Modèle cognitif de l'apprentissage inductif de concepts

Vivien Robinet, Gilles Bisson, Mirta B. Gordon, et Benoît Lemaire

Laboratoire TIMC-IMAG (UMR 5525), CNRS, Université de Grenoble I

Je souhaite que cette soumission donne lieu à :

x Communication orale uniquement.

☐ Communication affichée uniquement.

☐ J'accepte que le comité de programme requalifie ma proposition dans un sens ou dans l'autre (communication orale vers communication affichée ou l'inverse).

Modèle cognitif de l'apprentissage inductif de concepts

Résumé – Notre approche consiste à concevoir un modèle cognitif de l'apprentissage inductif de concepts basé uniquement sur le principe de simplicité. Nous utilisons une formulation mathématique de ce principe appelée Minimum Description Length, dans le but de créer un modèle informatique capable de générer automatiquement des hypothèses à partir d'un ensemble de données discrètes. Nous validons notre modèle en le comparant à la production d'humains confrontés à la même situation.

Mots-Clés – Modèle cognitif, MDL, apprentissage inductif, apprentissage de concepts, principe de simplicité.

1. INTRODUCTION

Au cours de notre vie une grande partie des apprentissages se fait de manière inductive à partir de stimuli provenant de notre environnement extérieur (Shepard, 1987). Nous sommes en mesure de repérer certaines invariances et d'en abstraire des régularités nous permettant de mieux comprendre et d'interpréter le monde qui nous entoure. Ainsi, French, Mareschal, Mermillod et Quinn (2004) défendent l'hypthèse que les catégories perceptuelles sont le reflet des représentations internes créées par l'observation statistique des co-occurrences présentes dans l'environnement. Nous apprenons également à combiner ces invariants pour former des concepts plus complexes (Goodman, Tenenbaum, Feldman, & Griffiths, 2008). La question qui nous occupe est de savoir dans quelle mesure il est possible de prédire les concepts que les individus sont susceptibles de créer à partir d'un ensemble de stimuli.

Le principe de simplicité a été proposé à plusieurs reprises comme principe cognitif fondamental (Chater, 1999) permettant de rendre compte de la façon dont l'être humain est capable d'apprendre des concepts à partir d'exemples (Feldman, 2000). Dans sa formulation la

plus générale, le principe de simplicité stipule que, parmi toutes les explications cohérentes avec un jeu de données, chaque individu privilégie la plus simple. Il fournit ainsi une solution au problème de l'induction (Chater, Vitanyi, 2003) en permettant de choisir une unique généralisation, la plus simple, parmi l'ensemble des généralisations qu'il est possible de construire à partir d'un ensemble de stimuli.

Notre but ici est de concevoir un modèle général basé sur le principe de simplicité et faisant le moins possible d'hypothèses annexes, afin d'évaluer jusqu'où il est possible d'aller en matière de prédiction, en utilisant ce seul principe. Nous nous intéressons à l'apprentissage inductif de concepts de façon non-supervisée, en nous focalisant plus particulièrement sur le cas où les stimuli sont temporellement discrets et peuvent être mis sous forme d'une liste non-ordonnée de symboles.

Nous avons conçu un modèle informatique simple implémentant ce principe. Nous lui avons soumis un ensemble de stimuli, puis nous avons comparé les généralisations créées par notre modèle à celles créées par les humains dans la même situation afin d'évaluer la capacité prédictive du modèle.

2. LE PRINCIPE DE SIMPLICITÉ

Un résultat très important (Li & Vitanyi, 2000) permet de relier la capacité prédictive d'un modèle, à la simplicité de sa formulation. Parmi un ensemble de modèles également descriptifs, celui qui peut être formulé le plus "simplement" est également celui qui a la plus grande probabilité d'être prédictif.

Si le lien entre simplicité et capacité prédictive est relativement récent, il est en réalité la formulation mathématique rigoureuse d'un principe bien plus ancien, connu sous le nom de rasoir d'Occam.

Si l'on prend l'exemple du genre des mots en "age" en partant des données {d1, d2, d3, d4, d5} suivantes :

d1 : message	M.
d2 : plage	F
d3 : mariage	M
d4 : partage	M
d5 : saxifrage	F

Alors les deux modèles m1 et m2 ci-dessous permettent de rendre compte du genre des données :

m1 : Les mots en "age" sont masculins sauf plage et saxifrage.

m2 : Les mots en "age" sont féminins sauf ceux qui ont un nombre paire de consonnes ou ceux qui ont sept lettres et commencent par la lettre "m".

Les deux modèles ont le même pouvoir descriptif, cependant m1 qui se "formule plus simplement" a un pouvoir prédictif plus élevé (environ 99%) tandis que m2 a un pouvoir prédictif faible (environ 50% lorsqu'on l'applique à l'ensemble des mots de la langue française).

Dans le cas général, déterminer le modèle le plus simple peut s'avérer difficile. Le principe de Minimum Description Length ou MDL (Rissanen, 1978), va fournir un cadre mathématique rigoureux au concept intuitif de "simplicité". Notre idée est d'utiliser ce principe pour modéliser l'apprentissage inductif de concepts par les humains.

3. MDL COMME PRINCIPE DE SIMPLICITÉ

Le principe de Minimum Description Length, littéralement "longueur minimale de description", est un critère permettant de déterminer entre deux modèles élaborés à partir de données empiriques, lequel est le plus simple. La simplicité d'un modèle s'exprime en terme de taille de codage, cette taille de codage étant dépendante du langage de représentation choisi.

3.1. Principe de MDL

Il existe plusieurs formes de MDL. La plus simple appelée "two part coding", consiste à représenter l'ensemble de données en deux parties. La première partie, qui correspond au modèle, synthétise les régularités présentes dans les données. Dans la seconde partie se trouvent les données, codées d'une façon concise où les régularités recensées dans la première partie n'apparaissent pas. Ce codage biparti permet une représentation moins redondante dans la mesure où les régularités sont définies une fois pour toutes dans la partie modèle et non plus pour chaque donnée.

Dowman (2007) utilise le principe du MDL pour identifier la grammaire la plus adéquate permettant de décrire un ensemble de phrases. Le codage biparti permet de trouver un compromis entre l'exigence de concision de la grammaire (qui, seule, pourrait aboutir à une généralisation tellement simple qu'elle engendrerait des phrases incorrectes) et l'exigence d'adéquation aux données (qui seule, pourrait produire une grammaire extrêmement compliquée).
Capturer les régularités permet de passer de l'espace initial des données à un espace de représentation des données sachant le modèle. Dans ce deuxième espace, le nombre de degrés de liberté qu'il est nécessaire de fixer pour reproduire les données est plus faible, car certains sont déjà fixés par le modèle. La partie modèle peut être vue comme l'information partagée par un grand nombre de données, tandis que la partie de recodage des données sachant le modèle, représente l'information propre à chaque donnée, que l'on considère généralement comme du bruit dans le cadre d'une modélisation.
Appliquer le principe de MDL consiste à minimiser la taille totale du codage, qui est la somme de la taille de codage du modèle d'une part, et des données sachant le modèle d'autre part.
Pour illustrer l'intérêt de cette séparation on crée les trois modèles suivants pour rendre compte du genre des mots en "age" dans la langue française :

m1 : ∅
m2 : Les mots en "age" sont masculins
m3 : Les mots en "age" sont masculins sauf ceux qui :
- ont 4 lettres et ne commencent ni par "g" ni par "m".
- commencent par "im"
- ont comme préfixe "pl", "saxifr" ou "ferl"

Le modèle m1 est vide. La partie des données|modèle doit donc coder l'intégralité de l'information, c'est-à-dire spécifier le genre pour chacun des 869 mots en "age" de la langue française. À l'inverse, le modèle m3 qui capture l'intégralité de l'information, permet à lui seul de prédire parfaitement le genre des mots en "age". Aucune information supplémentaire n'est nécessaire pour reconstruire les données (taux de prédiction de 100%), mais cela se fait au prix d'un modèle très complexe. Le modèle m2 qui est intermédiaire, capture juste l'information spécifique à la majorité des données (les mots en

"age" sont masculins) conduisant à un taux de prédiction de 99% pour un modèle relativement simple. Cependant 1% de l'information doit être spécifiée dans les données|modèle : il s'agit des 10 exceptions à la règle.
Dans le cas de m1 (resp. m3), l'intégralité de l'information est contenue dans la partie modèle (resp. données|modèle). Ils conduisent tous deux à un système global complexe. Dans le cas de m2, la distribution intelligente de l'information entre le modèle et les données sachant le modèle, permet d'aboutir à un système global plus simple.
Appliquer le principe de MDL pour prédire le genre des mots conduit à la conception d'un modèle ayant la forme de m2 (Marchal, Lemaire, Bianco, Dessus, 2008).

4. FONCTIONNEMENT DU SYSTÈME

Nous définissons le terme système comme étant l'assemblage de la partie "modèle" et de la partie "données sachant le modèle". Dans notre cas, le modèle est constitué d'une liste de concepts, chaque concept représentant une régularité observée au sein des données.
Le système que nous avons conçu se veut un modèle cognitif de l'apprentissage inductif de concepts. Il fonctionne incrémentalement, par l'agrégation successive de concepts, partant de concepts élémentaires, pour créer des concepts agrégés de façon hiérarchique. La figure 1 illustre cette hiérarchie avec des concepts simples.

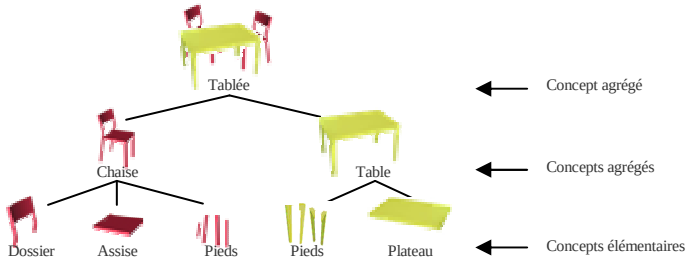


Figure 1. Exemple de création hiérarchique de concepts par agrégation de concepts plus simples.

4.1. Structure générale

Notre modèle cognitif de l'induction est découpé en quatre étapes, présentées sur la figure 2 :

(1) Codage des stimuli
(2) Encodage des données sachant les concepts
(3) Création de nouveaux concepts
(4) Réencodage des données sachant les nouveaux concepts

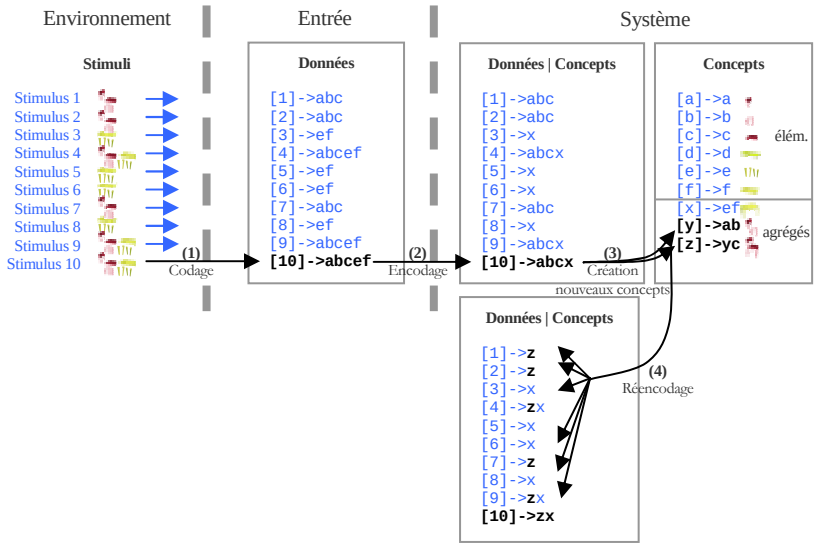


Figure 2. Schéma général du modèle cognitif de l'induction présentant les quatre étapes consécutives à l'ajout du stimulus n°10.

Chaque stimulus en provenance de l'environnement est codé dans un format compréhensible par le système (étape 1), puis ajouté à l'ensemble des données. Le système encode (reformule) ensuite les données, en utilisant les concepts qu'il possède (étape 2). Puis il teste si la création d'un nouveau concept pourrait permettre de réencoder les données de façon plus courte (étape 3). Dans l'affirmative, il réencode les données en prenant en compte le nouveau concept (étape 4). Ce processus recommence à la présentation du stimulus suivant.

Nous allons voir en détail le fonctionnement des étapes (1), (2), (3) et (4).

4.2. Étape 1 : codage des stimuli

Les stimuli provenant de l'environnement doivent être codés dans un format compréhensible par le système. Pour cela, on établit un codage élémentaire des données, sous la forme d'une liste non-ordonnée de symboles que nous appelons concepts élémentaires, comme illustré sur la figure 3.

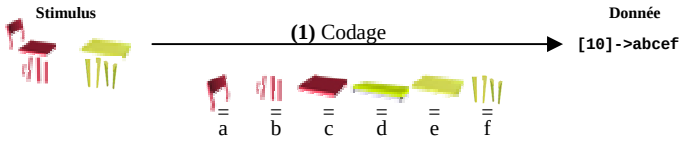


Figure 3. Exemple de codage. À chaque concept élémentaire correspond un symbole.

Le choix des concepts élémentaires utilisés par le système est primordial car il détermine quels éléments de l'environnement seront traités par le système. Il n'est pas nécessaire que les concepts élémentaires soient indépendants. Ils doivent en revanche fournir un langage de description assez riche pour que les concepts agrégés que l'on souhaite concevoir soient exprimables uniquement comme conjonction de concepts élémentaires.
Les données ainsi que les concepts, qu'ils soient élémentaires ou agrégés, sont représentés en utilisant le même formalisme.
Le concept "[x] ef" par exemple, est constitué du symbole "[x]" jouant le rôle de nom permettant à un autre concept d'y faire référence, et de la liste non ordonnée de symboles "ef" jouant le rôle de définition permettant de décrire le contenu du concept. Chaque symbole de la définition fait référence à un autre concept de niveau inférieur, et ainsi de suite hiérarchiquement jusqu'aux concepts élémentaires.
Pour calculer la taille de codage des deux parties du système, ces dernières sont mises sous forme de chaînes de caractères, comme présenté sur la figure 4.

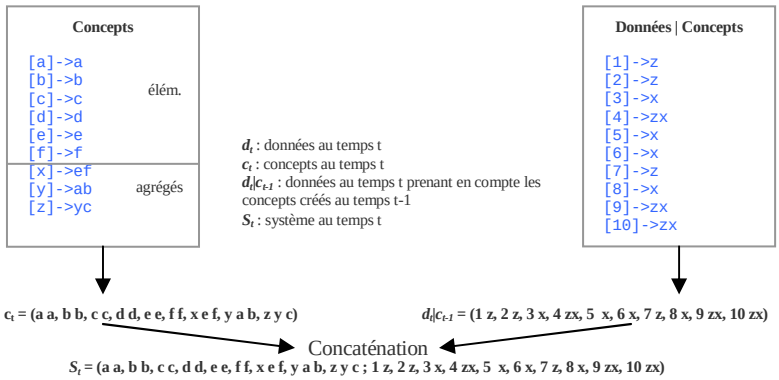


Figure 4. Exemple d'écriture sous forme de chaîne de caractères, des "concepts" c_t et des "données | concepts" d_t/c_{t-1}. Le séparateur utilisé est la virgule. La concaténation des deux est notée S_t.

La chaîne de caractères S_t d'éléments s_t^i ($1 \leq i \leq |S_t|$) résultant de la concaténation des deux chaînes c_t et $d_t|c_{t-1}$, contient la même information que les données de départ mais sous une forme différente. La théorie de l'information fournit le cadre pour calculer la taille de codage minimale que l'on obtiendrait si l'on souhaitait compresser S_t de façon à tenir compte des redondances internes. Il a été montré par Shannon que chaque symbole s apparaissant avec une probabilité p_s peut être idéalement compressé en un code de taille $I(s)$=-log(p_s), où $I(s)$ est appelée information propre de s. La taille de codage minimale de la chaîne complète, et donc du système, est donnée par :

$$Taille_codage(S_t) = \sum_{i=1}^{|S_t|} I(s_t^i)$$

Dans notre cas, p_s est approximé par f_s, la fréquence empirique d'apparition de s dans la chaîne de caractères. Pour plus de simplicité, on néglige la taille nécessaire au codage des séparateurs ainsi que celle nécessaire au codage du nom des données "[1]", "[2]", etc. En effet, une donnée peut être exprimée en fonction de concepts, mais pas en fonction d'une autre donnée, rendant inutile le codage du nom des données. La chaîne S_t' réellement utilisée est :

S_t' = (a a b b c c d d e e f f x e f y a b z y c z z x z x x x z x z x z x)
et comporte 34 symboles. Pour cet exemple, le concept "[x] ef" peut-être idéalement compressé par un code de taille :

$$I(x)+I(e)+I(f)=-\log\left(\frac{8}{34}\right)-\log\left(\frac{3}{34}\right)-\log\left(\frac{3}{34}\right)$$

Si on utilise la base deux pour le logarithme, la taille est exprimée en bits. On trouve pour "[x] ef" une taille de codage idéale de 9.09 bits. En pratique, il n'est pas nécessaire de trouver quel codage pourrait permettre d'atteindre cette taille, puisque les données ne sont pas réellement compressées. Il suffit d'être capable de déterminer la taille qu'aurait chaque concept si l'on utilisait ce codage optimal, afin de prédire l'impact de la création d'un nouveau concept sur la taille totale du système (concepts + données|concepts).

4.3. Étape 2 : encodage des données sachant les concepts

Si l'ajout de nouveaux concepts, par exemple "[x] ef", "[y] ab" et "[z] yc", augmente la taille de codage du modèle, cela permet en contrepartie de réexprimer les données de façon plus courte. Ainsi la donnée "[4] abcef" sera encodée plus simplement sous la forme "[4] zx", diminuant ainsi la taille globale du système. Créer un nouveau concept n'est intéressant que si son coût est compensé par un bénéfice supérieur en terme d'encodage des données.

Trouver l'encodage permettant d'exprimer une nouvelle donnée de la façon la plus concise en terme des concepts est une tâche difficile dans le cas général. En effet, la décomposition en concepts n'est pas unique, et il faut trouver le plus court parmi l'ensemble des encodages possibles.

Pour cela, le système utilise un algorithme de recherche arborescente de type A* (Pearl, 1990), mais ne garantit pas l'optimalité du résultat trouvé. En effet, il faudrait pour cela disposer d'un minorant de la distance entre l'étape courante et le but, et pas simplement d'une estimation de cette distance, comme le fait notre modèle.

Notre algorithme utilise les résultats fournis par le MDL, mais pourrait être remplacé par n'importe quel autre processus remplissant la même fonction.

4.4. Création de nouveaux concepts (3)

L'algorithme crée de nouveaux concepts, utilisables par la suite pour encoder les données et ainsi permettre une diminution de la taille totale du système. Cette création permet un gain de taille de codage d'autant plus important que le nouveau concept correspond à une régularité fréquente dans les données et qu'il est peu coûteux à définir.
Connaissant le nombre de fois où deux symboles a et b co-occurrent, il est possible de calculer l'augmentation ou la diminution de taille de codage qu'entraînerait la création du concept "[y] ab". Parmi tous les concepts envisagés, celui conduisant à la plus grande diminution de la taille de codage du système est ajouté à la liste des concepts agrégés. Il faut remarquer que le fait de créer toujours le meilleur concept en terme de taille de codage ne garantit pas l'optimalité du résultat obtenu.

4.5. Réencodage des données (4)

Après la création d'un concept nouveau, les données sont réencodées en tenant compte de ce concept. Par exemple, si un concept "[y] ab" est créé par agrégation des concepts "[a]" et "[b]", alors toutes les cooccurrences de a avec b sont remplacées par y.

5. EXPÉRIMENTATION

Il importe de mettre en œuvre une expérimentation afin de tester si il y a ou non adéquation entre les concepts créés par le modèle et ceux que créerait un humain dans la même situation. Il faut donc être en mesure de pouvoir placer l'humain et le système informatique face à des situations identiques, et faire en sorte que les concepts créés utilisent le même formalisme afin de pouvoir être comparés. Cela nécessite de surmonter deux difficultés inhérentes à ce type d'expérimentation.
La première est la difficulté d'avoir accès aux concepts construits par des participants humains, car le fait qu'un individu soit capable d'abstraire des régularités ne signifie pas qu'il soit capable de les formuler explicitement. A ce problème s'ajoute une seconde difficulté liée aux prérequis conceptuels mis en jeu lors de l'expérimentation. Tester l'apprentissage de concepts de la vie de tous les jours (chaise,

table) chez des sujets adultes doit tenir compte du fait que le concept à apprendre puisse être déjà partiellement connu. De plus, des prérequis (concepts de niveaux inférieurs) sont généralement nécessaires à la définition d'un concept de niveau élevé. Il faut imaginer que de tels prérequis puissent être très différents d'un individu à l'autre et difficilement quantifiables dans le cadre d'une expérimentation rigoureuse.

Une façon de surmonter ce problème est de s'assurer qu'il n'existe aucune familiarité avec les concepts que l'on souhaite apprendre en créant artificiellement un environnement pour les besoins de l'expérience. Cela soulève bien évidemment le problème de l'adéquation des comportements observés en laboratoire avec les situations de la vie réelle. Une autre façon de procéder et de tester le modèle sur de jeunes enfants dont on peut facilement être sûr qu'il ne possèdent aucune connaissance du concept à apprendre et dont les prérequis sont assez proches des perceptions sensorielles pour que les différences inter-individuelles soient faibles (French et al, 2004).

Pour cette expérimentation, nous avons choisi la première solution consistant à créer des concepts de façon artificielle.

Fisher et Aslin, (2001) ont créé un tel contexte artificiel sous la forme de grilles composées de symboles non familiers organisés selon une structure prédéfinie. Ils montrent que les humains sont capables d'apprendre des éléments de cette structure à partir de la simple observation répétée d'un grand nombre de grilles et notamment des associations indépendantes de la position des symboles. Le matériel de notre expérience reprend certains de ces symboles.

5.1. Matériel

Cette expérimentation consiste à proposer aux participants un environnement artificiel suscitant la création de concepts agrégés puis à comparer les concepts produits et le décours temporel de leur construction avec les résultats du modèle sur les mêmes données. Cet environnement comporte vingt symboles non familiers préalablement organisés selon une hiérarchie conceptuelle que les participants ne connaissent évidemment pas. La figure 5 illustre cette hiérarchie.

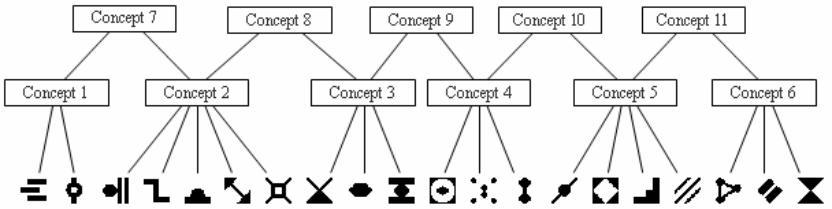


Figure 5. Structure ayant permis de générer le jeu de données utilisé durant l'expérimentation.

Dans l'expérience, on présente à l'individu une succession de planches de symboles. Chaque planche contient les symboles correspondant à un ou plusieurs concepts tirés aléatoirement dans la hiérarchie. Chaque concept de premier ou de deuxième niveau, a une probabilité de 10% d'être sélectionné, ce qui conduit à une dizaine de symboles par planche. Un mécanisme de génération aléatoire de bruit provoque de temps en temps l'ajout ou la suppression d'un symbole (en moyenne une erreur toutes les 2,5 planches). L'apparence visuelle des symboles est tirée aléatoirement avant le début de chaque expérimentation afin de limiter les effets dus à d'éventuelles associations basées sur les caractéristiques morphologiques des symboles.

5.2. Participants

Une expérience consiste en la présentation de 75 planches, les une après les autres sur un écran d'ordinateur. Chaque participant (18 au total) a comme consigne de déplacer avec la souris tous les symboles de la planche vers une zone particulière. Pour alléger cette tâche fastidieuse, ils peuvent créer des concepts agrégés qui leur permettront de déplacer en une seule fois les symboles regroupés. La figure 6 montre l'interface utilisée, où l'on voit que le participant a déjà créé trois concepts agrégés (en bas à gauche) ; chaque fois que ces symboles apparaissent ensemble, le logiciel les marque d'une même couleur dans l'historique (à droite) et permet leur déplacement global. Un même symbole peut appartenir à des concepts différents et peut donc être marqué de plusieurs couleurs.

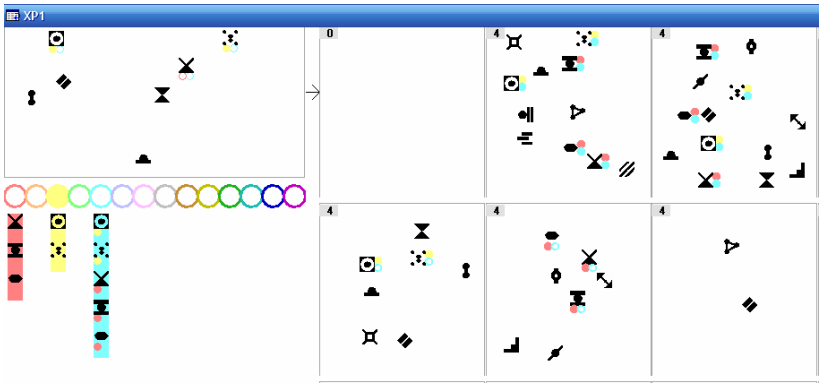


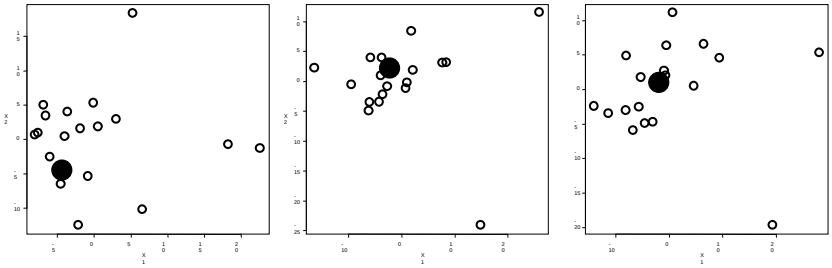
Figure 6. Interface utilisée lors de l'expérimentation. Les symboles apparaissant à gauche doivent être déplacés avec la souris dans la première case de l'historique. Trois concepts ont déjà été créés par le participant.

Afin de motiver les participants, il leur avait été signalé que la création appropriée de concepts leur rapportait des points, et qu'un classement final entre les participants serait établi, mais le barème de points ne leur était pas communiqué à l'avance.
Nous avons soumis le même problème à notre modèle, qui a créé des concepts, appliquant le principe du MDL comme expliqué dans la section 3.

5.3. Résultats

Parmi les onze concepts qu'il était possible d'apprendre, neuf ont été trouvés par presque tous les participants ainsi que par le modèle. Le concept 8 n'a été créé par aucun des participants ni par le modèle, et le concept 7 qui n'a pas été créé par le modèle a été remarqué par seulement deux participants. Quatre-vingt sept nouveaux concepts ont été créés de manière anecdotique, le plus souvent par un seul participant. La plupart de ces concepts sont en réalité des variations incomplètes de concepts existants. Deux nouveaux concepts qui n'existaient pas dans le matériel de départ ont été créés par un grand nombre de participants ainsi que par le modèle. ce qui est encourageant dans la mesure où cela montre que le modèle est capable

de reproduire les inférences faites par les participants, même dans le cas où elle ne font pas partie de ce qui était censé être appris.
Au niveau temporel, les concepts créés par le modèle le sont sensiblement en même temps que ceux créés par les participants, avec en moyenne un léger retard de quelques itérations.
Le modèle semble se comporter comme un individu type dans la mesure où il crée uniquement les concepts partagés par la majorité des participants et non les concepts anecdotiques propres à chacun. Cela confère au modèle une position centrale dans l'espace des concepts. Cet espace étant de dimension très élevée (99), nous utilisons un positionnement multidimensionnel (figure 7) pour projeter les individus sur le plan de plus forte inertie.



Itération 25 Itération 50 Itération 75

Figure 7. Projection bidimensionnelle dans l'espace des concepts, des 18 participants (cercles évidés) et du modèle (cercle plein), au premier tiers, au second tiers, et à la fin de l'expérimentation.

Pour quantifier le rôle central occupé par le modèle, nous avons calculé pour les différentes itérations la distance moyenne de chaque participant à l'ensemble des autres (le modèle étant considéré comme un participant supplémentaire). Le classement obtenu place le modèle à la troisième place des positions les plus centrales avec une distance moyenne aux autres participants de 11.3, la moyenne sur tous les participants étant de 15.2.

6. CONCLUSION

Nous avons présenté un modèle cognitif de l'induction de concepts. Ce modèle se base uniquement sur le principe de MDL, qui est une formulation mathématique du principe de simplicité. Les résultats montrent une bonne adéquation avec les données expérimentales en accordant au modèle une position centrale, ce dernier pouvant être vu comme un individu type ne créant que les concepts partagés par la majorité des individus. Ces résultats sont remarquables, car notre modèle ne comporte aucun paramètre ajustable.

L'apprentissage de nouveaux concepts se faisant par conjonction de concepts de niveau inférieur, le système ne permet pas d'exprimer les disjonctions ou les négations, ce qui limite son pouvoir descriptif. Une autre limitation réside dans le format non ordonné des symboles perçus en entrée, rendant équivalent les stimuli "abcd" et "cbad", empêchant entre autre la modélisation de certains processus linguistiques comme l'apprentissage de grammaires.

Le système se base sur le seul principe de simplicité, cependant comme tout modèle informatique, une expertise est nécessaire afin de choisir les concepts élémentaires pertinents pour décrire le domaine considéré.

Nous avons montré que dans le cas où aucun prérequis n'est nécessaire à l'apprentissage, le modèle permet une bonne prédiction de ce que font les humains. Dans le cas contraire, lorsque les concepts à apprendre font référence à d'autres concepts déjà connus, il est nécessaire d'intégrer cette connaissance au système lors de l'initialisation. Cette tâche est potentiellement délicate car il n'est pas facile d'avoir accès aux concepts possédés par les humains, ces derniers pouvant de plus être sujets à une forte variabilité interindividuelle. En absence de cette initialisation, l'apprentissage effectué par le système risque très vide de diverger de celui des humains, les mêmes stimuli risquant d'être encodés différemment par l'individu et le système, pouvant conduire à des interprétations différentes.

7. BIBLIOGRAPHIE

Chater, N., & Vitanyi, P. (2003). Simplicity: a unifying principle in cognitive science? *Trends in Cognitive Sciences*, *7*(1), 19-22.

Dowman, M. (2007). Minimum Description Length as a Solution to the Problem of Generalization in Syntactic Theory. *http://www.ling.ed.ac.uk/~mdowman/mike_dowman_publishing.html* [Non publié].

Feldman, J. (2000). Minimization of Boolean complexity in human concept learning. *Nature*, *407*(6804), 630-633.

Fiser, J., & Aslin, R. N. (2001). Unsupervised Statistical Learning of Higher-Order Spatial Structures From Visual Scenes. *Psychological Science*, *12*(6), 499-504.

French, R. M., Mareschal, D., Mermillod, M., & Quinn, P. C. (2004). The role of bottom-up processing in perceptual categorization by 3-to 4-month-old infants: Simulations and data. *Journal of Experimental Psychology: General*, *133*(3), 382-397.

Goodman, N. D., Tenenbaum, J. B., Feldman, J., & Griffiths, T. L. (2008). A Rational Analysis of Rule-Based Concept Learning. *Cognitive Science: A Multidisciplinary Journal*, *32*(1), 108.

Li, M., & Vitanyi, P. (1997). *An Introduction to Kolmogorov Complexity and Its Applications* (2 éd.). Springer.

Marchal, H., Lemaire, B., Bianco, M., Dessus, P. (2008). A MDL-based Model of Gender Knowledge Acquisition. *Proceedings of the 12th Conference on Computational Natural Language Learning (CoNLL'08).*

Mareschal, D., French, R. M., & Quinn, P. C. (2000). A connectionist account of asymmetric category learning in early infancy. *Developmental Psychology*, *36*(5), 635–645.

Pearl, J. (1990). *Heuristique, stratégies de recherche intelligente pour la résolution de problèmes par ordinateur* (p. 383). Cépaduès.

Rissanen, J. (1978). Modeling by shortest data description. *Automatica*, *14*(5), 465–471.

Shannon, C. E. (1948). A Mathematical Theory of Communication. *Bell System Tech. J.*, *27*(3), 379-423.

Shepard, R. (1987). Toward a universal law of generalization for psychological science. *Science*, *237*(4820), 1317-1323.

Vitanyi, P., & Ming Li. (2000). Minimum description length induction, Bayesianism, and Kolmogorov complexity. *Information Theory, IEEE Transactions on*, *46*(2), 446-464.