne se pose que si les hypothèses peuvent être des disjonctions, ce qui n'est en général pas le cas) ne se pose pas si on considère que ce qui est observé est E_N ={←e1-, ←e2-; ....}, E_N est ainsi assimilé à un ensemble de contraintes (cf. [13, 14, 15, 16]), et que h est consistant avec ce qui est observé plus les connaissances générales, ce qui s'écrit:

h ∧ B ∧ {←e1-, ←e2-; ....} |≠ ❒ (ii')

L'exemple précédent s'écrit alors :

E_N= {←vole(c) ; ←vole(d)}, B=∅.

h = {vole(c) ∨ vole(d)} n'est plus solution car il n'y aucune interprétation qui soit un modèle de {vole(c) ∨ vole(d) et de {←vole(c) ; ←vole(c}}.

En ce qui concerne la relation (i) elle peut se réécrire :

h ∧ B |= e1+∧ e2+ ∧ e3+ ∧ ...

Nous noterons E_P la conjonction e1+∧ e2+ ∧ e3+ ∧ ...
Les instances positives observées doivent être des conséquences de l'hypothèse, de ce qui est observé et des connaissances générales. Cependant ceci n'impose pas que h et B aient un modèle. Cette contrainte s'ajoute naturellement si on considère que les instances négatives font partie de ce qui est observé, la relation (i) se transformant alors en :

h ∧ B ∧ E_N|= E_P (i').

On se contente en général de (i) dans la mesure où la présence de E_N constitué de littéraux négatifs, ne contraint pas, pour les langages utilisés, l'ensemble des solutions.

Cependant, nous utiliserons ici le système (i', ii') comme point de départ et nous l'appellerons apprentissage de prédicat par Observation. Nous l'illustrons sur l'exemple suivant:

E_N= {←vole(c) ; ←vole(d)}, E_P ={vole(a);vole(b)}, B={oiseau(a); oiseau(b); camion(c); vélo(d) }

on a alors :

h ∧{oiseau(a);oiseau(b);camion(c);vélo(d) ;←vole(c) ;←vole(d)}|= vole(a)∧vole(b)
h ∧{oiseau(a); oiseau(b); camion(c); vélo(d) ;←vole(c) ;←vole(d)} |≠ ❒

Ce cadre, dans lequel nous avons une seule observation qui induit deux propriétés à satisfaire, se réduit à l'apprentissage par satisfiabilité de manière évidente comme nous l'illustrons sur cet exemple : il suffit de transformer (i') en

h ∧{oiseau(a);oiseau(b);camion(c);vélo(d) ;←vole(c) ;←vole(d);
←(vole(a)∧vole(b)) } |=❒

En réalité on peut étendre ce cadre uni-obervationnel à un cadre où on dispose de plusieurs observations : imaginons que chaque observation représente une cage contenant deux oiseaux, le prédicat-cible vole est alors étudié dans le contexte d'une cage, et plusieurs observations sont nécessaires pour apprendre ce prédicat. Il est evidemment toujours possible de considérer un seul monde contenant toutes les cages et de se ramener au cadre uni-observationnel, mais dans le cas d'incomplétude que nous présentons au paragraphe suivant cela alourdit beaucoup les calculs. L'application au RNA, si on la considère comme l'apprentissage d'un prédicat "structureCommune" est bien plus efficace en considérant chaque séquence comme une observation, ce que nous avons fait dans notre application.

2.2.2 Apprentissage par observation. Extension du cas uni-observationnel aux cas d'incomplétudes.

On ne peut simplement se ramener au cadre présenté en 2.1, lorsque l'observation est incomplète. Plus précisément nous supposons, comme en 2.1, un ensemble de possibles P= {P1, P2, .. , Pi} correspondant à un état d'incertitude sur l' observation. Nous supposons aussi une théorie B_g contenant des connaissances générales du domaine. Chaque Pi est ici une théorie clausale. Les conditions à satisfaire pour h s'écrivent alors, dans le cadre uni-observationnel:

Il existe Pi tel que
- $h \wedge B_g \wedge Pi \wedge E_N \models E_P$ (i').
- $h \wedge B_g \wedge Pi \wedge E_N \not\models ❒$ (ii').

Précisément nous considérerons ici, pour se comparer au système I-ACL de A.C. Kakas, le cas où Pi s'écrit :

Pi = Bo ∧ Δi

où Bo est une théorie clausale représentant ce que l'on sait de l'observation, et Δi est un ensemble de suppositions, c'est à dire de littéraux instanciés.

Ce système ne peut se réduire à l'extension présentée en 2.1 car le possible Pi est commun aux deux relations. Nous allons illustrer cela sur un exemple. Auparavant remarquons que le cadre d'apprentissage par observation présenté en 2.2.1 est un cas particulier de celui-ci où il n'y a qu'un possible Bo, avec B= Bg ∧Bo. Nous reprenons l'exemple du 2.2.1 en le rendant incertain. Nous supposons ici que les Δi sont les sous-ensembles d'un ensemble de littéraux instanciés incertains Δ.

E_N= {←vole(c) ; ←vole(d)}, E_P ={vole(a);vole(b)},
Bg={leger(X)←petit(X); ←(petit(X)∧lourd(X))},
Bo={oiseau(a); leger(a) ; oiseau(b); camion(c); lourd(c)←lourd(a); vélo(d)}
Δ ={petit(a);petit(b);petit(c);petit(d); lourd(a); lourd(b)}

h= vole(X)←(oiseau(X) ∧ leger(X)) est une solution car pour Δi ={petit(b)}, h satisfait les propriétés (i') et (ii').

Remarquons que h = vole(X) ←lourd(X) n'est pas une solution car pour vérifier (i') il est nécessaire que le Δi choisi contienne lourd(a), mais pour vérifier (ii') il est au contraire nécessaire que le Δi choisi ne contienne pas lourd(a). Nous voyons sur cet exemple qu'il est crucial que les deux conditions soient liées par un Δi commun.

Il est à noter que ce cadre subsume le cadre d'apprentissage abductif et inductif I-ACL de Kakas [10]. Dans ce système on dispose d'une théorie abductive T = (P, A, I), P représentant une théorie du domaine, A un ensemble de prédicats

abductibles, et I un ensemble de contraintes que nous considérerons comme un ensemble de clauses négatives (ce qui respecte entièrement les vues de l'auteur).
On dira que tout ensemble d'atomes instanciés formé à partir d'un sous-ensemble de A est une abduction, et que T dérive abductivement une formule F, notée $T \models_A F$ ssi il existe une abduction Δ tel que $P \wedge \Delta \wedge I \not\models \square$ [7] et $P \wedge \Delta \models F$.
Dans le cadre IACL Kakas définit $T \not\models_A e$ (ou e est un fait) ssi il existe une abduction Δ tel que $P \wedge \Delta \wedge I \not\models \square$ et $P \wedge \Delta \not\models e$ (noté $T \models_A$ not_e dans [10]). De ce fait $T \not\models_A e$ n'est pas défini comme la négation de $T \models_A e$.
Le problème d'apprentissage IACL consiste à, étant donnée une théorie abductive T = (P, A, I), un ensemble de faits exemples positifs $E^+ = \{e^+_1, \ldots, e^+_m\}$ et un ensemble de faits exemples négatifs $E^- = \{e^-_1, \ldots, e^-_n\}$, trouver un nouvelle théorie abductive T' = (P', A, I) (avec $P' \supseteq P$) telle que :
$\exists \Delta$, tel que
(K1) $P' \wedge \Delta \wedge I \not\models \square$
(K2) $P' \wedge \Delta \models E+$
(K3) $\forall e- \in E-, P' \wedge \Delta \not\models e-$

Or si l'on considère que toute clause de P' est définie, tout e- étant un fait, la relation (K3) s'écrit $P' \wedge \Delta \not\models e^-_1 \vee \ldots e^-_n \Leftrightarrow P' \wedge \Delta \wedge (e^-_1 \leftarrow \wedge \ldots e^-_n \leftarrow) \not\models \square$.
$I' = (e^-_1 \leftarrow \wedge \ldots e^-_n \leftarrow)$ est un ensemble de clauses négatives que l'on peut donc assimiler à un ensemble de contraintes qu'on intégre dans l'équation (K1), le système final devient :
$\exists \Delta$ tel que :
(K1) $P' \wedge \Delta \wedge I \wedge I' \not\models \square$
(K2) $P' \wedge \Delta \models E+$

On reconnait le système d'apprentissage par observations étendu (i') et (ii') avec les égalités suivantes $E_N = I'$, $Ep = E+$, $\exists$ Pi revient à $\exists \Delta$, $B_g = P \wedge I$ et h = P' - P, à la réserve près qu' I' n'apparait pas dans l'équation (K2).

3 Aspects pratiques du problème des séquences d'ARN.

Cette application a déja été présentée en [5] et de manière plus détaillée en [2,3]. Nous ne discutons ici que de quelques particularités.
L'application s'exprime dans le cadre présenté en §2.1 si on considère des hypothèses de la forme "$\exists$xyz Chevauche(x,y) $\wedge$ Justapose(x,z) $\wedge$ Chevauche(y,z)". Nous ne disposions pas de contre-exemples, et en conséquence nous avons cherché des hypothèses maximalement spécifiques couvrant au moins une structure possible pour chaque séquence-exemple. Les structures possibles sont obtenues en considérant des contraintes d'incompatibilités entre les entités (chaque entité représente une unité structurelle incertaine). Cependant une caractéristique importante de ce problème est la suivante : seuls les ensembles maximaux de suppositions cohérents avec ce que l'on sait de la séquence doivent être considérés, dans la mesure où, chimiquement, les unités structurelles se forment si leur formation n'est pas empêchée par la formation d'une unité structurelle incompatible. Ceci signifie que nous avons pu utiliser une contrainte forte : les suppositions doivent être considérés comme des défauts normaux au sens de Reiter. Sans cette contrainte forte nous n'aurions probablement pas obtenu les résultats voulus.
Une seconde remarque est que nous n'avons considéré pour chaque séquence que les structures les plus plausibles (au sens chimique du terme).
Une troisième remarque est la suivante: en l'absence de contre-exemples l'incertitude a pour conséquence que de nombreuses hypothèses artefacts apparaissent. Cela est renforcé par le fait que nous avons du relacher la contrainte de complétude en acceptant qu'une proportion seulement (80%) des exemples soient couverts. Nous avons pu éliminer ces

[7] Dans [10] Kakas écrit cette relation $h \wedge P \wedge \Delta \models I$, contredisant ainsi la même contrainte proposée dans [7]. De plus cette relation n'est pas compatible avec le fait que I soit représentée par un ensemble de clauses négatives comme le propose Kakas lui-même.

artefacts en simulant des exemples aléatoires, et en éliminant les hypothèses qui couvraient les exemples avec une proportion proche de celle observée dans les séquences aléatoires.

Enfin, il est aisé de reformuler ce problème dans le cadre de l'apprentissage de prédicat par observations étendu présenté en §2.2, cependant il est là utile de considérer que chaque séquence est une observation, ce qui permet de casser la combinatoire des suppositions.

4 Conclusion

Nous n'avons discuté ici essentiellement que des cadres d'apprentissage en présence d'incomplétude. Nous avons tenté de voir les liens entre différents cadres et nous avons proposé deux formulations générales permettant de rendre compte des situations d'incertitude d'une part lorsqu'on considère le paradigme où l'exemple correspond à une situation observée (représentée par une théorie clausale) et d'autre part le paradigme de l'apprentissage de prédicat, où les exemples sont des instances d'un concept-cible. Nous avons été guidés ici, par la recherche d'une définition de l'espace des solutions compatibles avec les exemples rencontrés. Il reste beaucoup d'aspects opérationnels à étudier et en particulier l'influence des choix de langage .

5 Références

1. B. Billoud, M. Kontic and A. Viari, Palingol: A declarative programming langage to describe nucleic acids' secondary structures and to scan sequence databases, *Nucleic Acids Research*, vol 24, n° 8, pp. 1395-1403, 1996.
2. D. Bouthinon, Apprentissage à Partir d'Exemples Ambigus : Etude Théorique and Application à la Découverte de Structures Communes à un Ensemble de Séquences d'ARN, *doctorat de l'université Paris XIII-Institut Galilée*, 1996.
3. D. Bouthinon et H. Soldano, A new method to predict the consensus secondary structure (possibly including pseudoknots) of a set of unaligned RNA sequences, *BioInformatics* (a paraître)
4. Bouthinon, D. and Soldano, H. (1998a) , Un cadre de programmation logique inductive pour apprendre un concept à partir d'exemples ambigus, *11ème Congrès Reconnaissance des Formes et Intelligence Artificielle (RFIA '98)*, 20-22 Janvier 1998, Clermont-Ferrand (France), pp. 137-147.
5. Bouthinon D., Soldano H. (1998b), An Inductive Logic Programming Framework to learn a Concept from Ambiguous Examples, *10th European Conference on Machine Learning*, Chemintz, Germany, April 1998, *Lecture Notes in Artificial Intelligence*, 1398, pp. 238-249.
6. T.G. Dietterich, R.H. Lathrop and T. Lozano-Perez, Solving the Multiple-Instance Problem with Axis-Parallel Rectangles, *Artificial Intelligence*, vol 89(1-2), pp. 31-71, 1997.
7. Y. Dimopoulos and A. Kakas, Abduction in Inductive Logic Programming, in *Advances in Inductive Logic Programming*, L. De Raedt (Ed.), IOS Press, 1996.
8. P.A. Flach, A Framework for Inductive Logic Programming, in *Inductive Logic Programming,* S. Muggleton (Ed.*),* Academic Pres*s,* 1992.
9. H. Hirsch, Generalizing Version Spaces, *Machine Learning*, 17, 5-46, 1994.
10. A.C. Kakas and F. Riguzzi, Learning with Abduction, *7th International Workshop on Inductive Logic Programming*, Prague, Republic Czesh, *Lecture notes in Computer Science*, Vol 1297, 1997.
11. S. Muggleton, Inductive Logic Programming, in *Inductive Logic Programming ,* Academic Press, 1992
12. L. De Raedt, Logical settings for concept-learning, *Artificial Intelligence*, 95 187-201 (1997).
13. M. Sebag, A constraint-based induction algorithm in FOL, *Machine Learning*, pp. 275-283, 1994.
14. M. Sebag and C. Rouveirol, Induction of maximally general clauses consistent with integrity constraints, *Fourth Int. Workshop on Inductive logic programming* (ILP-94*),* 1994.
15. M. Sebag and C. Rouveirol, Constraint inductive logic programming, in *Advances in Inductive Logic Programming*, L. De Raedt (Ed.), IOS Press, 1996.
16. M. Sebag, C. Rouveirol and J.F. Puget, Inductive Logic Programming + Stochastic Bias = Polynomial Approximate Learning, in *Multi Strategy Learning*, J. Wrete & R.S. Michalski (Ed.), MIT Press, 1996.

17. J. D. Zuker, Comprendre et résoudre les problèmes d'apprentissage multi-instances et multi-parties, 13ème Journées Francophones sur l'Apprentissage, Arras, France, 1998.

Remerciements.
Nous remercions Marc Champesme pour les fructueuses discussions auxquelles il a participé pendant l'élaboration de ce travail. Merci également aux membres de l'Atelier de BioInformatique pour avoir supporté nos bruits et nos fureurs lors de quelques débats orageux.