

# Introduction aux réseaux conceptuels appliqués à l'apprentissage automatique des machines

par

Patrice BOUCHER

THÈSE PRÉSENTÉE À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE  
COMME EXIGENCE PARTIELLE À L'OBTENTION  
DU DOCTORAT EN GÉNIE  
Ph.D.

MONTRÉAL, LE 2 NOVEMBRE 2023

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE  
UNIVERSITÉ DU QUÉBEC



Patrice Boucher, 2023



Cette licence Creative Commons signifie qu'il est permis de diffuser, d'imprimer ou de sauvegarder sur un autre support une partie ou la totalité de cette oeuvre à condition de mentionner l'auteur, que ces utilisations soient faites à des fins non commerciales et que le contenu de l'oeuvre n'ait pas été modifié.

**PRÉSENTATION DU JURY**

CETTE THÈSE A ÉTÉ ÉVALUÉE

PAR UN JURY COMPOSÉ DE:

M. Patrick Cardinal, Directeur de Thèse  
Département de génie logiciel et TI, École de Technologie Supérieure.

M. Pierrich Plusquellec, Co-directeur  
Centre d'Études sur le Stress Humain, École de PsychoÉducation, Centre de Recherche de  
l'Institut Universitaire en Santé Mentale de Montréal, Université de Montréal.

Mme. Sylvie Ratté, Présidente du Jury  
Département de génie logiciel et TI, École de Technologie Supérieure.

M. François Grondin, Examineur Externe Indépendant  
Génie électrique et génie électronique, Université de Sherbrooke

ELLE A FAIT L'OBJET D'UNE SOUTENANCE DEVANT JURY ET PUBLIC

LE 10 OCTOBRE 2023

À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE



## AVANT-PROPOS

Ma thèse initiale s'intéressait à réaliser un système ubiquitaire de détection du stress aigu à partir de données provenant de dispositifs portables, qui mesurent les activités cardiaques, respiratoires et physiques. Pour se faire, nous avons réalisé un corpus de données (2014-2016) nommé *Physiostress* (Boucher *et al.*, 2016) avec une trentaine de sujets soumis à des expériences de stress aigu et d'activités physiques, où l'ampleur du stress aigu est évalué à partir d'annotations de stress ressenti par le participant, de stress apparent et en considérant des mesures de cortisol <sup>1</sup>.

En 2016, j'étudiais les réseaux de neurones profonds afin d'effectuer la prédiction de stress à partir des données physiologiques. Les réseaux de neurones profonds permettent de résoudre efficacement des problèmes de classification et de régression complexes dans une multitude d'applications (Alzubaidi *et al.*, 2021). Le problème avec mes données de stress est essentiellement que la sortie du réseau (l'état de stress) suit de façon désynchronisée et très partielle les données physiologiques en entrée. Pendant que les données physiologiques sont actualisées à chaque 15 millisecondes, les annotations de stress ressenti sont actualisées au deux minutes (8000 fois plus lent que l'actualisation des entrées), et les prises de cortisol environ aux 15 minutes (60'000 fois plus lent). L'état des sorties n'est donc pas précisément connu à chaque instant. Dans ce contexte, un réseau de neurones simple ne permet pas d'encoder les divers phénomènes physiologiques à la base des états de stress et d'activités ; pendant que la pauvreté des annotations ne permet pas davantage d'entraîner adéquatement un réseau suffisamment complexe. Cette problématique nous a encouragés à rechercher un système habile pour modéliser les phénomènes physiologiques de façon non supervisée, lesquels seraient liés, dans un deuxième temps, au niveau de stress.

Dans cette optique, en 2017, j'ai commencé à m'intéresser à la détection des séquences caractéristiques de données physiologiques liées à l'état de stress. Plus particulièrement, j'ai réalisé un système qui détectait des récurrences dans les données, qui étaient ensuite associées à des états

---

1. La complexité de la relation entre le stress aigu et les mesures de cortisol exigea un effort d'analyse appréciable, et je publiai mes conclusions dans une revue d'endocrinologie : voir Boucher & Plusquellec (2019).

de stress et des états physiques. Chaque *réurrence* était définie par un ensemble de N valeurs pouvant chacune varier selon une certaine marge. Le groupe de *réurrences caractéristiques* devait être suffisamment complet, généraliste et utile pour que si une nouvelle expérience survient, elle puisse être décrite par des unités déjà présentes dans le groupe et rattachées à un état de stress et à un état physique en particulier. L'analyse statistique des séquences d'observations pour déterminer ce groupe d'unités, bien que réalisé par un programme, était plutôt lourde et l'approche me sembla pour le moment peu praticable.

Pour synthétiser le comportement d'une grande quantité de variables comportant des valeurs variées, j'ai ensuite eu l'idée d'employer des modèles mathématiques : équations par parties et à plusieurs variables. Le système pourrait automatiquement inférer, pour chaque plage de valeurs des entrées, le modèle d'équations le plus approprié : modèle linéaire, sinusoïdal, logarithme... ou encore un modèle plus complexe construit à partir de plusieurs modèles simples. La première étape était de pouvoir inférer les paramètres de chaque modèle à partir d'observations bruitées. J'ai alors recherché des méthodes pour faire ce travail. J'ai proposé, en particulier, une méthode pour déterminer les paramètres d'une fonction sinusoïde, que je prévoyais soumettre à une revue, mais le temps avançait et l'atteinte de l'objectif initial (utiliser le modèle mathématique pour prédire le stress) semblait encore loin d'être atteint, puis une idée détourna mon attention lors d'une conférence à l'École de technologie supérieure (*International Conference on Image Processing Theory, Tools and Applications, Novembre 2017*), en visualisant un schéma d'un réseau de neurones profond. Elle fut la première inspiration déterminante de cette thèse : *et si les expériences en entrées seraient encodées, non seulement via les valeurs des unités du réseau, mais surtout à même la structure du réseau ?*

Partant de cette réflexion, j'ai réalisé suivant la conférence (jusqu'à la fin 2017) un système pour la reconnaissance de chiffres écrits où chaque chiffre est représenté par un ensemble d'unités recouvrant sa représentation. Les unités sont liées par des liens d'attraction et de répulsion : les unités d'un même modèle se repoussent (jusqu'à l'atteinte de leur distance initiale) ; pendant que les unités de modèles distincts s'attirent. Pour tester qu'une nouvelle représentation ressemble à un chiffre, on superpose son modèle à celui du chiffre : les unités de l'un étant attirées

par celles de l'autre. Le chiffre le plus plausible est présumé être celui qui engendre le moins de déformations des deux modèles. Ce modèle, applicable seulement à la reconnaissance de chiffres écrits dans son état brut, se présentait plutôt limité, peu performant et plus comique que convainquant - considérant à l'écran les chiffres se déformer en des configurations inusitées. Finalement, l'idée initiale de la conférence pouvait être poussée beaucoup plus loin, avec l'effort de réflexion adéquate...

*Si le rattachement d'une nouvelle expérience aux expériences antérieures se basait sur l'identité des unités qui décrivent les composantes de l'expérience - chacune de façon spécifique ?*

En cherchant à réaliser ces unités, quelles en seraient leurs natures, leurs compositions, leurs propriétés et leurs interactions ? Comment seraient-elles construites, détectées, utilisées ? De plus, on doit différencier les unités qui représentent des zones d'entrées spécifiques (nous verrons les *noeuds de transpositifs d'entrées*) de celles qui caractérisent les phénomènes liés à l'état ou variation d'états des entrées, détectables sous différentes zones d'entrées (nous verrons les *concepts*).

Étant donnée la complexité de la structure logique des expériences phénoménologiques encodées dans un tel réseau, ce dernier doit impérativement se construire de façon dynamique d'une expérience à l'autre. On observera ainsi un nombre croissant d'unités et de relations s'adaptant à chaque nouvelle expérience. Les unités comme les relations seraient ainsi, pour la majorité, a priori inconnues par l'investigateur du système.

Nous avons ici beaucoup de problèmes qui expliquent pourquoi la théorie des réseaux conceptuels subit une dizaine de révisions majeures, depuis début 2018, avant d'aboutir à celle présentée dans cette thèse ; et pourquoi il fut difficile de revenir à une thèse sur le stress aigu. La théorie a été réalisée conjointement à un programme Java utilisé pour la preuve de concept<sup>2</sup>. L'application de reconnaissance de chiffres écrits a été considérée plus en particulier pour tester l'applicabilité du modèle. Ce n'est que vers la fin du projet (juillet 2022 à février 2023) que

---

2. À chaque révision, le code était recopié dans un nouveau projet et restructuré selon de nouveaux principes.

## VIII

nous avons également testé la reconnaissance de la parole, puis à la toute fin (mars 2023) la reconnaissance du locuteur.



## REMERCIEMENTS

J'aimerais d'abord remercier tous ceux et celles avec qui j'ai travaillé dans les divers projets de mon parcours. Je remercie Pierre Dumouchel, mon premier directeur de thèse qui m'a accueilli à l'ÉTS avec le projet de système de détection de stress aigu. Je remercie la formidable équipe du Centre d'étude sur le stress humain de l'Institut universitaire en santé mentale de Montréal, en particulier ceux et celles avec qui j'ai réalisé le corpus *Physiostress* : Pierrich Plusquellec, Pierre Dufour, Yannick Fouda, Amélie Paulus, Charles Boisvert, Florence Landry et Marie-Josée Richer. Je garde de bons souvenirs du dynamisme de l'équipe et de sa capacité à réaliser le travail dans la bonne humeur. Je remercie Patrick Cardinal, mon directeur de thèse, pour son ouverture d'esprit, sa patience persévérante vis-à-vis mes multiples recherches et son efficacité pour m'aider à la finalisation de la thèse.

Je remercie tous ceux et celles qui m'ont supporté durant ces longues années, toujours trop courtes : ma compagne Virginie Grenier-Deschènes qui me soutient de jour en jour avec mon fils Alfred Boucher qui m'encourage, indirectement, à conclure mes études ; mes soeurs Myriam Boucher et Christine Boucher, mon beau frère Pierre-Luc Lecours et mes parents Roger Boucher et Jacinthe Saint-Pierre qui s'intéressèrent aux multiples développements de la présente recherche et m'encouragèrent à persévérer.

Je remercie, enfin, mon univers musical avec tous mes amis qui se reconnaîtront, qui a pu marquer le fil du temps de ma vie et m'ancrer à terre au travers cet univers d'idées où l'esprit semblait emporté toujours plus loin de mon corps dans des réflexions, remises en question, recherche d'idées perpétuelles de jour en jour, semaine en semaine, année en année.



# Introduction aux réseaux conceptuels appliqués à l'apprentissage automatique des machines

Patrice BOUCHER

## RÉSUMÉ

Cette thèse introduit un modèle d'apprentissage machine, nommée *réseau conceptuel*, pour encoder les expériences sensorielles d'un système informatique ; avec ses unités (*prémices, concepts, concepts structuraux, concepts homogènes, concepts hétérogènes*), ainsi que les mécanismes nécessaires à la génération et à l'exploitation des unités du réseau : mécanismes de *conceptualisation, genèse, activation* et *évocation*. Formé d'unités interconnectées, nommés *concepts*, le réseau conceptuel se construit automatiquement à partir de données sensorielles brutes selon une complexité (nombre d'unités et relations entre unités) qui tend à suivre celle des phénomènes sensoriels. Ce faisant, la structure du réseau ne modélise pas les relations entrées/sorties (contrairement aux approches classiques : réseaux de neurones, SVM, arbres de décision, réseaux bayésiens), mais plutôt les phénomènes observés en tant que tels. Le réseau conceptuel se veut ainsi encoder l'univers phénoménologique du système à partir duquel il est possible de développer plusieurs stratégies pour la reconnaissance de formes, la régression ou éventuellement le contrôle de systèmes robotiques.

Par exemple pour l'objectif de prédiction, un concept-étiquette pour chaque classe peut être ajouté aux entrées sensorielles du réseau. Lors d'une nouvelle expérience, où les concepts qui encodent la structure des observations s'activent plus en particulier, il est possible de prédire la classe en comparant les énergies des différents concepts-étiquettes. Partant de ce principe, nous présentons quelques exemples préliminaires de réseaux conceptuels utilisés pour la reconnaissance de chiffres (écrits et parlés) et pour la reconnaissance du locuteur.

L'avantage du paradigme proposé, par rapport au paradigme traditionnel entré/sortie, est premièrement que le modèle ainsi réalisé peut servir à des tâches quelconques sans nécessiter d'adaptation importante de la structure et des paramètres du réseau, n'étant pas optimisés pour des tâches spécifiques. En second lieu, la construction automatique du réseau permet d'atteindre une complexité structurelle qui se limite à celle des phénomènes observés : avec, potentiellement, des millions d'unités et milliards de connexions. En opposition, encoder des expériences phénoménologiques de complexité inconnue (et potentiellement dynamique) s'avérerait plus difficile avec des modèles dont la structure est fixe ; si tel était l'objectif de ces modèles. Troisièmement, les réseaux conceptuels ont la particularité d'apprendre rapidement avec peu d'exemples. Pour les corpus de grande taille, la gestion du taux d'apprentissage (et la complexité inhérente de la structure) est un défi pour le moment. Les limites et les perspectives à venir y sont discutées.

**Mots-clés:** Réseaux conceptuels, apprentissage profond, reconnaissance de formes, conceptualisation, activation



# Introduction to Conceptual networks applied to automatic machine learning

Patrice BOUCHER

## ABSTRACT

This thesis introduces a machine learning model, named *conceptual network*, to encode the sensory experiences of a computer system; with the different variants of its units (*premises, concepts, structural concepts, homogeneous concepts, heterogeneous concepts*), as well as the mechanisms necessary for the generation and exploitation of the network units : *conceptualization, genesis, activation* and *evocation* mechanisms. Formed from interconnected units, called concepts, the conceptual network is automatically constructed from raw sensory data according to a complexity (number of units and relations between units) that tends to follow that of sensory phenomena. In doing so, the structure of the network does not model the input/output relations (contrary to classical approaches : neural networks, SVM, decision trees, Bayesian networks), but rather the observed phenomena as such. The conceptual network is thus intended to encode the phenomenological universe of the system from which it is possible to develop several strategies for pattern recognition, regression or possibly the control of robotic systems.

For example, for the prediction objective, a concept-label for each class can be added to the sensory inputs of the network. In a new experiment, where the concepts that encode the structure of the observations are activated more in particular, it is possible to predict the class by comparing the energies of the different concept-labels. Based on this principle, we present some preliminary examples of conceptual networks used for digit recognition (written and spoken) and for speaker recognition.

The advantage of the proposed paradigm, compared to the traditional input/output paradigm, is firstly that the model thus realized can be used for any task without requiring significant adaptation of the structure and parameters of the network, not being optimized for specific tasks. Second, the automatic construction of the network allows to reach a structural complexity that is limited to that of the observed phenomena : with, potentially, millions of units and billions of connections. In contrast, encoding phenomenological experiences of unknown (and potentially dynamic) complexity would be more difficult with models whose structure is fixed ; if that were the objective of these models. Third, conceptual networks have the particularity of learning quickly with few examples. For large corpora, managing the learning rate (and the inherent complexity of the structure) is a challenge at the moment. Limitations and future perspectives are discussed.

**Keywords:** Conceptual Networks, deep learning, pattern recognition, conceptualization, activation



## TABLE DES MATIÈRES

	Page
INTRODUCTION .....	1
CHAPITRE 1 SYSTÈMES D'APPRENTISSAGE MACHINE .....	7
1.1 Qu'est-ce qu'un système d'apprentissage machine ? .....	7
1.2 Type d'apprentissage .....	9
1.3 Vecteur d'entrées et observations .....	10
1.4 Analyse en composantes principales .....	11
1.5 Modèles d'apprentissage .....	12
1.5.1 Les k plus proches voisins .....	12
1.5.2 Arbres de décision et forêts aléatoires .....	12
1.5.3 Modèle bayésien .....	14
1.5.4 Machine à vecteurs de support .....	15
1.5.5 Réseaux de neurones .....	16
1.6 Modélisation séquentielle/temporelle .....	17
1.6.1 Réseaux de neurones LSTM .....	18
1.7 Réseaux conceptuels .....	20
1.8 Tenter un nouveau paradigme .....	22
CHAPITRE 2 RELATIONS ENTRE LES CONCEPTS .....	25
2.1 Représentation/Abstraction .....	25
2.2 Composant/Agrégat .....	26
CHAPITRE 3 NATURE DES CONCEPTS .....	29
3.1 Prémice .....	29
3.2 Concepts homogènes versus hétérogènes .....	31
3.3 Concept structurel .....	31
3.4 Relation d'abstraction/représentation des concepts secondaires .....	33
3.5 Plasticité et rigidité .....	33
3.6 Concepts et neurones .....	34
CHAPITRE 4 CONCEPTUALISATION .....	37
4.1 Transpositifs .....	37
4.1.1 Structure du transpositif et utilité .....	37
4.1.2 Information mémorisée en chaque noeud du transpositif .....	38
4.1.3 Évaluation du concept nodal .....	39
4.2 Mécanismes de conceptualisation .....	41
4.2.1 Initialisation .....	42
4.2.2 Processus de détection .....	42
4.3 Genèse conceptuelle .....	43
4.3.1 Grouper les observations .....	45

4.3.2	Évaluer une composition ordonnée/désordonnée admettant un groupe d'observations .....	48
4.3.3	Générer une ramification .....	49
4.4	Conceptualisation, en bref .....	50
CHAPITRE 5 ACTIVATION ET RECONNAISSANCE DE FORMES .....		51
5.1	Activation .....	51
5.1.1	Consolidation des concepts actifs .....	53
5.1.2	Évocation .....	53
5.2	Reconnaissance de formes .....	56
5.3	L'activation, en bref .....	58
CHAPITRE 6 PREUVE DE CONCEPT .....		61
6.1	Reconnaissance d'images : chiffres écrits .....	62
6.1.1	Entraînement .....	63
6.1.2	Prédiction .....	64
6.1.3	Tests sur des corpus maison .....	65
6.1.3.1	Motifs translétés .....	66
6.1.3.2	Reconnaissance des chiffres 0, 1, 2, 3 sur 7x7 pixels .....	68
6.1.3.3	Chiffres 0, 8, 1, 7 sur 10x10 pixels .....	69
6.1.3.4	Reconnaissance des chiffres 0, 8, 1, 7 sur 14x14 pixels .....	69
6.1.4	MNIST .....	69
6.2	Reconnaissance de la parole : chiffres prononcés .....	71
6.2.1	Reconnaissance de la parole dépendante du locuteur .....	71
6.2.2	Reconnaissance de la parole indépendante du locuteur .....	75
6.2.3	Reconnaissance de la parole : perspectives .....	76
6.3	Reconnaissance du locuteur .....	77
CONCLUSION ET RECOMMANDATIONS .....		79
7.1	Contributions principales .....	79
7.2	Comparaison avec les autres approches d'apprentissage machine .....	79
7.3	Perspectives .....	81
ANNEXE I	ÉVALUATION DE L'ABSTRACTION COMMUNE À PLUSIEURS CONCEPTS .....	85
ANNEXE II	IDENTIFICATION DES ABSTRACTIONS ACTIVES .....	87
BIBLIOGRAPHIE .....		89



## LISTE DES FIGURES

	Page
Figure 0.1	Représentation conceptuelle d'un chat général et d'un chat « Minex » ..... 4
Figure 1.1	Un système d'apprentissage machine, typiquement composé d'entrées, d'un modèle et de sorties ..... 8
Figure 1.2	Modèle d'apprentissage ayant à son entrée les composantes principales du vecteur de caractéristiques ..... 11
Figure 1.3	Exemple d'arbre de décision pour sélectionner l'espèce d'un félin adulte ..... 13
Figure 1.4	Aperçu d'une machine à vecteurs de support ..... 15
Figure 1.5	Fonction noyau sur des données non linéairement séparables ..... 15
Figure 1.6	Vu d'un perceptron ..... 16
Figure 1.7	Modélisation temporelle avec un système d'apprentissage sans mémoire ..... 18
Figure 1.8	Unité LSTM ..... 20
Figure 1.9	Vue générale d'un réseau conceptuel : les unités sont les concepts, liés entre eux par des relations sémantiques ..... 21
Figure 2.1	Exemple de conditions lors de l'ajout d'une abstraction ..... 27
Figure 2.2	Représentation conceptuelle d'un chat général et d'un chat « Minex » ..... 28
Figure 3.1	Hiérarchie d'abstraction pour une intensité lumineuse de 0 à 1 ..... 30
Figure 3.2	Exemple de quatre concepts ordonnés distinctifs représentant un seul concept désordonné ..... 31
Figure 3.3	Mémoriser les concepts dans un arbre de façon à trouver facilement le concept correspondant à une composition observée ..... 33
Figure 3.4	Relations entre plusieurs concepts interconnectés ..... 35

Figure 4.1	Région de l'espace d'entrée représentée par chaque prémice nodale, exemple .....	39
Figure 4.2	Exemple d'un transpositif dont les noeuds forment un arbre entrelacé, utilisé pour une image de 4 par 4 de niveau de gris .....	40
Figure 4.3	Liens entre les prémices et les zones de l'image, puis des noeuds d'une profondeur avec ceux de la profondeur inférieure - pour le transpositif de la figure 4.2 .....	40
Figure 4.4	Évaluation du concept nodal dans un transpositif .....	41
Figure 4.5	Processus de conceptualisation itératif. La profondeur de départ augmente à chaque itération, jusqu'à une profondeur maximale correspondant, tout au plus, à celle du transpositif .....	42
Figure 4.6	(A) Exemple de génération systématique de concepts structuraux dans un transpositif hypothétique. Les concepts réalisent une représentation exacte. (B) Exemple concepts structuraux possibles dans une troisième itération de conceptualisation .....	44
Figure 4.7	Abstraction figurée d'un chat (à gauche) et d'un cheval (à droite) .....	46
Figure 4.8	Troisième et quatrième itérations d'une conceptualisation hypothétique .....	47
Figure 5.1	Deux yeux détectés lors des expériences A, B et C, puis lors d'une expérience D .....	51
Figure 5.2	Consolidation d'un concept $\psi_x$ ayant pour composants les concepts $\chi_1$ et $\chi_2$ . L'énergie du concept $\psi_x$ enforcit les liens d'agrégation de $\chi_1$ vers $\psi_x$ et de $\chi_2$ vers $\psi_x$ .....	54
Figure 6.1	Hierarchie d'abstraction de prémices représentant des chiffres .....	62
Figure 6.2	Corpus maison utilisé pour la reconnaissance des chiffres écrits. A : Premier corpus de test : 18 images de 5x5 pixels représentant trois symboles à différentes positions. B : Chiffres 0, 1, 2 et 3 sur 32 images de 7x7 pixels. C : Chiffres 0, 1, 7 et 8 sur 48 images de 10x10 pixels. D. Les nombres 0, 1, 7 et 8 sur 32 images de 14x14 pixels .....	66
Figure 6.3	Aperçu du réseau conceptuel généré par l'entraînement sur un corpus. A : Corpus 6.2A. B : Corpus D (avec 4949 concepts). C/D : Agrandissement 2x/4x du corpus D .....	67

Figure 6.4	A : Taux de détection correcte en fonction de la profondeur maximale de conceptualisation, pour différentes tailles de corpus. B : Nombre de concepts générés en fonction du nombre de données utilisées du corpus MNIST et de la profondeur maximale de conceptualisation. C : Taux de bonnes détections en fonction du nombre de concepts générés .....	71
Figure 6.5	Extractions des MFCCs .....	72



## LISTE DES SYMBOLES ET UNITÉS DE MESURE

$C$	Désigne un concept.
$A$	Désigne une abstraction.
$R$	Désigne une représentation.
$\chi$	Désigne une composition.
$\psi$	Désigne un concept agrégat.
$\Upsilon^{A \Omega}$	Désigne une ramification d'un concept $A$ selon un ensemble d'observations $\Omega$ .
$o$	Signifie <i>ordonné</i> .
$\rho$	Signifie <i>désordonné</i> .
$\upsilon$	Désigne le type.
$\Gamma^\upsilon$	Transpositif d'entrée pour le type $\upsilon$ .
$\phi^{p,n}$	$n^e$ noeud de profondeur $p$ .
$\xi_{p+1,m}^{p,n}$	$m^{\text{ème}}$ enfant du noeud $\phi^{p,n}$ .
$c_{p+1,m}^{p,n}$	Concept lié au $m^{\text{ème}}$ enfant du noeud $\phi^{p,n}$ .
$s_{p+1,m}^{p,n}$	Prémice synthèse liée au $m^{\text{ème}}$ enfant du noeud $\phi^{p,n}$ .
$N^x$	Nombre d'éléments $x$ .
$s^{p,n}$	Prémice synthèse des prémices des feuilles du noeud $\phi^{p,n}$ .
$c^{p,n}$	Concept ayant pour composants les concepts liés aux enfants du noeud $\phi^{p,n}$ .



## LEXIQUE

**Abstraction** Concept qui généralise les caractéristiques communes de différents concepts. Ces derniers sont dits des *représentants* de l'abstraction.

**Abstraction explicite de X** Abstraction directement référencée par un concept représentant X.

**Abstraction implicite de X** Abstraction référencée par X ou une abstraction de X.

**Activation** L'*activation* inclut le (1) le processus de détection de concepts à la conceptualisation et (2) le processus d'évocation de concepts suivant la conceptualisation.

**Admettre** Si A *admet* B, A correspond à B ou à une abstraction de B.

**Agrégat** Unité qui réunit plusieurs concepts encodant, typiquement, de l'information complémentaire d'un même type d'expériences

**Agrégat explicite de X** Concept-agrégat directement référencé dans la liste d'agrégats de X.

**Agrégat implicite de X** Concept-agrégat référencé (à titre d'agrégat) par X, un agrégat de X, ou une abstraction de X.

**Agrégat nodal** Concept-agrégat d'un noeud du transpositif d'entrée, inféré lors du processus de conceptualisation.

**Arborescence structurelle** La structure de la composition d'un concept jusqu'aux éléments les plus élémentaires : les prémices.

**Composants** Concepts qui, mis ensemble, définissent un concept agrégat.

**Composition** Se caractérise par la nature des composants et leur ordonnance.

**Concept** Représentation d'une récurrence perceptuelle caractéristique.

**Concept hétérogène** Concept dont les feuilles de l'arborescence sont des prémices liées à différents types de phénomènes.

**Concept homogène** Concept dont les feuilles de l'arborescence sont des prémices liées à un seul type de phénomène.

**Concept ordonné/désordonné** Concept secondaire définit/non définit par l'ordre de ses composants.

**Concept primaire** Concept représentant l'état d'un phénomène perçu à une entrée du système.

**Concept structural nodal** Concept mémorisé à un noeud d'un transpositif d'entrées, résultant du processus de conceptualisation. Il décrit la structure d'une composition.

**Concept secondaire** Concept définit par une agrégation d'autres concepts.

**Conceptualisation** Mécanisme responsable de la détection et de la création des concepts. La conceptualisation peut être *générative* (détecte, crée des concepts et consolide des liens) ou *non générative* (détecte des concepts sans en créer de nouveau et sans modifier la structure du réseau).

**Consolidation** *d'un concept*. L'action d'enforcer le lien d'agrégation de chaque composant envers le concept.

**Énergie** Désigne l'intensité avec laquelle un concept caractérise une expérience.

**Expérience** Évènement qui définit l'état des entrées du système et leur évolution.

**Évocation** Processus de transfert d'énergie des concepts détectés vers leurs abstractions et agrégats.

**Genèse** Création d'un concept à partir d'observations.

**Observation** Groupe de composants liés à un élément structurel de l'expérience.

**Phénomène** Évènements physiques ou abstraits décrits par une dynamique et une structure spécifique. Il s'observe à partir d'outils de captation spécialisés (caméra, micro, autres capteurs).

**Plasticité** Capacité d'un concept à admettre différentes formes (représentants).

**Prémice** Un concept lié à une plage de valeurs sensorielles décrivant l'état d'un phénomène.

**Prémice nodale** Prémice à un noeud d'un transpositif d'entrées représentant une région de l'espace d'entrée.

**Profondeur (d'un concept)** Celle de son arborescence structurelle.

**Ramification** Génération d'un concept, à partir d'un concept souche, qui augmente la rigidité de représentation d'un ensemble d'observations (ce nouveau concept sera un *représentant* du concept souche et aura une plus grande profondeur).

**Représentant (Représentation) d'un concept X** Encode un sous-type d'expériences de celles encodées par X. Le représentant étant plus spécifique, son arborescence structurelle est potentiellement plus profonde et/ou les prémices à ses feuilles plus spécifiques.

**Réseau conceptuel** Ensemble de concepts interreliés.

**Rigidité** Niveau de spécificité avec lequel un concept encode un certain type d'expériences. Les concepts rigides permettent de différencier des types d'expériences similaires.

**Substitut** Abstraction implicite utilisée dans un noeud du transpositif d'entrées lors de la conceptualisation. L'utilisation d'un *substitut* survient lorsque la composition à ce noeud n'est liée à aucun concept existant pour le moment.

**Synthèse** Il s'agit de l'abstraction commune à plusieurs concepts.

**Transpositif** Décrit la structure morphologique d'un ensemble d'entrées liées à un type de phénomène sensoriel ou abstrait.

**Type** Le *type* de phénomène, décrivant de façon unique sa nature physique ou abstraite.



## INTRODUCTION

Que ce soit pour automatiser le travail humain, pour créer des outils, des jeux, des jouets ; ou encore par simple sens du défi, la prochaine ère technologique sera très plausiblement marquée par des machines dotées d'une capacité à « raisonner » inspirée de celle de l'homme. Or, ceci exige que la machine soit dotée d'une forme d'*intelligence*, qui resterait « artificielle » puisque créée par l'homme. Une question s'impose : « comment créer cette intelligence ? »

Le dictionnaire Larousse (2023) définit l'intelligence (réelle) comme étant l'« aptitude d'un être humain à s'adapter à une situation, à choisir des moyens d'action en fonction des circonstances ». L'intelligence impose donc une capacité à résoudre une situation ambiguë afin d'atteindre un objectif précis. Cette situation est perçue par nos sens biologiques. Pour un système informatique, elle sera perçue par des capteurs (caméra, micro, capteurs tactiles, thermomètres), des entrées informatiques (boutons, souris, clavier) ou via des systèmes tiers.

Ensuite, l'intelligence s'intéresse à réaliser une tâche, par exemple : effectuer une conclusion sur la nature de l'information perçue (classification de l'information) ; mesurer l'état d'un phénomène observé (l'ampleur du stress à partir de données physiologiques) ; commander les mouvements d'une plateforme mobile, etc. Il y aura donc une forme de transposition, effectuée par un modèle d'apprentissage machine, de l'information perçue en entrée vers le travail nécessaire à la tâche en sortie. Cette vision entrée/sortie de l'intelligence définit le paradigme de conception actuel des modèles d'apprentissage machine. En optimisant les paramètres d'un modèle de façon à minimiser l'erreur de prédiction dans un corpus de données annotées, il est possible d'obtenir un modèle compact ayant une bonne capacité de généralisation. Avec ce paradigme où la structure du modèle est fonction de son utilisation, il est, cependant, généralement difficile de réutiliser le modèle à des fins variées : c'est-à-dire que l'optimisation des paramètres pour un objectif compromet les performances des objectifs voisins. Avec ce paradigme, l'éla-

laboration d'un modèle complexe est réalisée manuellement et nécessite beaucoup de données pour entraîner les nombreux paramètres.

Dans cette thèse, nous avons pour objectif de tenter un paradigme de conception différent où le modèle vise plutôt à encoder une représentation logique et généraliste des phénomènes perçus. Sous ce paradigme, l'information est incorporée au modèle selon sa nature et sa structure logique, sans égard à son utilité pratique, et il n'y a donc pas de sortie proprement dite.

Construit de cette façon, le modèle pourrait servir à des fins variées sans compromettre les performances de chaque aspect. C'est ainsi que nous réalisons au chapitre 6, avec un même modèle, à la fois la reconnaissance de chiffre prononcé et la reconnaissance du locuteur.

Ce modèle, construit de façon automatique de manière à encoder la structure des expériences en entrée, peut également suivre la complexité de ces expériences sans égard aux annotations. Il s'agit d'un avantage important pour les applications où les annotations sont sporadiques.

Après de nombreuses recherches s'étalant sur quelques années, je propose sous ce paradigme de conception « les réseaux conceptuels ».

Tout d'abord, qu'entendons-nous par « concept » ? Le dictionnaire Larousse (2020) définit un concept comme étant une « idée générale et abstraite que se fait l'esprit humain d'un objet de pensée concret ou abstrait, et qui lui permet de rattacher à ce même objet les diverses perceptions qu'il en a, et d'en organiser les connaissances. »

Pour le Robert (2021), un concept se définit comme étant une *idée générale; représentation abstraite d'un objet ou d'un ensemble d'objets ayant des caractères communs*.

Dans le domaine de l'intelligence artificielle, nous pouvons considérer, en d'autres termes, le concept comme étant une *représentation logique d'une récurrence perceptuelle caractéristique*. Ainsi, un concept est un élément perceptuel en soi qui peut être défini par ses relations

avec d'autres concepts : le type de ces relations, leur ordonnance et l'identité des concepts liés. L'ensemble des concepts et leurs relations forme un *réseau conceptuel*.

*Conceptualiser* une expérience sensorielle consiste à la représenter par un ensemble d'unités caractéristiques : les concepts. Par exemple, si j'observe plusieurs images d'un même chat nommé « Minex », je peux constater qu'un ensemble d'éléments perceptuels se répètent. Ils caractériseront une tête, des yeux, un nez, une patte, etc. La récurrence perceptuelle observée pour chaque *composant* du chat (tête, yeux, nez, patte) donne lieu à des concepts spécifiques pour chacun d'eux. Si le concept « Minex » est défini par un *agrégat* structuré d'éléments perceptuels récurrents de Minex, ces éléments incluront de facto les composants « Tête de Minex », « Patte de Minex »... où la tête inclura à son tour « Yeux de Minex », « nez de Minex », comme l'illustre la figure 0.1.

La conceptualisation est un processus qui généralise la structure des données observées en des concepts, qui y sont à la fois détectés et possiblement générés (on parlera alors de conceptualisation générative). Supposons que le concept « Chat » est inféré de plusieurs exemples de chats, dont le chat Minex. Le concept « Minex » est alors dit *représentant* du concept « Chat » ; puis le concept *Chat* est dit une *abstraction* du concept « Minex ». Nous disons également que « Chat » *admet* « Minex ». L'observation d'un concept implique l'observation de ses abstractions : si j'observe « Minex », j'observe un « Chat ».

Contrairement à une catégorie/classe spécifique, le concept n'a pas de frontière fixe. Comme il relie diverses perceptions, sa composition dépend des expériences perceptuelles propres à chaque observateur. Il est donc irréaliste de rattacher directement un concept à une catégorie/classe spécifique telle que définie par exemple via les mots d'une langue, comme réalisé plus haut avec le « Chat ». Il est en effet fort possible qu'aucun concept n'abstrait la totalité des expériences perceptuelles liées aux chats. Remarquez que je rattache, malgré cela, des concepts à des classes spécifiques dans les explications afin de produire des exemples intuitifs.

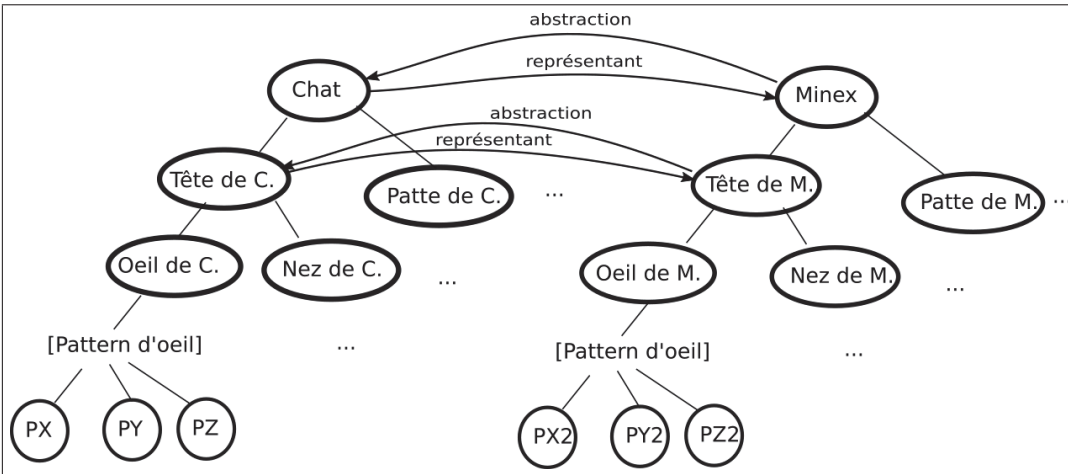


Figure 0.1 Représentation conceptuelle d'un chat général et d'un chat « Minex »

Lors d'une expérience, des données sont observées. La structure de ces données est décrite par les concepts sélectionnés lors de la conceptualisation. Ces concepts seront dits *actifs* au sens qu'ils décrivent des éléments de la structure de l'expérience courante avec une certaine ampleur, ou *énergie*. Lorsqu'un concept est actif, ses abstractions le sont implicitement. Par conséquent, si Minex est actif, Chat est actif. Ensuite, les concepts actifs peuvent *évoquer* les agrégats qui les contiennent. Si un poil de Minex est observé, un concept composé d'un ou plusieurs poils (tel un chat) sera *évoqué* avec une énergie d'autant plus grande que ce concept fut conjointement observé avec un poil similaire. Le terme *évocation* fait référence au mécanisme de transfert d'*énergie* des concepts actifs aux abstractions et agrégats (chapitre 5). Le terme *activation* désigne à la fois la détection des concepts qui encodent l'expérience, puis l'évocation de leurs énergies vers d'autres concepts.

Dans un réseau conceptuel, l'association d'un concept à une catégorie/classe impose que ses éléments perceptuels caractéristiques soient expérimentés avec ceux de la catégorie. Dans cette optique, la catégorie est également considérée en tant qu'expérience perceptuelle. Par exemple, la conceptualisation d'un ensemble d'expériences acoustiques d'un même mot générera des concepts mettant en relief les structures acoustiques récurrentes du mot. Maintenant, si le mot

« chat » est expérimenté avec des expériences visuelles de chats, la conceptualisation générera des concepts qui agrègent les éléments perceptuels récurrents des images de chats à ceux des expériences acoustiques du mot « chat ». Dans un tel système, la catégorisation d'une expérience visuelle (la visualisation d'un nouveau chat) en une étiquette acoustique (le mot « chat ») signifie que la nouvelle expérience visuelle évoque (via le mécanisme d'activation) davantage l'étiquette acoustique « chat » que l'étiquette acoustique « chien ». La classification est ainsi réalisée en analysant les énergies relatives des différents concepts évoqués.

Cette thèse contribue à l'état de la science avec :

1. l'introduction des *réseaux conceptuels*, un nouveau modèle pour l'apprentissage machine permettant d'encoder les expériences phénoménologiques d'un système informatique ; ce qui inclut la définition des concepts, leurs propriétés et leurs relations.
2. l'introduction des algorithmes de *conceptualisation* et de *genèse conceptuelle* permettant la construction des réseaux conceptuels ; d'un algorithme d'*activation* pour l'activation des unités qui décrivent l'expérience courante ; et d'un algorithme préliminaire de classification basé sur les unités actives.
3. la présentation des propriétés, des avantages et des limites des réseaux conceptuels pour des applications pratiques ; via en particulier des preuves de concept dans les applications de reconnaissance d'images, reconnaissance de la parole et identification de locuteurs.

Avant d'entrer en la matière, je présente d'abord une brève revue de l'apprentissage machine au chapitre 1, dans laquelle les réseaux conceptuels seront finalement situés. Les chapitres suivants se concentrent sur les réseaux conceptuels. Les types de relations entre les concepts sont présentés au chapitre 2, puis les types de concepts au chapitre 3. Ensuite, le chapitre 4 décrit les mécanismes de conceptualisation et de genèse conceptuelle, puis le chapitre 5 le mécanisme d'activation. Une preuve de concept est réalisée au chapitre 6 pour les applications de reconnaissance de chiffres écrits et prononcés, puis pour la reconnaissance du locuteur.

Ces exemples d'applications mettent en lumière les propriétés, les avantages et les limites des réseaux conceptuels. Les limites et perspectives de l'approche sont finalement résumées à la conclusion.

## CHAPITRE 1

### SYSTÈMES D'APPRENTISSAGE MACHINE

#### 1.1 Qu'est-ce qu'un système d'apprentissage machine ?

Dans les temps anciens, l'homme utilisait des « outils » qu'il manipulait directement selon ses mouvements et volontés spontanés ; autant dans les domaines de la chasse, de la guerre, de la cuisine ou du transport. Ensuite fut l'idée de créer des outils de mécanisation plus sophistiqués, dits « machines », qui ont la capacité de perpétuer leurs mouvements propres, tels que des catapultes, des horloges, des trains ou encore des automobiles. Ces machines sont purement entraînées par des mécanismes physiques pouvant être guidés par une volonté humaine, et opèrent pour l'instant sans égard à leurs environnements.

Et si la machine pouvait, maintenant, amorcer ses propres actions en fonction de son contexte d'opération ? Nous aurions ainsi, à la limite, une automobile pouvant naviguer sur la route sans pilote, un train pouvant automatiquement arrêter et redémarrer à chaque station, des missiles pouvant se piloter pour atteindre leurs cibles, etc. Dans tous ces exemples, la machine doit percevoir une certaine information sur laquelle se base la prise de décision ; au même titre que la visualisation de l'état de la route oriente notre conduite automobile. L'information nécessaire à la prise de décision est considérée comme une « entrée » de la machine ; tandis que la décision d'action est considérée comme une « sortie ».

Dans le domaine informatique, nous avons également des machines générant une information abstraite (telle que la classe d'un objet sur une image, le mot énoncé sur un enregistrement audio, le niveau de stress lié à des données physiologiques) à partir d'une information plus primaire (l'image, le fichier audio, les données physiologiques). Dans ce cas, la sortie est l'étiquette de la classe (classification) ou encore une valeur numérique qui caractérise l'état des entrées (régression).

La figure 1.1 présente une telle machine « intelligente », composée d'entrées telles que des sons, des images, des boutons, des vitesses mesurées ; d'un modèle et de sorties telles que des actions, des classes ou des valeurs numériques.

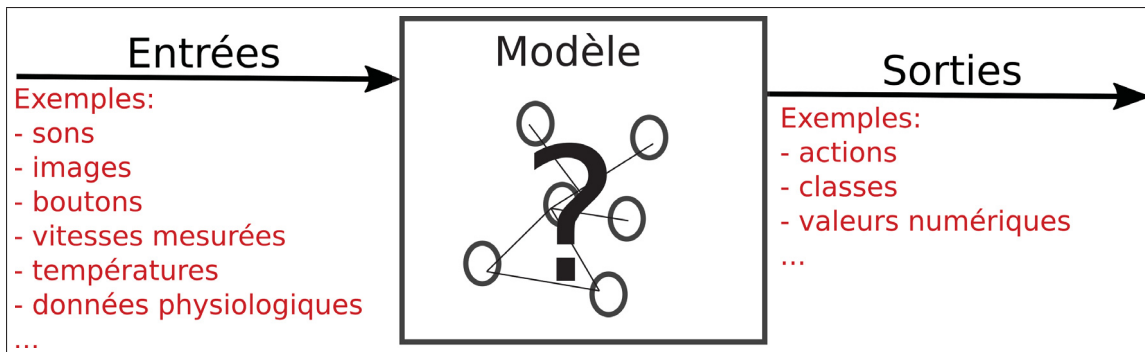


Figure 1.1 Un système d'apprentissage machine, typiquement composé d'entrées, d'un modèle et de sorties

Bien que l'idée d'avoir des machines autonomes est séduisante pour limiter l'effort de supervision humaine ; leur réalisation se présente plus complexe puisqu'il est, en effet, impossible de définir manuellement un comportement-machine adéquat pour chaque état observable des entrées. Prenons par exemple le contexte d'une automobile intelligente qui utilise une caméra de 1024x1024 pixels, où chaque pixel représente une couleur rouge/vert/bleu de 256 valeurs pour chaque ton. Nous avons donc  $16.8M$  ( $256^3$ ) couleurs différenciables pour chaque pixel, puis  $16.8M^{1024^2}$  états possibles des entrées. Si ensuite on considère la vitesse courante de l'automobile numérisée selon un pas de 1km/h de 0 à 120km/h, nous aurons plutôt  $121 \times 16.8M^{1024^2}$  combinaisons d'entrées possibles - l'ajout de chaque entrée multiplie le nombre d'états possibles. Il va de soi qu'aucun concepteur ne sera en mesure : ni de déterminer quelle est l'action/décision adéquate pour chaque combinaison d'entrée ; ni de préprogrammer autant de comportements dans une même machine.

Plutôt, la machine doit « apprendre » automatiquement à synthétiser les états observables des entrées de façon à générer une action/décision adéquate. Cet apprentissage repose sur quelques exigences :



1. la machine doit avoir un « modèle » qui synthétise les données observables pour permettre la génération d'une action/décision adéquate.
2. le modèle doit apprendre (c.-à-d. se construire ou se paramétrer) de façon automatique à partir d'exemples d'expériences types (états, ou suite d'états, particuliers des entrées).
3. une banque d'expériences types, réunie dans un « corpus de données », doit être utilisée pour entraîner le système. Ce corpus, ayant une grande incidence sur la structure du modèle inféré, doit être retraçable.

## 1.2 Type d'apprentissage

Au fur de l'apprentissage du modèle, son état est continuellement altéré de manière à synthétiser de mieux en mieux les expériences d'entrées.

Ce travail est parfois possible qu'en se basant sur les expériences en entrées. Nous parlerons alors d'« apprentissage non supervisé » au sens qu'on a pas besoin de superviser les valeurs de sorties, qui ne sont en effet pas considérées. Il s'agit d'un apprentissage pratique pour enrichir le modèle via des expériences dont les sorties sont inconnues ou trop abstraites (pour permettre d'entraîner convenablement un modèle complexe).

Si, plutôt, le modèle doit se structurer de façon à générer des sorties spécifiques, ces dernières sont considérées lors d'un apprentissage dit « supervisé ». Ce type d'apprentissage repose sur une banque d'expériences dont les valeurs de sorties sont bien annotées.

Un apprentissage « semi-supervisé » peut aussi être réalisé lorsque les expériences sont partiellement annotées, c'est-à-dire que nous réaliserons un apprentissage non supervisé avec toutes les expériences et un apprentissage supervisé avec les expériences annotées. Parfois, nous effectuerons un apprentissage non supervisé sur une section du modèle affinée aux entrées, puis un apprentissage supervisé considérant le modèle dans son ensemble.

Pour les systèmes robotiques où la sortie est une action, l'« apprentissage par renforcement » consiste à bonifier la génération d'actions menant le robot dans un état adéquat selon ses objectifs, et à pénaliser les autres actions.

### 1.3 Vecteur d'entrées et observations

Considérons un vecteur d'entrées comportant  $N$  caractéristiques, soit  $\vec{x} = \{x_1, \dots, x_N\}$ . À chaque instant où la machine est utilisée, ces caractéristiques prennent des valeurs particulières (continues ou discrètes) qui décrivent le contexte d'opération à cet instant. Nous pouvons ensuite accumuler ces observations afin d'obtenir un portrait des combinaisons de caractéristiques probables dans le contexte d'opération de la machine.

Pour donner un exemple simple, nous pourrions avoir une caractéristique  $x_1$  qui décrit le rythme cardiaque {lent, moyen, rapide}, une variable  $x_2$  qui décrit la variabilité cardiaque {faible, moyenne, élevée} et une variable  $x_3$  qui décrit la transpiration de la peau {faible, moyenne, élevée}. L'espace de caractéristiques comprend ainsi 27 états (soit  $3^3$ ), combinaisons possibles des trois entrées. Cependant, nous nous intéressons plus particulièrement aux états observés. Nous observons chez un participant qu'à un instant : son rythme cardiaque est lent, sa variabilité cardiaque est moyenne et sa transpiration est faible. Ce cas correspond à l'observation 1  $\vec{x}^{(1)} = \{x_1 = \text{lent}, x_2 = \text{moyenne}, x_3 = \text{faible}\}$ . À un autre instant, son rythme cardiaque est élevé, sa variabilité cardiaque est élevée et sa transpiration est élevée : ce qui correspond à l'observation  $\vec{x}^{(2)} = \{x_1 = \text{élevé}, x_2 = \text{élevé}, x_3 = \text{élevé}\}$ . Puis, nous continuons à accumuler des observations jusqu'à un nombre limité d'observations  $M$ .

Dans le corpus d'entraînement, les observations sont couramment liées à une ou plusieurs valeurs de sortie que les modèles d'apprentissage devront éventuellement prédire. Avec une sortie  $y$  décrivant un état de stress {faible, modéré, élevé}, nous pourrions ainsi avoir la première observation  $\vec{x}^{(1)}$  avec un état de stress faible  $\vec{y}^{(1)} = \{\text{faible}\}$  et la deuxième  $\vec{x}^{(2)}$  avec un stress modéré  $\vec{y}^{(2)} = \{\text{modéré}\}$ . Les données d'entraînement pour les  $M$  observations seront les tuples  $\{\vec{x}^{(1)}, \vec{y}^{(1)}\}, \{\vec{x}^{(2)}, \vec{y}^{(2)}\}, \dots, \{\vec{x}^{(M)}, \vec{y}^{(M)}\}$ .

## 1.4 Analyse en composantes principales

Dans l'exemple plus haut, si nous observons toujours  $x_1 = \text{élevé}$  avec  $x_3 = \text{élevé}$ , alors nous pouvons prédire que  $x_3$  sera probablement élevé si  $x_1$  est élevé. Le fait de pouvoir prédire la valeur d'une caractéristique à partir d'une autre caractéristique indique, en somme, que les valeurs des deux caractéristiques apportent à peu près la même information pour différencier les états du contexte. Par conséquent, le modèle fonctionnerait tout aussi bien en utilisant une seule des deux caractéristiques. En pratique, nous considérons plus spécifiquement la corrélation entre les caractéristiques d'entrée prises deux à deux. Plus les variables sont corrélées, plus l'information qu'elles apportent est, en quelque sorte, redondante.

L'objectif de l'analyse en composantes principales (Bro & Smilde, 2014) est de créer un second vecteur de caractéristiques  $\vec{x}'$  (à partir du vecteur d'entrées  $\vec{x}$ ) de dimension réduite où les caractéristiques sont davantage décorrélées afin de minimiser la perte d'information. Ces nouvelles caractéristiques sont nommées « composantes principales ». En diminuant le nombre de variables, l'analyse en composantes principales permet de limiter la complexité du modèle avec peu d'effet sur sa performance, qui est liée plus particulièrement à la richesse de l'information contenue à ses entrées. L'utilisation conséquente du modèle est illustrée à la figure 1.2.

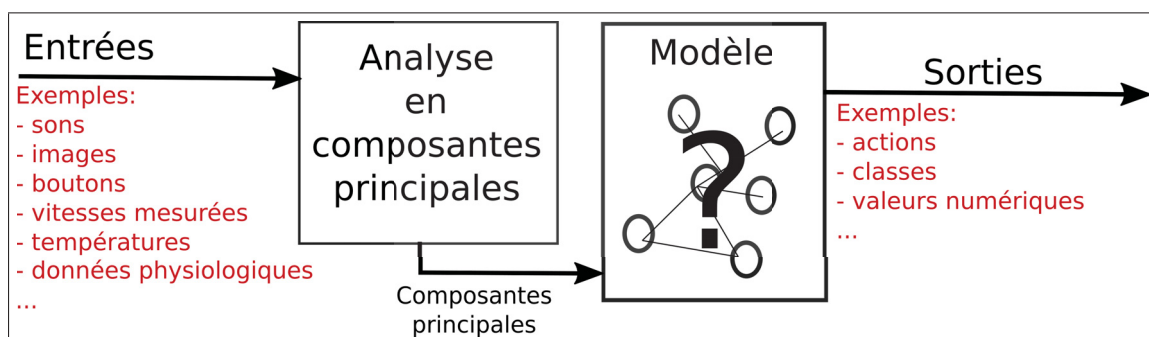


Figure 1.2 Modèle d'apprentissage ayant à son entrée les composantes principales du vecteur de caractéristiques

## 1.5 Modèles d'apprentissage

Comment synthétiser l'information d'entrées afin, éventuellement, de générer une sortie adéquate ? Il s'agit ici d'une question ouverte. Plusieurs modèles ont été suggérés. Chacun a une structure particulière pour représenter l'information, une logique spécifique qui sous-tend cette structure, une ou plusieurs techniques originales pour l'entraîner et une façon conséquente d'être utilisé. Nous présentons succinctement ci-dessous quelques modèles d'apprentissage incontournables, qui furent largement exploités dans les différentes sphères d'application de l'intelligence artificielle.

### 1.5.1 Les k plus proches voisins

Dans la technique des k-plus-proches-voisins (Zhang & Zhou, 2007), le modèle est directement formé des tuples du corpus d'apprentissage. En cours normal d'utilisation, la sortie est attribuée à celle du tuple dont le vecteur d'entrée du modèle est le plus près de l'observation : c'est-à-dire de « distance » minimale (par exemple, la distance euclidienne). Ce modèle est facile à implémenter. Un inconvénient important est qu'il ne modélise pas réellement de liens entre les différentes observations, ni davantage de liens entre les observations et les valeurs de sorties. Ce manque de synthèse le rend sensible aux erreurs d'annotations ; peu performant lorsque l'observation s'éloigne des tuples du corpus ; et d'une complexité qui augmente linéairement avec le nombre de données d'entraînement.

### 1.5.2 Arbres de décision et forêts aléatoires

Dans un arbre de décision, la sortie est sélectionnée en considérant une suite de conditions se reportant aux variables d'entrées. Chaque condition détermine le prochain ensemble de conditions jusqu'à la sélection (ou génération) de la sortie.

La figure 1.3 présente un exemple d'arbre de décision permettant de prédire un félin probable en fonction de ses caractéristiques. Si nous avons un félin de 255Kg, d'Asie et de fourrure

uniforme, l'arbre indiquerait qu'il s'agit d'un Lion d'Asie en considérant les critères de la racine jusqu'à la feuille, qui indique la conclusion.

Dans un système d'apprentissage comportant beaucoup de variables et de données, les conditions peuvent être inférées automatiquement selon différentes techniques d'optimisation (Barros *et al.*, 2012; Safavian & Landgrebe, 1991; Murthy, 1998). Dans ce processus d'optimisation, il importe de sélectionner près de la racine des conditions permettant de bien discriminer les classes des différents embranchements. Pour la classification des félins à la figure 1.3, ceci implique qu'il y aurait par exemple très peu (idéalement aucun) chats de plus de 6 kg, très peu de panthères au-dessus de 100 kg et très peu de Lion d'Asie en deçà de 100 kg. Plus la condition se rapproche de la feuille, moins drastique sera l'erreur sur la conclusion : ainsi pouvons nous considérer qu'il est plus grave de prendre un lion pour un chat qu'un lion pour un tigre.

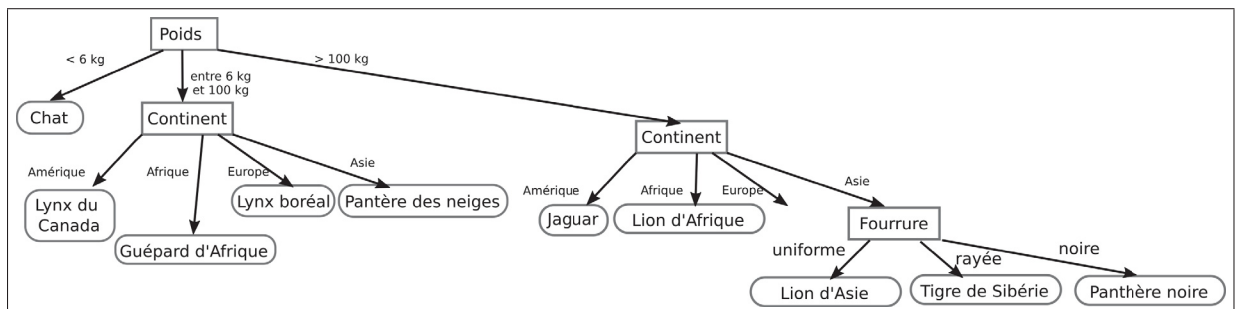


Figure 1.3 Exemple d'arbre de décision pour sélectionner l'espèce d'un félin adulte

Comme la conclusion d'un arbre de décision est sensible aux choix des conditions tout au long des multiples embranchements, et que la génération de ces conditions est sensible aux données d'entraînement, il est difficile de garantir que l'arbre qui sera inféré est optimal. Afin d'obtenir un modèle plus robuste, une stratégie consiste plutôt à construire plusieurs arbres en alternant, d'un arbre à l'autre, les données d'entraînement et/ou les techniques d'optimisation. Dans cette *forêt aléatoire* (Breiman, 2001), chaque arbre vote pour une sortie. La sortie du modèle est celle ayant obtenu le plus de votes.

### 1.5.3 Modèle bayésien

En classification, l'approche bayésienne (Robert, 2005) consiste à choisir la classe  $C^*$  la plus probable considérant les observations  $X$  du vecteur de caractéristiques. Soit  $W$  classes  $\{C^1, C^2, \dots, C^W\}$ , nous cherchons donc la classe  $C^*$  :

$$C^* = \arg \max_{c^w, w \in \{1, \dots, W\}} P(C^w | X)$$

Cependant il est plus facile, en pratique, de modéliser les densités de probabilités des observations conditionnelles à chaque classe  $p(X|C^w), w \in \{1, \dots, W\}$  et les densités de probabilités des classes  $p(C^w), w \in \{1, \dots, W\}$ . Ces dernières quantités sont donc utilisées pour évaluer les probabilités  $p(C^*|X)$  via le théorème de Bayes :

$$C^* = \arg \max_{c^w, w \in \{1, \dots, W\}} \frac{P(X|C^w)p(C^w)}{p(X)}$$

où la probabilité des observations  $p(X)$  n'a pas besoin d'être modélisée ni évaluée puisqu'elle ne dépend pas de la classe, et ne change donc pas le choix de la classe lors de la maximisation. Nous pouvons donc simplifier l'équation plus haut comme suit :

$$C^* = \arg \max_{c^w, w \in \{1, \dots, W\}} P(X|C^w)p(C^w) \quad (1.1)$$

Pour réaliser un modèle bayésien, on doit choisir une façon de modéliser les fonctions de densité de probabilités de l'équation 1.1. Pour la probabilité des observations conditionnelle à la classe, on choisira typiquement une fonction de densité de forme prédéfinie comportant des paramètres ajustables, tels que des fonctions gaussiennes, Gammas et des mixtures de gaussiennes (*GMM : Gaussian mixture models*) (Reynolds, 2009). L'entraînement consiste à évaluer les valeurs des paramètres de chaque fonction  $w$  qui maximisent la vraisemblance des données du corpus obtenues avec les tuples ayant pour sortie la classe  $C^w$ , à partir d'une technique adaptée à la fonction sélectionnée. Nous avons par exemple l'estimateur de vrai-

semblance maximale de Aldrich (1997) pour une fonction de densité gaussienne et de Thom (1958) pour une fonction Gamma, puis l'algorithme de maximisation de l'espérance (*EM* : *Expectation-Maximization*) pour une mixture de fonctions gaussiennes (Moon, 1996).

La probabilité des classes  $P(C^w), w \in \{1, \dots, W\}$  correspond à leurs occurrences relatives. Dans le contexte où le système sert à détecter des anomalies, une stratégie préférable<sup>3</sup> consiste à choisir la classe qui minimise le risque (Berger, 2013).

#### 1.5.4 Machine à vecteurs de support

L'idée de la *machine à vecteurs de support* (Schölkopf *et al.*, 2002) présentée à la figure 1.4 est de partitionner l'espace de caractéristiques en régions délimitées par des frontières (des hyperplans), où chaque région est attribuée à une classe spécifique. Les frontières sont construites de façon à maximiser la marge avec les données d'entraînement. Les données qui définissent la largeur des marges sont utilisées pour représenter les hyperplans et sont ainsi appelées les « vecteurs de support ».

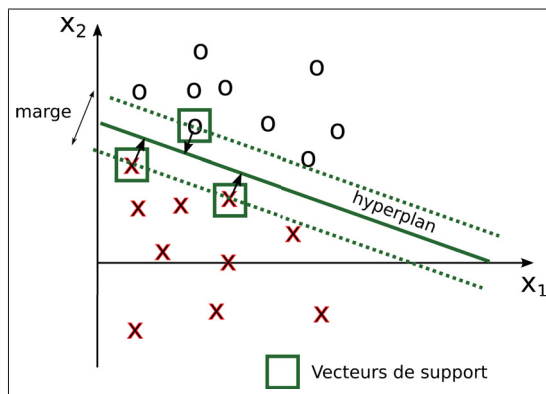


Figure 1.4 Aperçu d'une machine à vecteurs de support

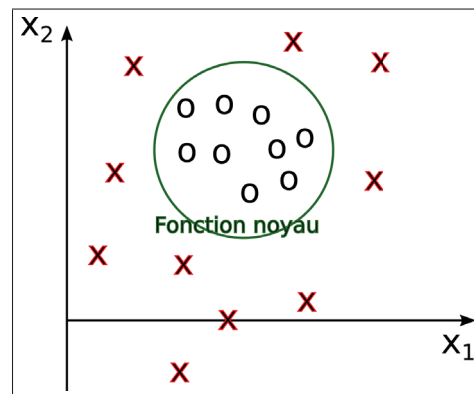


Figure 1.5 Fonction noyau sur des données non linéairement séparables

3. Par exemple, si le système sert à détecter une forme de cancer rare survenant une fois sur 1 million ( $p(C^w = cancer) = 10^{-6}$ ), la classe cancer ne sera probablement jamais sélectionnée même si elle explique très bien les observations (dans lequel cas  $P(X|C^w = cancer)$  est élevé). Le système passera donc à côté de son objectif.

Dans le cas où les données sont linéairement inséparables comme à la figure 1.5, il n'est pas possible de définir un hyperplan qui sépare les données. Dans cette circonstance, la stratégie est d'appliquer une fonction *noyau* qui transforme les données vers un espace où les données sont linéairement séparables tel qu'à la figure 1.4, soit l'*espace de représentation*. La stratégie expliquée plus haut peut dès lors être appliquée.

### 1.5.5 Réseaux de neurones

Inspiré du neurone humain par Rosenblatt (1958), le *perceptron* présenté à la figure 1.6) est une unité logique dont la sortie peut s'*activer* ou non selon des valeurs d'entrées.

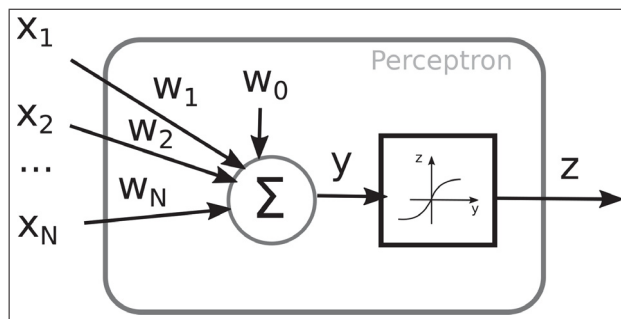


Figure 1.6 Vu d'un perceptron

Le *perceptron* réalise une somme pondérée des valeurs observées à ses entrées  $\vec{x} = \{x_1, x_2, \dots, x_N\}$  :

$$y = w_0 + \sum_{n=1}^N x_n w_n \quad (1.2)$$

où  $y$  est la sortie de la sommation,  $w_0$  un biais et  $w_n, n \in \{1, \dots, N\}$  les poids accordés pour chaque entrée.

La sortie  $y$  de la sommation peut ensuite être transformée par une *fonction d'activation* dérivable  $f(y)$ , incluant par exemple les fonctions sigmoïde, tangente hyperbolique, arc tangente, ReLu (Sharma *et al.*, 2017). Afin d'effectuer la classification, la sortie de la fonction tendra vers une constante parmi deux selon qu'une classe est détectée ou non. Nous pourrions ainsi avoir respectivement  $f(y) \rightarrow K_1$  et  $f(y) \rightarrow K_2$  (typiquement -1 ou 0 pour  $K_1$  et 1 pour  $K_2$ )



pour la non-détection, puis pour la détection d'une classe A ; ou encore pour la détection d'une classe A, puis pour la détection d'une classe B. Utilisé de cette façon, le perceptron agit en tant que classificateur linéaire où les données sont séparées par un hyperplan paramétré par les valeurs des poids.

Il est ensuite possible de combiner plusieurs perceptrons tel qu'à former un « réseau de perceptrons », ou « réseau de neurones », permettant de modéliser des relations hautement non linéaires entre des entrées et des sorties. Lors de l'*apprentissage* du réseau de neurones, les poids des entrées des neurones sont automatiquement corrigés, par exemple via la méthode de rétropropagation de l'erreur (Rumelhart *et al.*, 1986), de sorte à minimiser l'erreur de prédiction sur un corpus de données annotées.

Les réseaux de neurones, en particulier les réseaux de neurones profonds et convolutionnels (Fukushima, 1988; O'Shea & Nash, 2015), repoussent les scores de performances des systèmes d'apprentissage automatique dans plusieurs domaines, dont la reconnaissance d'image et la reconnaissance de la parole (Gu *et al.*, 2018).

## 1.6 Modélisation séquentielle/temporelle

Avec les techniques de la section précédente, la reconnaissance visuelle d'une forme pourrait être réalisée à partir de l'état des pixels d'une image à un instant donné. Toutefois, certaines classes telles que relatives à des mots prononcés ou l'état de stress ne se caractérisent pas par l'état des entrées à un instant donné. Plutôt, elles se caractérisent par l'évolution des entrées au cours de plusieurs états consécutifs. Par exemple, l'évolution des énergies fréquentielles caractérise la parole ; tandis l'évolution de caractéristiques physiologiques (intervalles cardiaques et respiratoires) caractérise le stress. Comment modéliser l'évolution des valeurs d'entrées pour éventuellement pouvoir leur attribuer une classe ou une valeur ?

Les approches vues précédemment effectuent une modélisation entrée/sortie instantanée, donc sans mécanisme (ou mémoire) permettant de rattacher les valeurs d'entrées d'un instant à l'autre. Leur utilisation à l'état brut (c'est-à-dire sans stratégie venant altérer le fonctionne-

ment des modèles) nécessite ainsi d'avoir en entrée un nombre  $T$  prédéterminé et fixe d'états consécutifs du vecteur de caractéristiques, comme présenté à la figure 1.7.

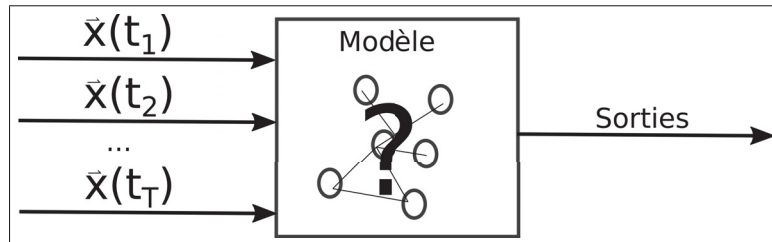


Figure 1.7 Modélisation temporelle avec un système d'apprentissage sans mémoire

Lorsque le nombre d'états consécutifs nécessaires à la modélisation est grand, ou encore variable, cette stratégie s'avère plus difficile. Remarquons, en outre, que le besoin de modéliser une quantité grandissante de relations entre variables augmente de façon conséquente la complexité requise pour le modèle.

Il reste donc pertinent de considérer des modèles spécialisés pour les données séquentielles/temporelles, tels que le modèle de Markov caché (Rabiner & Juang, 1986) et les réseaux de neurones LSTM (*Long-Short-Term-Memory*) (Hochreiter & Schmidhuber, 1997) détaillés ci-dessous pour donner un exemple de modèle séquentiel à l'état-de-l'art.

### 1.6.1 Réseaux de neurones LSTM

Un réseau de neurones LSTM est composé d'unités LSTM connectées en cascade venant remplacer les « perceptrons » des réseaux de neurones standards. La sortie  $y$  d'une unité LSTM s'obtient par une cascade de fonctions non linéaires à la fois : du vecteur d'entrée  $x$ , de l'état interne de l'unité  $c$  et de la valeur de sortie à l'instant précédent. La logique du calcul, bien résumée par Van Houdt *et al.* (2020), est schématisée à la figure 1.8.

La cellule est composée :

- d'une porte d'oubli servant à pondérer la valeur de la cellule à l'instant précédent  $c(t-1)$  via sa sortie  $f$  :

$$f^{(t)} = \sigma \left( W_f x^{(t)} + R_f y^{(t-1)} + p_f c^{(t-1)} + b_f \right) \quad (1.3)$$

où  $W_f$   $R_f$  et  $p_f$  sont les poids de  $x(t)$ ,  $y(t-1)$  et  $c(t-1)$ ,  $b_f$  est un biais, et  $\sigma$  une fonction d'activation (sigmoïde).

- d'une porte d'entrée servant à activer/désactiver l'unité via sa sortie  $i$ , selon l'entrée  $x(t)$ , les valeurs de la cellule  $c(t-1)$  et de la sortie du LSTM  $y(t-1)$  à l'instant précédent :

$$i^{(t)} = \sigma \left( W_i x^{(t)} + R_i y^{(t-1)} + p_i c^{(t-1)} + b_i \right) \quad (1.4)$$

où  $W_i$   $R_i$  et  $p_i$  sont les poids de  $x(t)$ ,  $y(t-1)$  et  $c(t-1)$ ,  $b_i$  est un biais.

- le bloc d'entrée qui génère une valeur  $z$  en fonction de l'entrée (équivalent d'un perceptron) :

$$z^{(t)} = g \left( W_z x^{(t)} + R_z y^{(t-1)} + b_z \right) \quad (1.5)$$

où  $W_z$   $R_z$  sont les poids de  $x(t)$  et  $y(t-1)$ ,  $b_z$  est un biais.

- la porte de sortie dont la sortie  $o$  est fonction de la valeur de la cellule  $c(t)$ , l'entrée  $x(t)$  et la sortie  $y(t-1)$  à l'instant précédent :

$$o^{(t)} = \sigma \left( W_o x^{(t)} + R_o y^{(t-1)} + p_o c^{(t)} + b_o \right) \quad (1.6)$$

où  $W_o$   $R_o$  et  $p_o$  sont les poids de  $x(t)$ ,  $y(t-1)$  et  $c(t)$ ,  $b_o$  est un biais.

- puis du bloc de sortie qui génère la sortie  $y(t)$  du LSTM à partir de la valeur de la cellule  $c(t)$  et de la valeur de la porte de sortie  $o$  :

$$y(t) = g(c^{(t)} \times o^{(t)}) \quad (1.7)$$

Les poids des unités peuvent également être calculés via l'algorithme de rétropropagation.

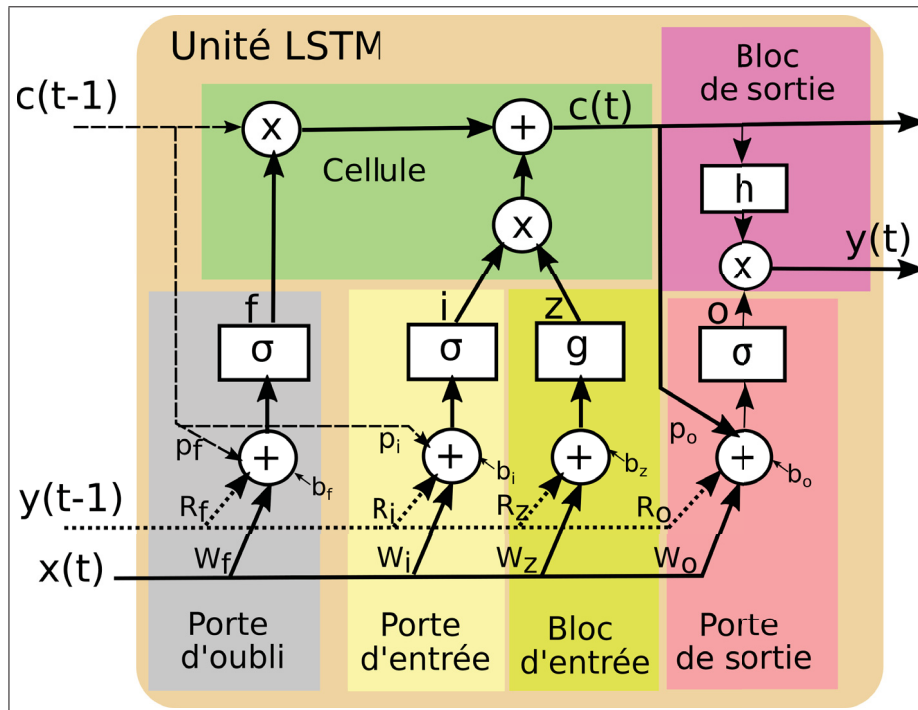


Figure 1.8 Unité LSTM

Les réseaux LSTM contribuent significativement aux records de performances des systèmes contemporains de reconnaissances de la parole (Van Houdt *et al.*, 2020).

## 1.7 Réseaux conceptuels

Les réseaux conceptuels ont jusqu'à présent été utilisés comme outils analytiques, par exemple en économie (Yarlikas & Bilgen, 2012; Earp, 1996), en éducation (Tomičić & Schatten, 2020; GhaviDel *et al.*, 2020), en conception (Georgiev & Nagai, 2011) et en traitement du langage naturel (Zhao *et al.*, 2008; Jadidinejad *et al.*, 2016; Caballero *et al.*, 2017). La façon dont les idées sont organisées et représentées est généralement définie manuellement en fonction de l'utilité du réseau dans son contexte d'application. Il est facile de schématiser et de comprendre le réseau : les concepts se rattachent à des mots et les liens expriment leurs relations. Le nombre de concepts et de relations demeure généralement faible. Certaines approches ont été développées pour identifier les concepts automatiquement, par exemple en identifiant les

mots dans les textes pour des applications de représentation de la langue : (Zhao *et al.*, 2008; Jadidinejad *et al.*, 2016; Caballero *et al.*, 2017). La figure 1.9 présente la façon générale de représenter le réseau : on voit les concepts encadrés (nous aurions normalement des mots à la place de *Concept X*, *Concept Y* et *Concept Z*), liés par des relations sémantiques. Le sens, la nature et la présence de relations dépendent de la nature des concepts.

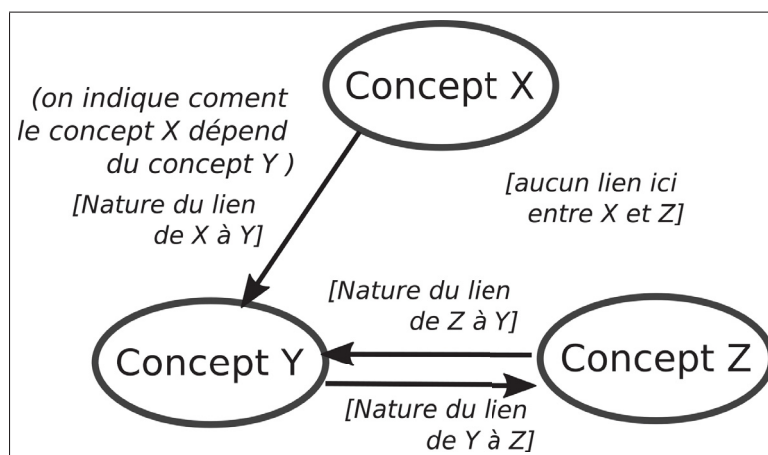


Figure 1.9 Vue générale d'un réseau conceptuel : les unités sont les concepts, liés entre eux par des relations sémantiques

Contrairement aux réseaux conceptuels précédents, ceux présentés dans cette thèse ne servent pas d'outils d'analyse, mais plutôt de modèle pour l'apprentissage automatique des machines. Ils sont inférés automatiquement à partir d'entrées sensorielles brutes de nature quelconque : image, son, données tactiles, thermiques, etc. Ainsi, la structure du réseau, la signification des concepts et leurs relations spécifiques ne sont ni définies ni connues a priori par l'investigateur ; elles résultent plutôt d'un mécanisme dit de *conceptualisation* (décrit au chapitre 4). Comme nous le verrons au chapitre 6, la complexité des réseaux inférés (qui peuvent avoir des milliers, voire des millions de concepts et de relations) rend leur utilisation comme outil d'analyse effectivement plus difficile, contrairement aux réseaux conceptuels antérieurs - c'est-à-dire que l'inspection manuelle (exploration de la structure des relations et des concepts) sera pratiquement infaisable. Cependant, l'analyse informatique automatique de la structure logique reste envisageable.

## 1.8 Tenter un nouveau paradigme

Les modèles d'apprentissage-machine existants (réseaux de neurones, machine à vecteurs de supports, réseaux bayésiens, arbres de décision) fonctionnent sous un paradigme *entrée-sortie*, telles des fonctions non linéaires proposant un résultat à partir de variables d'entrée. La structure et les paramètres du modèle sont choisis, inférés et optimisés de manière à ce que le système puisse générer une ou plusieurs sorties appropriées (classification, régression ou contrôle pour les systèmes robotiques) en fonction de variables, ou séquence de variables, en entrée du système.

Comme la structure et les valeurs optimales des paramètres du modèle dépendent d'un objectif spécifique, elles seront compromises si l'on souhaite utiliser le modèle pour différents objectifs. Par conséquent, un modèle optimisé pour un seul objectif donnera de meilleurs résultats pour cet objectif qu'un second devant également performer sur un deuxième objectif. Si les deux objectifs impliquent des structures et des paramètres optimaux trop différents, il devient nécessaire d'employer un modèle distinct pour chaque objectif. C'est ainsi que la multiplication des objectifs entraîne la multiplication des modèles.

Remarquons qu'un modèle basé sur le paradigme *entrée-sortie* peut fructueusement être réutilisé, en partie, à des fins variées. Par exemple, la machine de Boltzmann (un type de réseaux de neurones) (Hinton, 2007) permet de détecter des caractéristiques pouvant servir d'entrée à des systèmes de prédiction variés. Cependant, la proportion réutilisable du modèle d'un objectif à l'autre reste limitée.

L'approche classique de l'apprentissage automatique via des unités d'entrée/sortie se limite, implicitement, à utiliser l'information observée en entrée pour générer des sorties. Dans cette approche, les unités ne modélisent donc pas, en réalité, la structure logique du phénomène observé. Plutôt, elles extraient de ce phénomène des éléments utiles à un usage spécifique : détection de caractéristiques, détection de formes, génération d'une mesure ou d'une commande, etc. Mais si l'intelligence s'appuyait plutôt sur une représentation pure des expériences phénoménologiques ? Par exemple, l'esprit humain crée et génère continuellement de nouveaux

objectifs au cours de sa vie. La capacité à accomplir des tâches complexes et abstraites augmente avec l'expérience, ce qui implique que les structures acquises sont réutilisées et que leur optimalité ne dépend pas étroitement des tâches spécifiques pour lesquelles elles ont été motivées. La réalisation d'une intelligence globale et polyvalente, comparable à celle de l'animal, pourrait ne pas être possible sans une représentation évoluée des phénomènes observés.

Dans cette perspective, l'approche proposée vise à extraire des expériences sensorielles leurs éléments structuraux, représentés par des unités ayant des relations composants/agrégats et représentation/abstraction qui reflètent celles des objets réels liés aux expériences. La complexité de la structure est par conséquent liée à la complexité des phénomènes observés ainsi qu'à la sensibilité du système à ses états et variations ; sans dépendre directement d'objectif d'utilisation du système, sortie désirée de classification, régression ou contrôle. Il demeure néanmoins possible d'utiliser le réseau en amont à des fins spécifiques telles que la prédiction, ce qui sera démontré au chapitre 5.





## CHAPITRE 2

### RELATIONS ENTRE LES CONCEPTS

Dans un réseau conceptuel, les concepts sont liés par des relations *sémantiques* où la nature de la relation d'un concept A vers un concept B contribue soit à :

- *composition*. définir quelle est la composition de A via des relations de compositions (un chat est composé d'un oeil) ;
- *agrégation*. indiquer dans quels contextes A est observé via les relations d'agrégations (un oeil est observé dans un chat) ;
- *abstraction*. indiquer quels sont les concepts (parents) qui généralisent A via les relations d'abstraction (un chat est un félin) ;
- *représentation*. et présenter des exemples plus spécifiques de A (concepts enfants), via les relations de représentation (un exemple de félin est un chat).

Ces types de relations ont été utilisées notamment dans la programmation orientée objet (Dahl *et al.*, 1968) pour décrire les relations entre les *classes*. Elles sont réutilisées ici pour décrire les relations générales entre les « concepts ». Remarquons cependant que le réseau conceptuel créé par les mécanismes présentés au chapitre 4 ne suit pas la structure d'un arbre quant aux relations d'agrégation/composition ou d'abstractions/représentations, comme ils permettront au concept d'avoir plusieurs abstractions et plusieurs agrégations.

#### 2.1 Représentation/Abstraction

« Minex » est un « Chat ». Minex est dit *représentant* de Chat et Chat est dit *abstraction* de Minex. On dit alors que Chat *admet* Minex en tant que représentant. Chaque concept mémorise ses abstractions et ses représentants. Lors de l'ajout de liens d'abstraction et de représentation, la cohérence globale de la hiérarchie d'abstraction doit être respectée. Je nomme *abstraction explicite* de A une abstraction référencée par A (un félin pourrait être une abstraction explicite d'un chat) ; une *abstraction implicite* une abstraction obtenue récursivement des listes des

abstractions de A (un animal serait une abstraction implicite d'un chat); et simplement *abstraction* pour dénoter une abstraction explicite ou implicite. La même logique de nomenclature est utilisée pour les représentants.

Pour ajouter B dans la liste des abstractions de A, les conditions suivantes doivent être respectées. Un exemple pour chaque cas est présenté à la figure 2.1.

- **Condition A.** A ne doit pas avoir B dans ses représentants; et B ne doit pas avoir A dans ses abstractions, ce qui n'aurait en effet aucun sens (à la figure 2.1 : A :Oeuf, B :Poule).
- **Condition B.** A ne doit pas déjà avoir B dans sa liste d'abstractions implicites; autrement rien n'est à faire (A :Violon, B :Instrument).

Ensuite, afin d'assurer la cohérence de la hiérarchie d'abstraction :

- **Condition C.** Si un représentant C de A possède B dans sa liste d'abstractions explicites, B peut être retiré de cette liste avant d'être ajouté à A (A :Instrument de musique, B :Instrument, C :Violon).
- **Condition D.** Si une abstraction explicite C de A est également une abstraction de B, A ne doit plus la référer (A :Violon, B :Instrument à cordes, C :Instrument).

## 2.2 Composant/Agrégat

Une « Tête » est composée, par exemple, d'un « Nez ». Tête est dit un *agrégat* de Nez, et Nez un *composant* de Tête. Chaque concept mémorise une liste de ses agrégats et une liste de ses composants. Les *agrégats explicites* d'un concept X sont directement référencés par X; tandis que les *agrégats implicites* incluent aussi les agrégats des abstractions de X. Considérant la figure 0.1, le concept « Oeil de Minex » aurait par exemple « Tête de Minex » comme agrégat explicite et {« Tête de Minex »,« Tête de Chat »} comme agrégats implicites.

L'*arborescence structurelle* d'un concept désigne l'ensemble de sa composition considérée de façon récursive avec celles de ses composants, sous-composants, etc, telle que présentée à la figure 2.2 pour le concept « Chat » à gauche et « Minex » à droite.

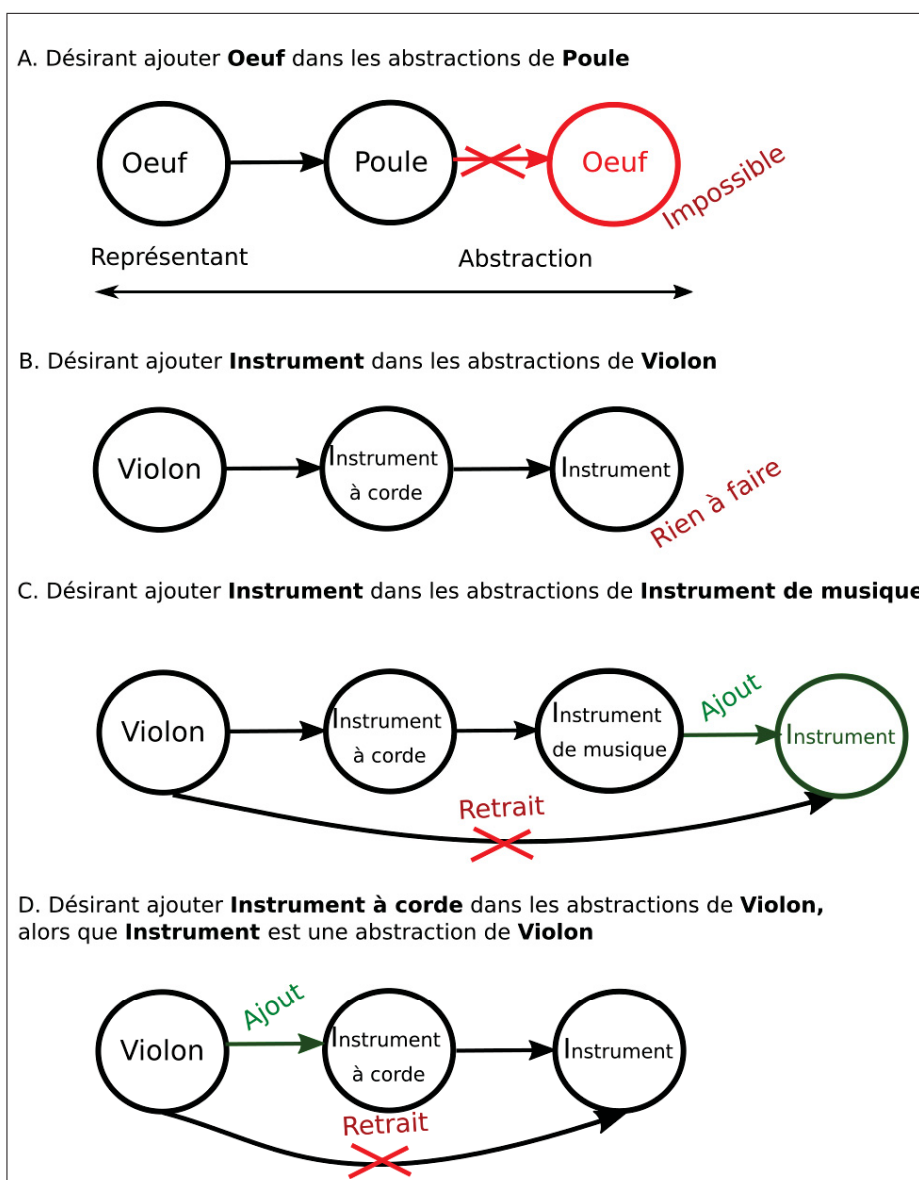


Figure 2.1 Exemple de conditions lors de l'ajout d'une abstraction

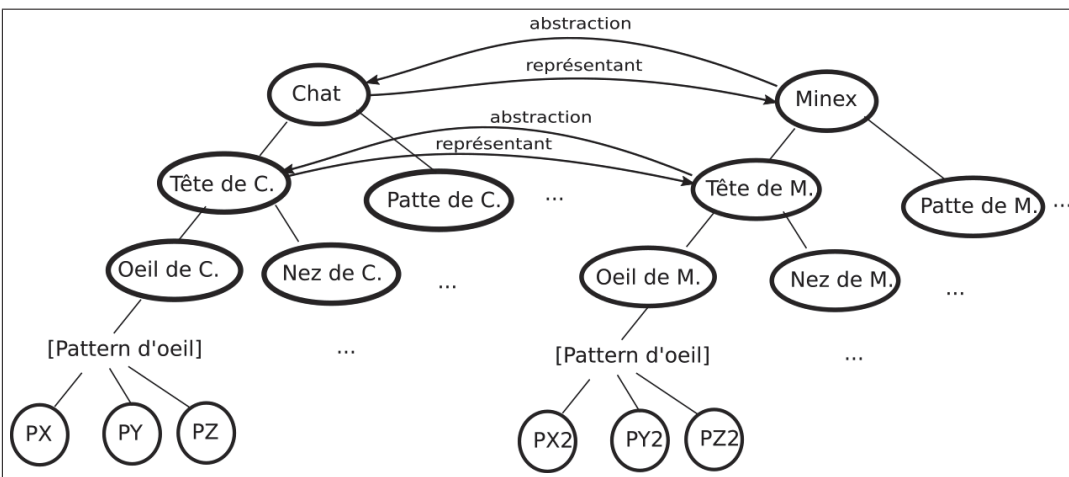


Figure 2.2 Représentation conceptuelle d'un chat général et d'un chat « Minex »

## CHAPITRE 3

### NATURE DES CONCEPTS

Lors de l'introduction, nous avons défini un concept comme étant une *représentation logique d'une récurrence perceptuelle caractéristique*. Or, nous percevons le monde via nos sens : des images avec nos yeux, des sons avec nos oreilles, la chaleur avec notre peau... Pour chaque sens, nous sommes en mesure de percevoir différents états : fréquences audibles à différents niveaux d'énergie, intensités lumineuses, etc. Cette perception s'appuie sur des récepteurs sensoriels (oreille, oeil, nez) dotés de cellules sensorielles (cellules ciliées, photoréceptrices, olfactives) pouvant générer un influx nerveux qui dépend de l'état d'un phénomène (son, image, odeur) (Cordier, 1964). L'activation est acheminée en cascade à d'autres neurones interconnectés dans un réseau complexe, lesquels pourraient encoder des récurrences perceptuelles de façon plus abstraite (Dehaene *et al.*, 1998). Sans nécessairement chercher à copier le système neurologique humain, quelles unités logiques permettraient tout de même de créer un système de représentation des *récurrences perceptuelles caractéristiques* ?

La première étape serait de créer une unité qui représente une plage de valeurs d'un état sensoriel particulier. Ces unités en entrée de la perception, que nous pouvons ainsi nommer « prémices », formeraient la base d'une hiérarchie de concepts, dits *secondaires*, de plus en plus abstraits. Ces derniers agrégeraient ainsi les prémices, puis éventuellement d'autres concepts secondaires, selon une certaine structure. Est-ce que la séquence des composants est importante ou non ? Nous distinguerons alors les concepts dits *ordonnés* des concepts dits *désordonnés*. Est-ce que ces concepts seraient formés de composants relatifs à un même type de perceptions ? Nous distinguerons alors les concepts dits *homogènes* des concepts dits *hétérogènes*.

#### 3.1 Prémice

La *prémice* est l'unité de base d'une hiérarchie de concepts.

Elle peut d'abord représenter un état sensoriel particulier décrit par une plage de valeurs. La prémice est liée, dans ce cas, à un type spécifique de phénomènes perceptuels, tel qu'un son, l'intensité d'une couleur, etc. Les prémices de même type sont liées par une hiérarchie d'abstraction telle que représentée à la figure 3.1 où elles encodent, dans l'exemple, des intensités lumineuses de 0 à 1. Nous vérifions que L31 est une abstraction de L41 puisque la plage lumineuse de L41[0-0.25] est comprise dans celle de L31[0-0.5]. De cette façon, la prémice L11[0-1] à la racine de la hiérarchie d'abstraction est l'abstraction commune à toutes les prémices d'intensité lumineuse.

Les arbres de prémices peuvent être créés à l'initialisation du réseau de façon supervisée, par l'investigateur, avec les relations d'abstraction/représentation conséquentes, qui demeureront par conséquent fixes<sup>4</sup>. Formant la « matière première » de toute composition, je considère pour l'instant préférable (pour fins d'optimisation) que les prémices ne mémorisent pas par défaut, contrairement aux autres concepts, des listes d'agrégats. Une prémice serait ainsi dans un état statique tout au long de sa durée de vie.

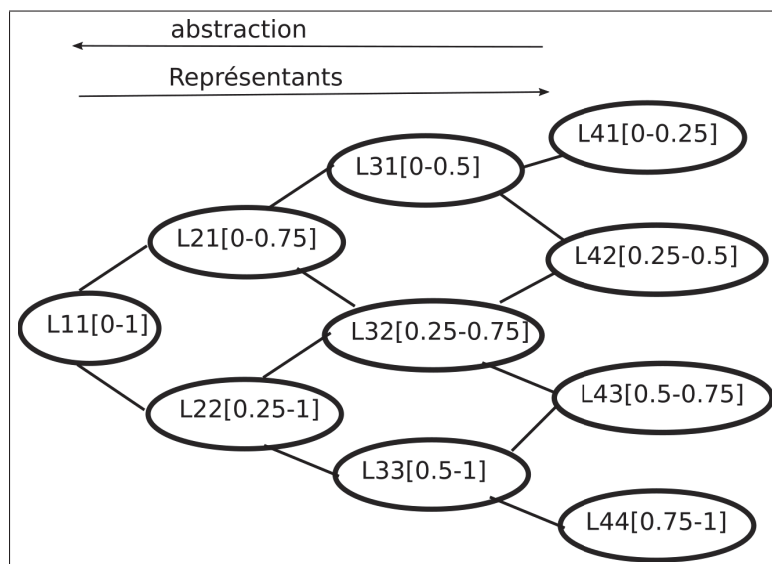


Figure 3.1 Hiérarchie d'abstraction pour une intensité lumineuse de 0 à 1

4. Remarquons toutefois qu'il serait intéressant, pour de futures recherches, d'étudier le bien-fondé de créer dynamiquement l'arbre de prémices de façon à augmenter l'acuité de la perception dans certaines zones.

### 3.2 Concepts homogènes versus hétérogènes

Un concept composé d'expériences sensorielles de différentes natures est dit *hétérogène*, tandis qu'un concept composé d'expériences sensorielles de même nature est dit *homogène*.

Le *type* d'un concept homogène, désigné par le symbole  $v$ , désigne le type d'expérience phénoménologique qu'il encode. Il peut s'agir d'un phénomène physique : visuel, sonore, tactile ; comme d'un phénomène abstrait synthétisé par un système tiers.

Le type d'un concept hétérogène est défini par les types de ses composants.

### 3.3 Concept structurel

Un concept secondaire est *structurel* au sens qu'il encode une composition d'éléments selon un ordre flexible ou non. Son identité repose, par conséquent, d'abord sur celles des composants qu'il rassemble, soient des concepts possiblement de différentes natures : prémices et/ou concepts structurels.

Ensuite, si l'identité du concept repose aussi sur l'ordre des éléments, le concept est dit *ordonné*. Le symbole  $o$  désigne un concept ordonné. Pour qu'un concept ordonné soit détecté, ses éléments doivent être observés selon un ordre spécifique. Par exemple, la figure 3.2 présente quatre compositions distribuées selon différents ordres. Ces compositions peuvent être représentées respectivement par quatre concepts ordonnés tels que :  $C1\{L41, L41, L42, L44\}^o$ ,  $C2\{L41, L44, L42, L41\}^o$ ,  $C3\{L41, L42, L44, L41\}^o$  et  $C4\{L44, L42, L41, L41\}^o$ .



Figure 3.2 Exemple de quatre concepts ordonnés distinctifs représentant un seul concept désordonné

Si l'identité du concept ne dépend pas de l'ordre des éléments, mais seulement de leurs identités, il est dit *désordonné*. Le symbole  $\rho$  désigne un concept désordonné. Les quatre compositions de la figure 3.2 seront donc tous représentées par un seul concept désordonné tel que  $C5\{L41, L41, L42, L44\}^\rho$  (en indiquant les prémices dans l'ordre de leur numéro). Remarquons que ce concept est une abstraction commune des quatre concepts ordonnés  $C1$  à  $C4$ , puisqu'il est implicitement détecté dès qu'un de ces quatre derniers concepts est détecté.

Les concepts structuraux sont générés automatiquement (section 4.3 sur la genèse conceptuelle).

L'arborescence structurelle d'un concept peut impliquer une mixture de concepts ordonnés et de concepts désordonnés. Les concepts ordonnés augmentent la rigidité de la structure ; tandis que les concepts désordonnés augmentent sa plasticité (comme on l'observe à la figure 3.2 où le concept désordonné  $C5$  accepte les quatre représentations  $C1$  à  $C4$ ).

En pratique, le fait de trier en mémoire les composants des concepts désordonnés facilite leur identification. Les critères de tri doivent être invariants et assurer la concordance de position entre deux compositions équivalentes. Dans notre implémentation par exemple, les concepts désordonnés sont contenus en mémoire dans les feuilles d'un arbre où l'embranchement d'un noeud à l'autre, à partir de la racine, est réalisé selon les concepts consécutifs de la composition. Selon la figure 3.3, si nous observons une composition désordonnée  $\{C2, C44\}$ , nous pouvons trouver qu'elle définit le concept  $C65$ .

Puisque nous pouvons considérer qu'une prémice est composée que d'elle-même, un concept dont tous les composants correspondent à une même prémice correspond à cette prémice. Par exemple, rassembler quatre éléments jaunes donne encore un élément jaune équivalent. Par opposition, rassembler quatre fois un même motif carotté donne un motif avec quatre fois plus de carreaux, lequel est donc distinctif par rapport à un élément de sa composition. Par conséquent, un concept composé d'un même concept secondaire ne correspond pas à ce concept.



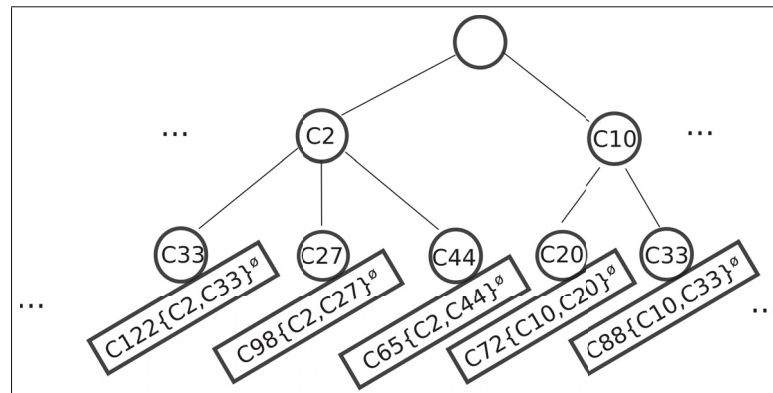


Figure 3.3 Mémoriser les concepts dans un arbre de façon à trouver facilement le concept correspondant à une composition observée

### 3.4 Relation d'abstraction/représentation des concepts secondaires

Un concept structural A a pour abstraction un concept B :

- Si A et B sont des concepts structuraux ordonnés, et que chaque composant de B admet le composant de A de même position ;
- Si A est un concept ordonné ou non, B un concept désordonné et qu'il est possible de jumeler chaque composant X de B à un composant distinctif Y de A qui est soit le même concept X ou un représentant de X ;
- Si B est une prémice qui admet l'ensemble des prémices aux feuilles de l'arborescence structurelle de A ;

### 3.5 Plasticité et rigidité

La « plasticité » d'un concept est sa capacité à admettre différentes formes ; par opposition à sa « rigidité » qui est sa capacité à discriminer des formes particulières. La plasticité est au niveau de la structure du concept et au niveau de son essence, c'est-à-dire de la nature des prémices aux feuilles de sa composition. Plus son arborescence structurelle comporte des concepts désordonnés, plus sa structure est plastique. Plus elle comporte à ses feuilles des prémices abstraites, plus son essence est plastique. Par exemple, le concept  $\{L41, L43, L44\}^{\phi}$

aurait une plus grande plasticité que le concept  $\{L11, L32, L44\}^o$  quant à sa structure, mais une plus grande rigidité quant à son essence puisque  $L41$  et  $L43$  sont plus spécifiques que  $L11$  et  $L32$ .

Chaque système d'apprentissage comporte un *dilemme biais-variance*, soit un compromis entre la capacité du système à représenter fidèlement un phénomène ; et sa capacité à généraliser ce phénomène de manière à pouvoir l'associer à des phénomènes analogues. Dans un réseau conceptuel, le *dilemme biais-variance* s'observe dans un compromis entre la rigidité et la plasticité des structures de représentation.

### 3.6 Concepts et neurones

L'image 3.4 présente une vue des éléments des concepts dans un réseau conceptuel. Chaque concept a une identité définie par sa composition et, pour les prémices, une plage de valeurs décrivant un phénomène physique ou abstrait. Les concepts ont des connexions statiques (en gris) vers d'autres concepts qui déterminent leurs compositions, puis ils ont des connexions dynamiques (en vert) vers des abstractions, des représentants et des agrégats. Ces connexions sont en effet dynamiques puisqu'elles doivent être mises à jour tant que le réseau se construit par ses expériences selon l'occurrence des agrégats et selon les nouveaux concepts qui s'ajoutent, lesquels changeront les liens de représentation et d'abstraction selon les règles décrites à la section 2.1. Les prémices, cependant, initient que des connexions statiques vers des représentants et des abstractions (soient d'autres prémices) ; bien que d'autres concepts peuvent avoir des liens dynamiques vers les prémices. Les liens d'abstractions et de représentations des prémices sont établis à l'initiation du réseau selon leurs valeurs et plages de variation respectives.

Comme le concept est, en somme, une unité distincte connectée à d'autres concepts, il peut évoquer un « neurone » du cerveau humain. Les auteurs des premiers modèles d'apprentissage machine basés sur des unités interconnectées ont ainsi appelé « neurones » leurs unités mathématiques ayant la capacité d'apprendre de leurs expériences (Müller *et al.*, 2012). Le neurone fut alors proposé sous le nom de *perceptron* qu'on a décrit à la section 1.5.5. Le « concept »

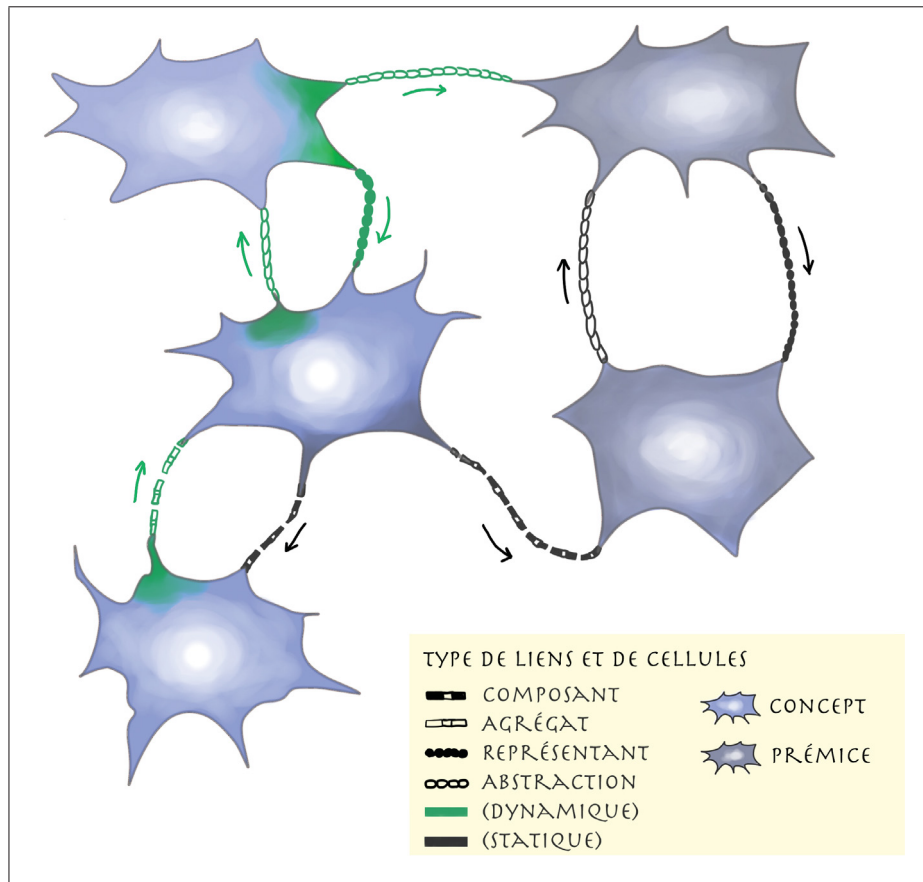


Figure 3.4 Relations entre plusieurs concepts interconnectés

peut s'apparenter au perceptron au moment de l'activation (voir chapitre 5), puisque l'énergie est acheminée d'un concept à l'autre avec une intensité qui est pondérée par un poids spécifique à la connexion. À la grande différence des réseaux de perceptrons :

- le réseau conceptuel a une structure dynamique, puisque les concepts secondaires et les connexions sont créés de façon automatique avec l'expérimentation du système ;
- le réseau conceptuel encode les structures logiques des expériences phénoménologiques (sa structure n'est donc pas fonction d'objectif de prédiction, classification, etc). Les relations entre les unités ont ainsi une sémantique.
- l'intensité avec laquelle l'énergie est transférée d'une unité A vers une unité B dépend de l'affinité entre les deux unités, qui est défini à partir des observations sans égard aux objectifs d'utilisation du réseau.

Par rapport au perceptron, le « concept » peut s'apparenter davantage à un neurone biologique pour les deux premières distinctions susmentionnées. Cependant, je considère que le terme « concept » reflète mieux la fonction logique de l'unité, en plus d'éviter la confusion avec le terme « neurone » déjà largement employé pour désigner le perceptron.

Bien que les processus biologiques des neurones du cerveau ont été étudiés (Allen & Eroglu, 2017; Elson *et al.*, 1998; Korn & Faure, 2003) et comportent quelques caractéristiques similaires aux réseaux de neurones ou réseaux conceptuels, il importe de souligner que le fonctionnement logique du système de la pensée est encore à découvrir et pourrait s'avérer différent de ces deux systèmes. Néanmoins, la quantité de neurones, entre 100 millions et 100 milliards de neurones en fonction de l'espèce, et de connexions présentes dans le cerveau, autour de 164 téras synapses (Nguyen, 2010), peut donner une idée de la complexité des structures requises à d'obtention d'une intelligence *évoluée*.

Maintenant que nous avons présenté la nature des concepts, une question s'impose en particulier : « comment les détecter et les générer à partir d'expériences sensibles ? » Le prochain chapitre apportera une solution à cette question via des méthodes dites de « conceptualisation » et de « genèse conceptuelle ».

## CHAPITRE 4

### CONCEPTUALISATION

La conceptualisation est le mécanisme responsable de la détection des concepts spécifiques admettant les valeurs d'entrées, et du groupement des récurrences perceptuelles en vue de la création des concepts du réseau. La conceptualisation s'effectue sur un transpositif d'entrée particulier, en définissant à chaque noeud du transpositif le concept le plus spécifique (déjà présent dans le réseau) pour représenter la composition liée aux noeuds enfants.

#### 4.1 Transpositifs

##### 4.1.1 Structure du transpositif et utilité

Le *transpositif* détermine l'arborescence structurelle des concepts décrivant une expérience. Il sert ainsi de support au mécanisme de conceptualisation. Il est composé de noeuds typiquement interconnectés selon la structure d'un arbre entrelacé.

Un transpositif est utilisé pour détecter/générer des concepts de même type  $v$ . Il sera ainsi désigné par le symbole  $\Gamma^v$ . Si plusieurs transpositifs d'entrées de mêmes types sont utilisés, le numéro  $n$  est ajouté au transpositif désigné :  $\Gamma^{nv}$ . Un transpositif peut également servir à définir l'arborescence structurelle de concepts hétérogènes et/ou temporels, comme nous le verrons pour l'application de reconnaissance de la parole 6.2.

Les noeuds intérieurs d'un même transpositif ont tous le même nombre d'enfants afin de garantir la cohérence structurelle des concepts de même type, laquelle est nécessaire par exemple pour qu'un même concept puisse être détecté à différents endroits du transpositif et pour établir des relations d'abstractions et représentations entre les concepts de même type.

Chaque feuille du transpositif est liée à une entrée et contient une prémisse qui est mise à jour selon la valeur de l'entrée. Le transpositif est structuré de sorte à décrire la relation (temporelle, spatiale ou autre) entre les entrées. Un système peut avoir plusieurs transpositifs d'entrées de

même type. Ce sera le cas par exemple pour un système utilisant deux caméras, deux micros, etc. Si deux transpositifs ont une structure identique et utilisent des données de même type, les concepts structuraux détectés sur un transpositif pourront également être détectés sur l'autre transpositif.

Pour l'instant, la structure de chaque transpositif d'entrée est réalisée a priori par le développeur du réseau conceptuel, et demeure fixe.

Le nième noeud de profondeur  $p$  est noté  $\phi^{p,n}$  et le nombre de noeuds sur cette profondeur est noté  $N^p$ . Chaque noeud intérieur  $\phi^{p,n}$  possède des enfants  $\xi_{p+1,m}^{p,n}$ , où  $m$  désigne l'indice d'un noeud de la profondeur subséquente  $p + 1$  et varie de 1 au nombre d'enfants du noeud  $N^{\xi}$ , identique pour tous les noeuds intérieurs.

#### 4.1.2 Information mémorisée en chaque noeud du transpositif

Chaque noeud intérieur  $\phi^{p,n}$  mémorise un *concept nodal*  $c^{p,n}$  (actualisé par la conceptualisation) qui définit la *structure* du phénomène observé dans la région sensorielle couverte par le noeud ; et une *prémice nodale*  $p^{p,n}$  qui définit l'essence des valeurs observées dans cette même région du noeud (voir figure 4.1). La prémice nodale est actualisée avant la conceptualisation.

Le choix de la prémice nodale, selon la région qu'elle représente, n'est pas univoque et cet aspect pourrait être étudié davantage dans de futures recherches<sup>5</sup>. Pour l'instant, la prémice nodale retenue est celle dont la valeur correspond à la moyenne pondérée (en donnant plus de poids au centre) et à l'écart-type des valeurs des entrées représentées par les prémices aux feuilles de l'arborescence du noeud.

Un exemple de transpositif est présenté à la figure 4.2 avec en entrée, à la base, une grille de niveaux de gris actualisée selon une image observée. La prémice à chaque feuille est sélection-

---

5. J'ai testé quelques approches : considérer la prémice dont la valeur correspond au mieux à celle observée à l'entrée au centre de la région ; considérer la prémice dont la valeur correspond à la moyenne pondérée (en donnant plus de poids au centre) et à l'écart-type des valeurs de la région du noeud ; considérer la prémice correspondant à l'abstraction commune des prémices aux feuilles de l'arborescence du noeud. Les deux premières approches permettaient des performances analogues dans les tests de reconnaissances d'images, mais la première apparaît plus robuste aux bruits sur les pixels des images

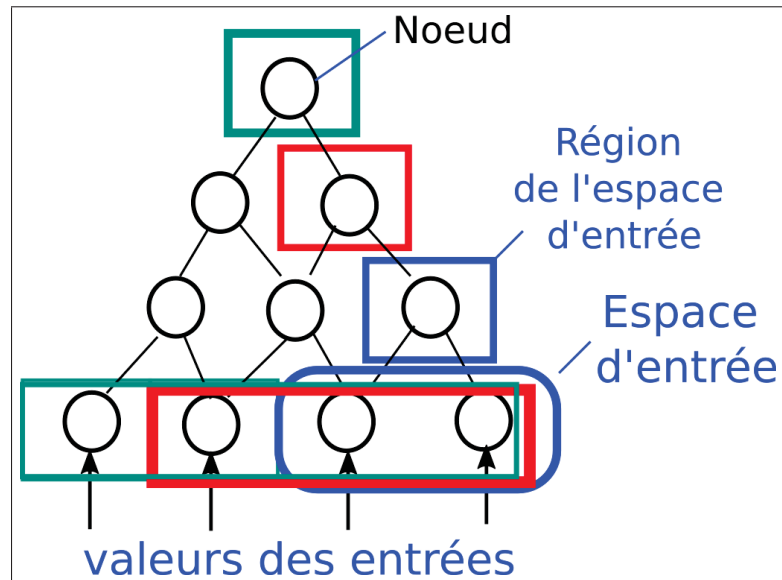


Figure 4.1 Région de l'espace d'entrée représentée par chaque prémice nodale, exemple

née selon la valeur de l'intensité lumineuse du pixel correspondant de l'image. Par exemple, une valeur d'intensité lumineuse de 0.6 donnerait lieu à la prémice L43, considérant la figure 3.1. Comme présenté aux figures 4.2 et 4.3, chaque noeud sur une profondeur a pour enfant quatre noeuds de la profondeur subséquente liés à des zones contiguës de l'image.

#### 4.1.3 Évaluation du concept nodal

La conceptualisation, expliquée plus loin, nécessite d'évaluer les concepts nodaux dans le transpositifs. Or, le processus d'évaluation d'un concept nodal est illustré à la figure 4.4. Le concept  $c^{p,n}$  admet la composition ordonnée des concepts des noeuds enfants  $\{c^{\xi p+1,m}, m = 1..N^{\xi}\}$ . On observe par exemple au noeud  $\phi^{2,2}$  la composition des concepts L41 et L43 des noeuds enfants  $\phi^{3,2}$  et  $\phi^{3,3}$ , laquelle est admise par le concept  $c^{1L}\{L41,L43\}^o$ . Remarquons que le concept nodal détecté  $c^{p,n}$  est celui le plus rigide : qui admet la composition résultant des concepts nodaux des noeuds enfants ; et qui existe a priori dans le réseau. Si le concept défini par  $\{L41,L43\}^o$  n'existe pas dans le réseau, le concept nodal de  $\phi^{2,2}$  pourrait être par exemple  $\{L41,L43\}^d$ ,  $\{L41,L32\}^o$  ou encore  $\{L21,L21\}^o$  supposant que ces concepts existent.

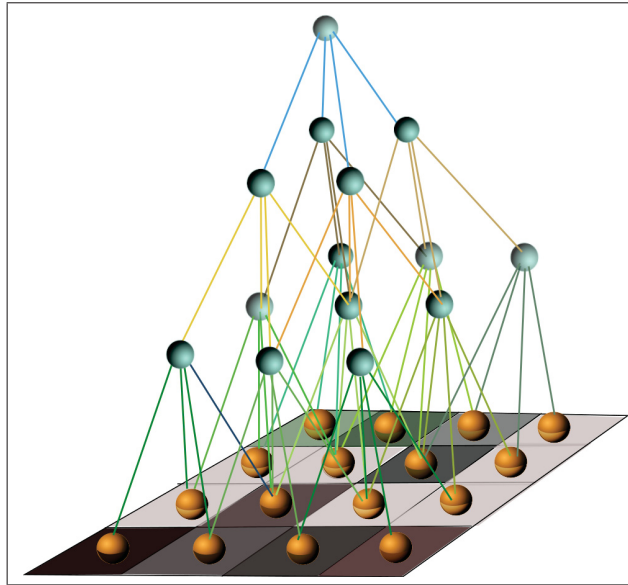


Figure 4.2 Exemple d'un transpositif dont les noeuds forment un arbre entrelacé, utilisé pour une image de 4 par 4 de niveau de gris

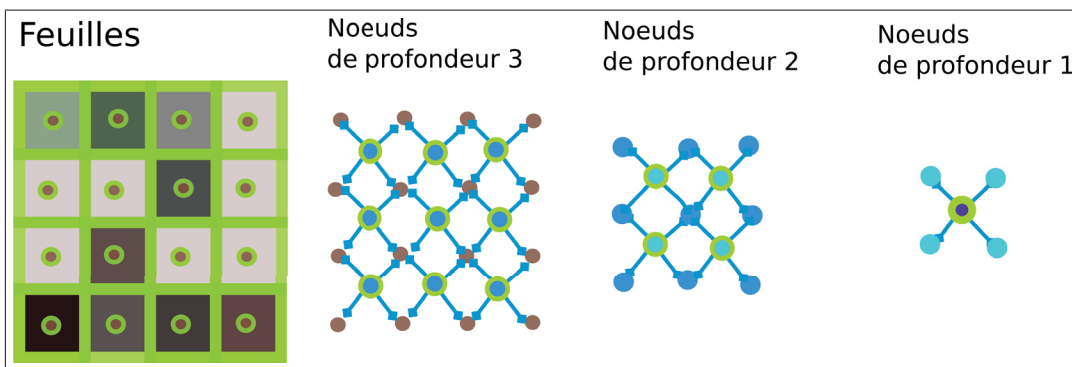


Figure 4.3 Liens entre les prémices et les zones de l'image, puis des noeuds d'une profondeur avec ceux de la profondeur inférieure - pour le transpositif de la figure 4.2

En effet, ces concepts admettent tous  $\{L41, L43\}^o$ . Dans ce cas,  $\{L41, L32\}^o$  serait choisi avant  $\{L21, L21\}^o$  puisqu'il est plus rigide (remarquons d'ailleurs que  $\{L21, L21\}^o$  est une abstraction de  $\{L41, L32\}^o$ ). L'objectif est de représenter le plus fidèlement possible l'expérience détectée avec les éléments disponibles dans le réseau.



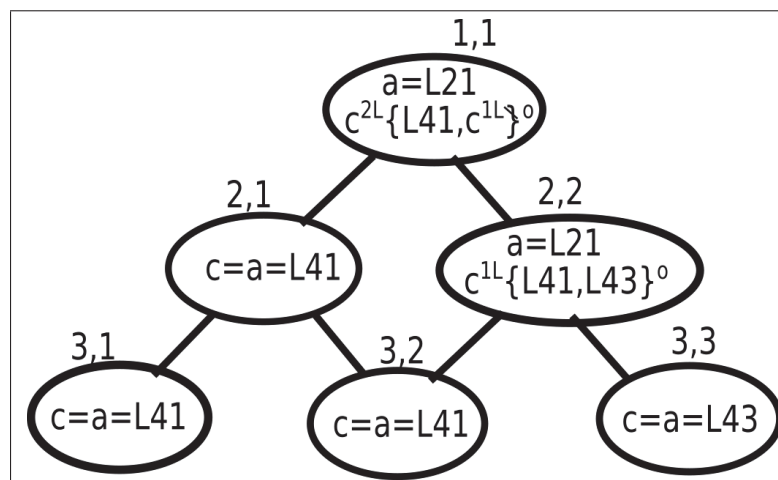


Figure 4.4 Évaluation du concept nodal dans un transpositif

## 4.2 Mécanismes de conceptualisation

Suite à une nouvelle observation, les prémices nodales sont actualisées, puis une conceptualisation peut être effectuée.

La conceptualisation repose sur les mécanismes (1) de *détection* de concepts actifs d'une profondeur de départ cible (de conceptualisation) en remontant jusqu'à la racine de l'arbre d'entrée; et (2) de *genèse* de nouveaux concepts.

Elle se répète de façon itérative en incrémentant, à chaque itération, la profondeur cible. La profondeur cible passera donc de 0 à la première itération, à 1 à la deuxième... jusqu'à la profondeur du transpositif moins un à la dernière itération (voir figure 4.5).

Cette approche itérative permet de considérer l'information d'abord sous une perspective macroscopique, en détectant et en stimulant la genèse de concepts plus abstraits (moins profonds); pour cheminer d'itération en itération vers la détection et la genèse de concepts plus spécifiques à l'expérience. Si l'expérience est celle de l'image de Minex, le concept « Chat » serait ainsi détecté dans une itération précédant la détection du concept « Minex ». Étant donnée cette

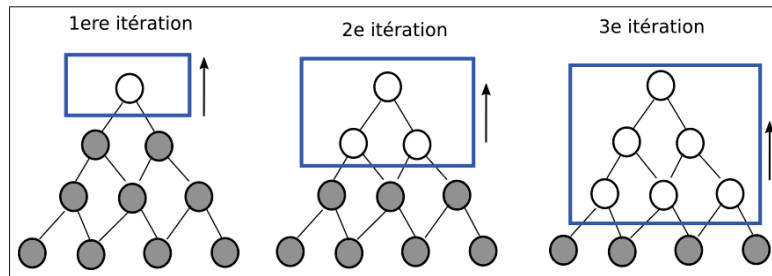


Figure 4.5 Processus de conceptualisation itératif. La profondeur de départ augmente à chaque itération, jusqu'à une profondeur maximale correspondant, tout au plus, à celle du transpositif

approche progressive, la détection d'un chat occupant une grande proportion de l'image serait aussi réalisée avant celle d'un chat occupant une plus faible proportion de l'image.

La conceptualisation dite *homogène* génère des concepts homogènes à partir d'un transpositif d'entrée particulier  $\Gamma^v$ , donc à partir de prémices de même type  $v$ .

La conceptualisation peut être *générative* ou non. Une conceptualisation *générative* participe à la synthèse de nouveaux concepts, tandis qu'une conceptualisation *non générative* est utile que pour la détection des concepts décrivant l'expérience.

#### 4.2.1 Initialisation

Avant la conceptualisation, les prémices nodales doivent être mises à jour selon les régions sensorielles, tel que décrit à la section 4.1.2.

#### 4.2.2 Processus de détection

En partant des nœuds de la profondeur de départ vers la racine du transpositif, on doit en chaque nœud  $\phi^{p,n}$ , :

1. Identifier le concept nodal  $c^{p,n}$  le plus rigide admettant la composition ordonnée formée des concepts  $c_{p+1,m}^{\xi p,n}$ ,  $m = 1 \dots N^\xi$  des nœuds enfants  $\xi_{p+1,m}^{p,n}$ ,  $m = 1 \dots N^\xi$  de  $\phi^{p,n}$  (voir

section 4.1.3)<sup>6</sup>. Pour un noeud de la profondeur de départ, le concept nodal correspond à la prémice nodale<sup>7</sup>.

2. Si aucun concept existant n'a directement la composition observée des noeuds enfants de  $\phi^{p,n}$  et que la conceptualisation est générative, alors cette composition est mémorisée comme observation pour le processus de genèse expliqué à la Section 4.3.

Ce choix de design peut susciter quelques interrogations qu'il vaille la peine de justifier ci-dessous.

- *Si un concept est généré lors de la genèse, pourquoi ne pas implicitement utiliser ce dernier comme concept nodal  $c^{p,n}$  ?* Comme nous le verrons à la section 4.3, le concept est généré à partir d'observations survenant à différents moments. Or, il est possible qu'un concept différent de celui généré soit plus adéquat pour décrire la composition courante.
- *Pourquoi ne pas considérer dans les noeuds toutes les abstractions implicites ?* Le processus de conceptualisation vise à définir et inférer des concepts spécifiques à l'expérience. Par conséquent, seul le concept le plus spécifique est utilisé en chaque noeud. Les concepts plus abstraits, bien qu'absents du transpositif, sont détectés/activés par le mécanisme d'activation décrit au chapitre 5.

Une fois qu'une itération est réalisée, des noeuds de la profondeur cible jusqu'à la racine, le transpositif encode en chaque noeud  $\phi^{p,n}$  un concept  $c^{p,n}$  décrivant le phénomène observé. En augmentant la profondeur cible d'une itération à l'autre, les concepts nodaux seront ainsi, potentiellement, de plus en plus profonds, donc également de plus en plus rigides en encodant des structures phénoménologiques plus complexes.

### 4.3 Genèse conceptuelle

La *genèse conceptuelle*, c'est la création de nouveaux concepts à partir d'observations groupées à diverses ramifications des arborescences structurelles des concepts existants.

---

6. Remarquons qu'il existe toujours une abstraction admettant la composition, que ce soit un concept agrégat ou une prémice - à la limite la plus abstraite (telle qu'au sommet de la figure 3.1).

7. Rappelons qu'une prémice est un concept

Pour illustrer la problématique, considérons la troisième itération d’une conceptualisation hypothétique sur le transpositif d’entrée qui était présenté à la figure 4.6. On suppose également qu’aucun concept structurel n’existe initialement. Référez-vous à la figure 3.1 pour les liens d’abstractions/représentations entre les prémices. Si un concept-agrégat était créé systématiquement à chaque nouveau groupement, comme illustré à la figure 4.6 A, la conceptualisation inférait/détecterait les concepts  $C1\{L11, L31\}^o$ ,  $C2\{L31, L22\}^o$ ,  $L22$  aux noeuds  $\phi\{3, 1\}$ ,  $\phi\{3, 2\}$  et  $\phi\{3, 3\}$ ; puis les concepts  $C3\{C1, C2\}^o$ ,  $C4\{C2, L22\}^o$  aux noeuds  $\phi^{2,1}$  et  $\phi^{2,2}$ ; puis le concept  $C5\{C3, C4\}^o$  au noeud  $\phi^{1,1}$ . Cette stratégie apparaît, cependant, problématique.

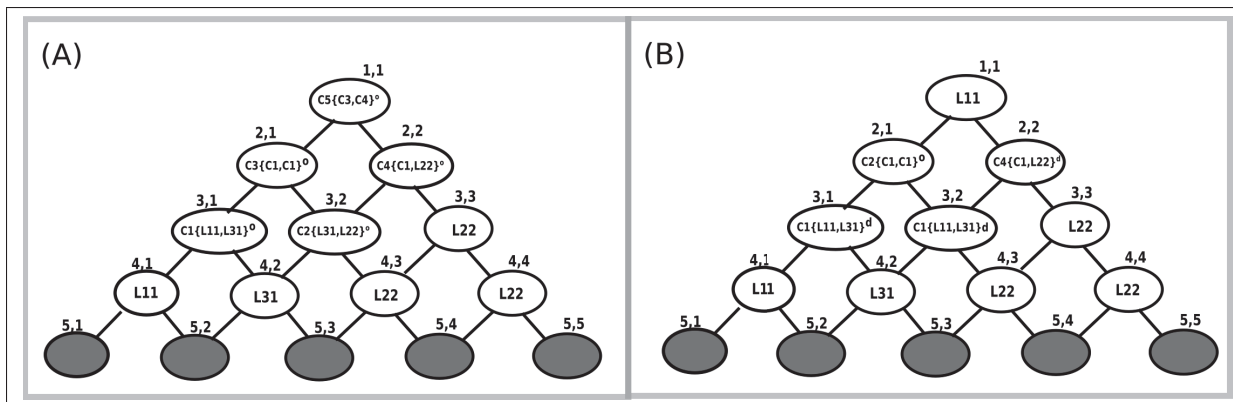


Figure 4.6 (A) Exemple de génération systématique de concepts structureux dans un transpositif hypothétique. Les concepts réalisent une représentation exacte. (B) Exemple concepts structureux possibles dans une troisième itération de conceptualisation

La question est de savoir : quels concepts devrait-on générer afin d’obtenir le meilleur compromis rigidité-plasticité ? Si un concept était généré pour chaque observation, le réseau conceptuel serait en mesure d’encoder de façon exacte l’ensemble de ses expériences, mais serait-ce utile ? En effet, une représentation exacte implique beaucoup de concepts qui sont spécifiques à chaque expérience. Comme les structures de ces concepts n’ont pas été observées au cours d’expériences antérieures, ces concepts spécifiques doivent être générés au moment de la conceptualisation pour ensuite ne plus servir. Il s’en suit une augmentation quasi linéaire de la taille du réseau, et inutile puisque la majeure partie des concepts générés, trop rigides, ne seront pas réutilisés. Cette complexité élevée du réseau, venant avec un manque de capacité de généralisation, est conséquemment un cas de *surentraînement*.

On cherche à générer des concepts apportant une ramification des concepts existants (on apporte une plus grande rigidité, capacité de différentiation), qui permettent néanmoins de représenter plusieurs observations (on conserve ainsi une certaine plasticité).

En utilisant des concepts un peu plus généraux, la conceptualisation du transpositif de la figure 4.6A pourrait plutôt donner lieu, par exemple, aux concepts nodaux présentés à la figure 4.6B. On remarque que chaque concept de la figure admet la composition exacte, c'est-à-dire :  $C1\{L11, L31\}^d$  admet  $\{L11, L31\}^o$  et  $\{L31, L22\}^o$  puisque L11 admet L22 et L31 admet L31 ;  $C4\{C1, L22\}^d$  admet  $\{C1, L22\}^o$  ; et L11 admet  $\{C2, C4\}^o$  puisque L11 admet C2 et C4.

Les concepts sont générés à partir de *ramifications* des concepts existants à des noeuds de leur arborescences, basées sur les compositions spécifiques observées à ces noeuds.

Considérant un ensemble de  $N^\Omega$  observations  $\Omega = \{\omega_1, \dots, \omega_{N^\Omega}\}$  liées par une abstraction  $A\{\Omega\}$ , le problème de *ramification* de  $A$  est de générer une représentation de  $A$  qui abstrait toujours, mais de façon plus rigide, l'ensemble des observations  $\Omega$ . Le concept ainsi généré est dit *ramification* de  $A$  basée sur les observations  $\Omega$ , et se note  $\Upsilon^{A|\Omega}$ .

Plus le nombre d'observations augmente, plus la ramification garde un niveau de plasticité élevé. À la limite, il deviendra impossible de générer un concept plus rigide que  $A$  lui-même et admettant toutes les observations. Dans ce cas, la tentative de ramification conduira à l'abstraction même :  $\Upsilon^{A|\Omega} = A(\Omega)$ .

L'évaluation des ramifications implique les sous-problèmes de : (i) former des groupes d'observations pertinents à la base des nouvelles ramifications ; (ii) évaluer une composition ordonnée et une composition désordonnée admettant un groupe d'observations ; (iii) générer la ramification, c'est-à-dire le concept-agrégat (ordonné ou désordonné).

### 4.3.1 Grouper les observations

Les observations sont groupées selon les contextes dans lesquels elles apparaissent. Le *groupe-contexte*, noté  $k(C^{ramif}|_x(C^{ref}))$ , est celui d'un concept à ramifier  $C^{ramif}$  situé à une position

$x$  de l'arborescence structurale d'un concept de référence  $C^{ref}$  (plus loin, on prend l'exemple où  $C^{ramif}$  est un oeil-de-chat et  $C^{ref}$  la tête d'un chat). Remarquons que cette position possède un niveau de plasticité dans le cas où des concepts structuraux désordonnés sont présents dans l'arborescence structurale de  $C^{ref}$  jusqu'à  $C^{ramif}$ . Lorsque le concept à ramifier correspond au concept de référence, la position  $x$  de l'observation est à sa racine  $r$ . On notera par exemple  $k(C234|r(C234))$ .

À la figure 4.7, on observe à gauche une image qui présente l'abstraction d'un chat  $C^{chat}$  à côté de celle d'un cheval  $C^{cheval}$ . Cette abstraction a pour composant une tête de chat  $C^{tete-chat}$  dans laquelle se trouve un oeil de chat  $C^{oeil-chat}$ . Comme cette abstraction admet les différentes formes de chat, elle sera détectée lors de la conceptualisation de chaque chat  $C^{chat1}$ ,  $C^{chat2}$  et  $C^{chat3}$  en haut de la figure.

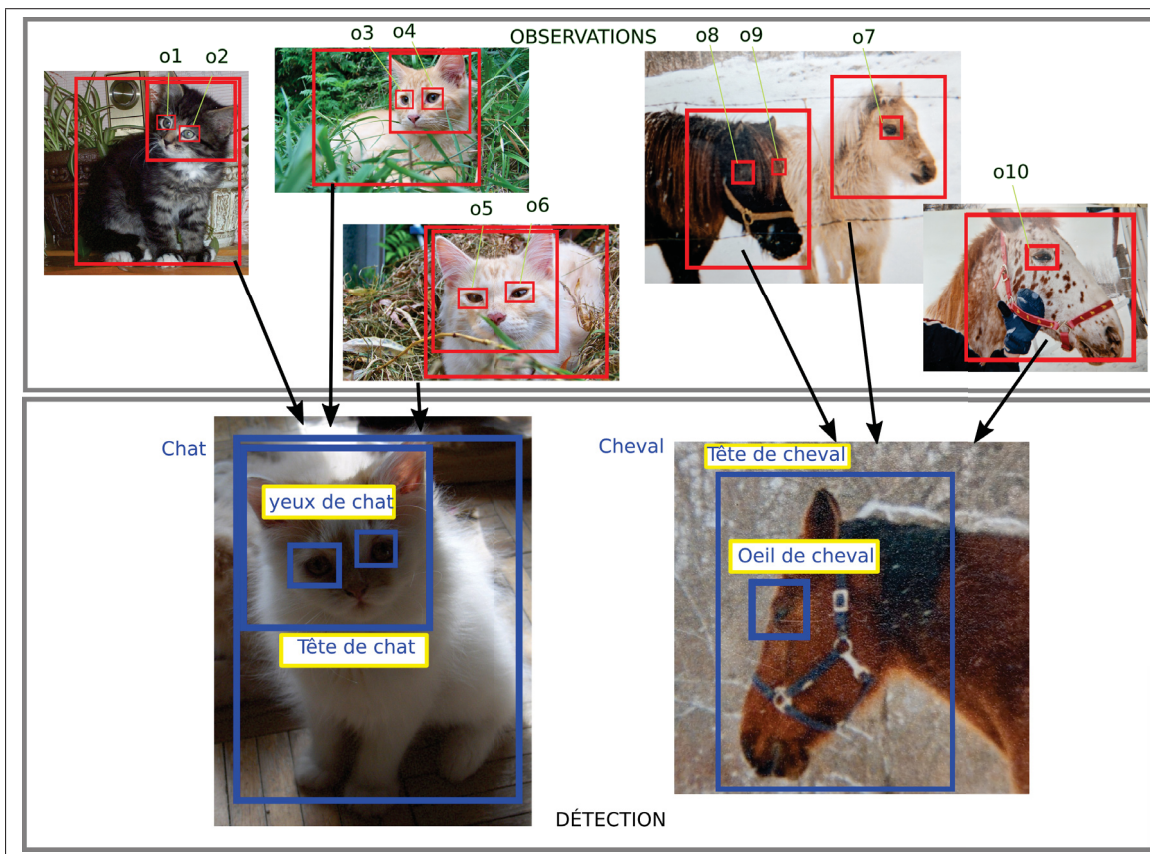


Figure 4.7 Abstraction figurée d'un chat (à gauche) et d'un cheval (à droite)

Observant différentes compositions d'yeux  $\{o_1, o_2, \dots, o_{10}\}$ , on voudrait ramifier les concepts existants liés à des yeux vers de nouvelles représentations plus descriptives, rigides. Le groupe des yeux  $k(C^{oeil}|x(C^{oeil})) = \{o_1, o_2, \dots, o_{10}\}$  inclut les observations liées au contexte d'un oeil, les groupes  $k(C^{oeil-chat}|x_G(C^{tete-chat})) = \{o_1, o_3, o_5\}$  et  $k(C^{oeil-chat}|x_D(C^{tete-chat})) = \{o_2, o_4, o_6\}$  aux contextes d'un oeil de chat observé respectivement à une position gauche  $x_G$  et droite  $x_D$  d'une tête de chat; et les groupes  $k(C^{oeil-chat}|x_G(C^{chat})) = \{o_1, o_3, o_5\}$  et  $k(C^{oeil-chat}|x_D(C^{chat})) = \{o_2, o_4, o_6\}$  aux contextes d'un oeil de chat observé respectivement à une position gauche  $x_G$  et droite  $x_D$  d'un chat. À côté, les groupes  $k(C^{oeil-cheval}|x_G(C^{tete-cheval})) = \{o_7, o_8, o_{10}\}$  et  $k(C^{oeil-cheval}|x_D(C^{tete-cheval})) = \{o_9\}$  correspondent aux contextes d'un oeil de cheval observé à une position gauche  $x_G$  et droite  $x_D$  d'un cheval. Ceci permettra de générer des ramifications pertinentes à chaque contexte afin de préciser respectivement la représentation d'un oeil en général, d'un oeil de chat et d'un oeil de cheval.

Puisque le processus de conceptualisation itératif détecte des concepts moins profonds d'abord, les observations d'une itération se rapportent aux groupes contextes de l'itération précédente.

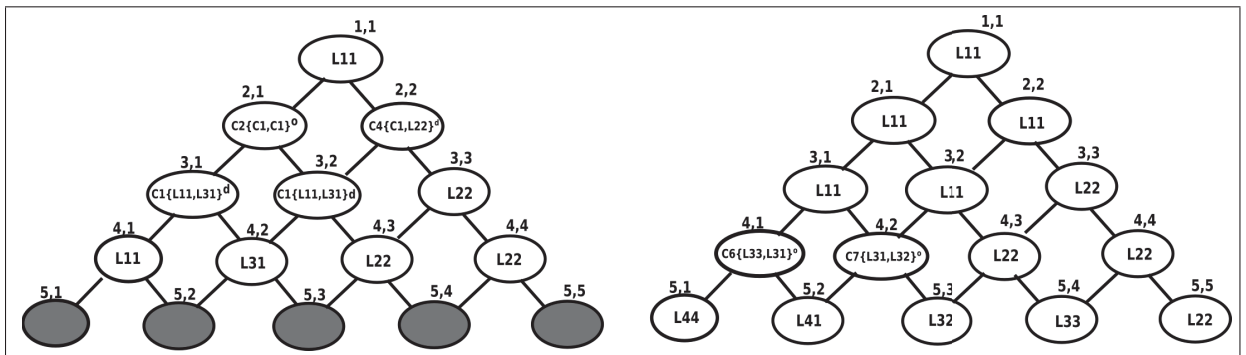


Figure 4.8 Troisième et quatrième itérations d'une conceptualisation hypothétique

Considérons à la figure 4.8 la quatrième itération de la conceptualisation de la figure 4.6B. On aurait lors de la conceptualisation de la quatrième profondeur :

- une observation  $\{L44, L41\}^o$  au noeud  $\psi^{4,1}$  dans les contextes  $k(L11|r(L11))$ ,  $k(L11|x_{2,1}(C1))$  et  $k(L11|x_{3,1}(C2))$

- une observation  $\{L41, L32\}^o$  au noeud  $\psi^{4,2}$  dans les contextes  $k(L31|r(L31))$   $k(L31|x_{2,2}(C1))$ ,  $k(L31|x_{2,1}(C1))$  et  $k(L31|x_{3,2}(C2))$ .
- *ainsi de suite*

Lors de la conceptualisation de la profondeur 3, nous aurions :

- une observation  $\{C6, C7\}^o$  au noeud  $\psi^{3,1}$  dans les contextes  $k(C1|r(C1))$  et  $k(C1|x_{1,1}(C2))$
- *ainsi de suite*

L'observation s'ajoute à chaque groupe-contexte où elle est observée. Lorsqu'un groupe contient un nombre suffisant d'observations, on peut procéder à la génération d'un nouveau concept admettant les différentes observations du groupe, lesquelles peuvent ensuite être retirées du groupe afin de laisser place aux prochaines observations. De façon pratique, le nombre d'observations utilisées pour générer un nouveau concept peut être fixe  $N^{genese}$ . Le choix de l'hyperparamètre ( $N^{genese}$ ) a un impact sur le rythme d'apprentissage : son augmentation diminue le nombre de concepts générés pour un même ensemble de données d'entraînement ; et il a un impact sur le degré d'abstraction des concepts générés : son augmentation entraîne la génération de concepts plus abstraits (ayant un pouvoir discriminant conséquemment plus faible) et à la limite une incapacité à générer de nouveaux concepts. Pour de futures recherches, il pourrait néanmoins être intéressant d'étudier la possibilité d'un choix automatique et dynamique de ce paramètre (sans qu'il soit forcément identique d'un groupement à l'autre).

### 4.3.2 Évaluer une composition ordonnée/désordonnée admettant un groupe d'observations

Soit un ensemble de  $N^\Omega$  observations  $\Omega = \{\omega_1, \dots, \omega_{N^\Omega}\}$ , où  $\omega_i$  est une composition ayant  $N^\omega$  concepts composants :  $\omega_i = \{C_j^i, j = 1 \dots N^\omega\}$ . Puisque les compositions sont de même type, ils ont tous le même nombre de composants (voir section 4.1) qu'il est possible de mettre en correspondance.

Pour définir une composition *ordonnée*  $\chi^{*o}$  qui admet l'ensemble des observations, les éléments des différentes compositions sont mis en correspondance *selon leur position*. Ainsi, chaque composant de position  $j, j = 1 \dots N^\omega$  de la composition ordonnée cible correspond à



l'abstraction commune (de distance minimale) des concepts observés à la position  $j$  des différentes observations (voir section I pour leur évaluation).

$$\chi^{*o} = \{A_j(C_j^i, i = 1 \cdots N^\Omega), j = 1 \cdots N^\omega\}^o \quad (4.1)$$

Cette dernière composition définira un nouveau concept ordonné. Par exemple, en se référant à la figure 3.1, si nous avons  $\Omega = \{\omega_1, \omega_2, \omega_3\}$  où  $\omega_1 = \{L41, L44\}$ ,  $\omega_2 = \{L42, L43\}$  et  $\omega_3 = \{L42, L42\}$  la composition ordonnée serait  $\chi^{*o} = \{L31, L22\}^o$  puisque  $L31$  est l'abstraction commune de  $\{L41, L42, L42\}$  et  $L22$  est l'abstraction commune de  $\{L44, L43, L42\}$ .

La composition désordonnée la plus rigide  $\chi^{*d}$  pour représenter l'ensemble des observations est obtenue en réalisant le *meilleur* appariement des éléments. Si on observe par exemple  $\{L41, L44\}^o$  et  $\{L43, L42\}^o$ , le meilleur appariement est obtenu en combinant  $L41$  à  $L42$ , puis  $L44$  à  $L43$ ; donnant lieu à la composition  $\{L31, L33\}^o$ . Cette dernière est, effectivement, plus rigide que la composition  $\{L21, L22\}^o$  obtenue avec les appariements  $\{L41, L43\}$  et  $\{L44, L42\}$ . On considère que le meilleur appariement est celui qui minimise la distance d'abstraction moyenne entre les éléments des observations et leurs abstractions communes. Par exemple, la distance d'abstraction moyenne du premier appariement serait de 1, car la distance entre  $L41$  et  $L31$ ,  $L42$  et  $L31$ ,  $L43$  et  $L33$ ,  $L44$  et  $L33$  est toujours de 1, La distance d'abstraction moyenne du deuxième appariement serait de 2, car la distance entre  $L41$  et  $L21$ ,  $L43$  et  $L21$ ,  $L44$  et  $L22$ ,  $L42$  et  $L22$  est toujours de 2.

### 4.3.3 Générer une ramification

Des concepts structuraux ordonnés et désordonnés sont générés pour représenter les compositions ordonnées et désordonnées obtenues à partir des observations tel que présenté plus haut. Si la composition désordonnée  $\chi^{*d}$  admet la composition ordonnée  $\chi^{*o}$ , alors les liens d'abstraction/représentation sont ajoutés aux concepts.

#### 4.4 Conceptualisation, en bref

Au cours de ce chapitre, j'ai d'abord présenté une méthode de « conceptualisation » permettant de détecter les concepts du réseau qui décrivent le mieux une expérience. Il a ensuite été présenté la *genèse conceptuelle* responsable de générer de nouveaux concepts à partir d'observations groupées aux ramifications des concepts existants. Le prochain chapitre positionnera la conceptualisation dans un processus *d'activation* des concepts du réseau selon des énergies qui décrivent leurs niveaux d'implication dans l'expérience en cours. Je proposerai dès lors une façon de réaliser la reconnaissance de forme, en considérant en particulier les énergies des concepts actifs.

## CHAPITRE 5

### ACTIVATION ET RECONNAISSANCE DE FORMES

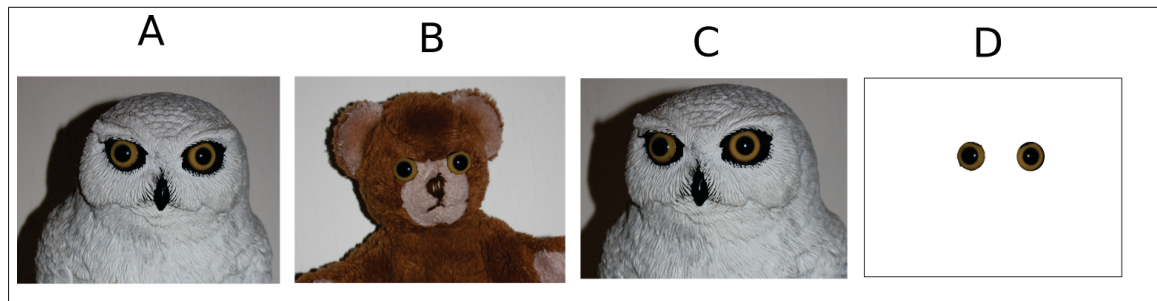


Figure 5.1 Deux yeux détectés lors des expériences A, B et C, puis lors d'une expérience D

#### 5.1 Activation

Au chapitre précédent, j'ai présenté le processus de conceptualisation, lequel permet de détecter les concepts les plus spécifiques du réseau pour décrire l'expérience observée. Ces concepts sont liés à leurs abstractions et aux contextes dans lesquels ils furent observés. De cette façon, si j'observe le chat Minex (concept spécifique), j'observe aussi un chat (une abstraction de Minex). Par ailleurs, si Minex se trouve habituellement sur un coussin (contexte d'observation), le fait d'observer Minex peut évoquer l'image du coussin ; mais nous savons que Minex peut également se trouver dans une ruelle, à la cuisine, etc. Le degré avec lequel Minex évoque un contexte particulier dépend de l'importance relative de ce contexte par rapport aux autres contextes où Minex est observé. L'expérience est donc décrite de façon spécifique avec les concepts détectés ; puis elle est liée aux expériences antérieures par *évocation*.

Ainsi, nous pouvons considérer une sorte de « transfert d'énergie » descriptive partant des concepts détectés vers les concepts évoqués, où l'*énergie* d'un concept décrit le degré avec lequel il est impliqué dans l'expérience en cours.

Pour donner un autre exemple, les mêmes yeux sont observés à la figure 5.1 aux expériences A, B et C. Si maintenant on observe encore les yeux (expérience D), nous feront-ils penser à la tête du hibou ou à celle de l'ourson ? Assumant que l'évocation d'une tête ou l'autre est fort possible, une autre question est : évoqueront-ils davantage une tête en particulier ? Remarquons d'abord que cela dépend de plusieurs facteurs, dont (1) l'occurrence relative de ces expériences, c'est-à-dire le nombre de fois que (ou la durée avec laquelle) ces yeux ont été observés avec le hibou versus avec l'ourson, et (2) l'importance relative de chaque expérience liée, entre autres, à la quantité d'expériences antérieures qu'elle évoque et (3) de la capacité à représenter chaque expérience à travers le réseau conceptuel.

Si on s'en tient à l'occurrence, l'énergie d'évocation de la tête du hibou serait le double de celle de l'ourson, puisque la tête du hibou occupe deux fois plus d'images. Par contre, si l'expérience de l'ourson évoque une grande quantité d'expériences antérieures analogues (films, imaginaire), cette dernière peut devenir plus marquante ; de sorte que l'image des yeux D évoque davantage la tête de l'ourson bien qu'elle fut observée qu'une seule fois sur trois. Cette observation renforce l'idée qu'une expérience phénoménologique ne se caractérise pas seulement par les concepts détectés, mais aussi par ceux qu'elle évoque.

Ensuite, pour pouvoir évoquer l'ourson ou le hibou, il faut effectivement que chacune de ces formes d'expériences soit encodée dans le réseau. La facilité avec laquelle il est possible d'encoder une expérience renforce sa plausibilité d'être évoquée par la suite. Ainsi est-il plus plausible d'avoir en tête une pièce musicale simple (le *vers d'oreille*) que complexe et original.

En résumé, l'*activation* inclut : le (1) le processus de détection de concepts à la conceptualisation et (2) le processus d'évocation de concepts suivant la conceptualisation. Les concepts dits *actifs* sont ceux qui caractérisent l'expérience. Ils comprennent les concepts détectés et les concepts évoqués. Chaque concept actif caractérise l'expérience à un certain degré, ou en d'autres termes : avec une certaine énergie. Nous présentons ci-dessous le mécanisme de consolidation, qui renforce le lien entre un concept et son contexte d'observation ; puis le processus d'évocation.

### 5.1.1 Consolidation des concepts actifs

Lors de la conceptualisation, des concepts sont détectés dans leurs contextes, ici représentés par les concepts qui les agrègent à des noeuds parents du transpositif. Or, chaque relation composant/agrégat peut être renforcée, *consolidée*, de sorte à augmenter la proportion de l'énergie transmise à l'agregat lors d'éventuelles évocations.

La *consolidation* est nécessaire lorsqu'on désire que la force des liens composants/agrégats s'adapte à l'expérimentation du système, ce qui est implicitement le cas par exemple en mode génératif où la structure du réseau est en construction.

Ayant un concept agrégat  $\psi_x$  activé par une énergie  $e$ , la force  $f_p(\chi_n \rightarrow \psi_x)$  du lien d'agrégation de chaque composant  $\chi_n$  envers ce concept, parmi  $N$  composants, sera augmentée de  $e/N$ . Désignant  $f(\chi_n \rightarrow \psi_x)$  la valeur de la force après consolidation, on aura simplement :

$$f(\chi_n \rightarrow \psi_x) = f_p(\chi_n \rightarrow \psi_x) + e/N \quad (5.1)$$

La figure 5.2 présente un exemple de consolidation d'un concept  $\psi_x$  ayant les composants  $\chi_1$  et  $\chi_2$ . Chaque composant renforce son lien d'agrégation vers  $\psi_x$  avec la moitié de l'énergie de  $\psi_x$ . La consolidation des concepts détectés peut être effectuée juste après l'ensemble de la conceptualisation (incluant les multiples itérations potentielles d'une profondeur de départ à l'autre).

### 5.1.2 Évocation

Le processus d'évocation suit la conceptualisation en distribuant l'énergie des concepts détectés vers leurs abstractions et leurs agrégats. Les concepts actifs correspondent à tous ceux ayant une énergie d'activation supérieure à un seuil pratique.

La méthode d'évocation présentée ici s'appuie sur l'idée que l'énergie provient des concepts détectés, puis est diffusée à travers le réseau selon la force des liens directionnels entre les

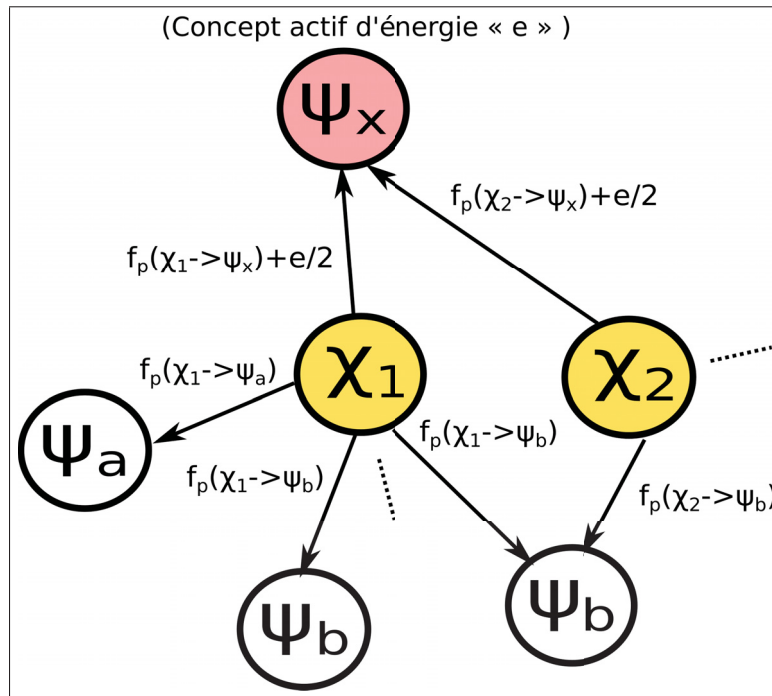


Figure 5.2 Consolidation d'un concept  $\psi_x$  ayant pour composants les concepts  $\chi_1$  et  $\chi_2$ . L'énergie du concept  $\psi_x$  enforcit les liens d'agrégation de  $\chi_1$  vers  $\psi_x$  et de  $\chi_2$  vers  $\psi_x$

concepts : représentants vers abstractions, composants vers agrégats. Aucune énergie n'est créée lors de la diffusion. Cette méthode a permis les meilleurs résultats lors des tests pratiques au chapitre 6<sup>8</sup>, mais pourrait être étudiée à nouveau et potentiellement améliorée dans de futures recherches.

Lors de l'initialisation du processus, on attribue une énergie initiale à chaque concept détecté, que l'on distribue ensuite en parcourant (de façon récursive ou itérative) chaque concept vers ses abstractions et agrégats.

L'énergie initiale dépend de la profondeur du noeud du transpositif sur lequel le concept est détecté. Assumant que la racine du transpositif correspond à la profondeur 0, nous utilisons la

8. Plusieurs autres méthodes ont été développées et abandonnées.

formule empirique :

$$e_{activ} = \frac{1}{1 + profondeur} \quad (5.2)$$

Si le même concept est détecté à des noeuds ayant différentes profondeurs, l'énergie maximale des deux détections est retenue pour ce concept.

Le processus d'évocation de l'énergie  $e_{activ}$  d'un concept  $C$  implique :

1. De diviser l'énergie  $e_{activ}$  entre : l'énergie emmagasinée par le concept ( $e_C$ ); l'énergie transmise aux abstractions ( $e_A$ ), l'énergie transmise aux agrégats ( $e_\psi$ ) et l'énergie rediffusée aux composants ( $e_\chi$ ) de sorte que

$$e_{activ} = e_C + e_A + e_\psi + e_\chi \quad (5.3)$$

Il est possible de distribuer l'énergie  $e_{activ}$  avec une proportion fixe pour  $e_C$ ,  $e_A$ ,  $e_\psi$  et  $e_\chi$ <sup>9</sup>.

2. Propager l'activation vers chaque abstraction  $A_n, n = 1 \dots N^A$  du concept  $C$ . Chaque abstraction sera activée avec une énergie

$$e(A_n) = e_A / N^A. \quad (5.4)$$

3. Propager l'activation vers chaque agrégat  $\psi_n, n = 1 \dots N^\psi$  du concept  $C$ . L'énergie attribuée à chaque agrégat est proportionnelle à la force du lien d'agrégation :

$$e(\psi_n) = \frac{f(C \rightarrow \psi_n)}{\sum_{i=1}^{N^\psi} f(C \rightarrow \psi_i)} \times e_\psi \quad (5.5)$$

4. Rediffuser l'énergie du concept  $C$  vers ses composants. L'énergie distribuée à un composant  $i$  est proportionnelle à la force de ce composant vers  $C$ ,  $f(\chi_i \rightarrow C)$ , relativement à la force totale de la composition,  $\sum_j^{N^\chi} f(\chi_j \rightarrow C)$  avec  $N^\chi$  le nombre de composants :

$$e_\chi(i) = \frac{f(\chi_i \rightarrow C)}{\sum_j^{N^\chi} f(\chi_j \rightarrow C)} e_C \quad (5.6)$$

---

9. Au chapitre 6, j'utilise  $e_C/e_{activ} = 0.30$ ,  $e_\psi/e_{activ} = 0.15$ ,  $e_\chi/e_{activ} = 0.05$  et  $e_A/e_{activ} = 0.50$ .

L'énergie est ainsi davantage rediffusée vers les composants spécifiques au concept C.

La propagation est contrainte par les deux règles suivantes :

1. Afin de limiter la quantité de traitements, on peut attribuer un seuil minimal sur la valeur de l'énergie propagée : l'énergie inférieure à ce seuil ne sera pas propagée. Dans notre implémentation, le seuil correspond à un pourcentage de l'énergie du concept destinataire (par exemple, l'énergie transférée doit être augmentée de celle du concept destinataire de plus de 5%).
2. Lorsqu'un concept a atteint une activation complète (unitaire), il n'est plus apte à recevoir et propager une quantité d'énergie supplémentaire (l'énergie supplémentaire reçue est donc perdue).

## 5.2 Reconnaissance de formes

Si le terme *forme* désigne une unité caractéristique ayant une certaine plasticité (déformation admissible) et rigidité (caractéristiques distinctives des autres formes), un concept peut être vu comme une forme. L'activation serait ainsi, en soi, un processus de *reconnaissance de formes* : celles détectées et évoquées.

Cependant, le concept, tel que défini à l'introduction, est avant tout une unité permettant d'encoder une expérience phénoménologique. Tel que généré par le processus décrit à la section 4.3, le concept n'a pas d'utilité pratique en soi.

Or, les *formes* que nous désirons être *reconnues* par un système d'apprentissage machine existent, typiquement, pour leurs utilités pratiques, pour un objectif en particulier. Une *forme pratique* peut rattacher de nombreuses expériences phénoménologiques ayant peu de chose en commun. Contrairement au concept, la *forme pratique* a une composition indéfinie, et peut admettre de nouveaux types d'expériences. Par conséquent, une forme pratique n'existe pas, typiquement, en tant que concept distinctif dans le réseau. Plutôt, elle se rattache, à différents degrés, à un ensemble de concepts.



Considérons par exemple un « chat ». Lorsqu'on pense à un chat, on peut penser, plus spécifiquement, à différentes images de chats, différents sons que peut émettre un chat, différentes actions de chats : un chat qui attrape une souris, un chat qui dort, ainsi de suite. Les expériences phénoménologiques visuelles, sonores, et même tactiles que peut évoquer le mot « chat » sont sans bornes, et varient selon les expériences propres à chaque individu. Maintenant, qu'est-ce qui rattache la forme « chat » à l'ensemble de ces expériences ? Serait-ce simplement l'étiquette « chat » ? Lorsqu'on voit un chat, on pense au fait qu'il s'agit d'un « chat ». Cette classification peut être aussi bien visuelle (le mot « chat » écrit) qu'auditive (le mot « chat » prononcé). Elle ajoute à l'expérience de la vision du chat, l'expérience de sa classe, qu'elle soit visuelle, auditive, ou autre. La forme « chat » existe pour un usage pratique : en particulier, celle de désigner ce type d'animal dans une communication avec d'autres individus.

Puisque l'expérience de la vision du chat et l'expérience de sa classe ont lieu en même temps, elles peuvent, dans une première approche, être *conceptualisées* en des concepts agrégats hétérogènes. Ces agrégats seront composés, à la fois, de concepts encodant l'expérience visuelle du chat et ceux encodant la classe. Par la suite, si le chat est revu, le processus d'évocation décrit à la section précédente activera avec une certaine énergie ces agrégats qui incluent des concepts de classes, lesquels seront à leur tour activés via l'énergie rediffusée aux composants.

La classification d'une expérience peut ainsi être réalisée en comparant les énergies des différents concepts de classes lors de l'activation du réseau.

L'approche plus haut exige cependant de définir un algorithme efficace pour générer des concepts hétérogènes pertinents en nombre limité. Par exemple, si un concept hétérogène agrège l'ensemble des concepts détectés, incluant une prémisse-étiquette pour la classe, la probabilité que ce concept soit observé plus qu'une fois est très mince. En pratique, un concept hétérogène sera donc créé à chaque nouvelle expérience. On peut alors tenter de générer le concept hétérogène en considérant les concepts détectés lors d'une série d'expériences de sorte à augmenter leur capacité de généralisation, et de ce fait leur réutilisabilité. Ces deux approches ont été explorées lors de nos expériences.

Pour limiter la lourdeur du réseau et optimiser les ressources, une seconde approche a également été explorée : celle de considérer chaque concept-étiquette en tant qu'agrégat des concepts détectés ; et de consolider les liens entre les concepts détectés et le concept-étiquette à partir de l'énergie de ce dernier. Dans cette approche, le concept-étiquette est typiquement une prémice qui n'a, pourtant, aucun composant prenant part à sa définition (la prémice ne mémorise aucun composant). Ce système reflète la nature « floue » du concept-étiquette où, par exemple, différentes observations de composants de chats (oreille, queue, miaulement) peuvent s'enregistrer comme étant, effectivement des « composants de chat » en ajoutant le concept-étiquette « chat » dans la liste de leurs agrégats ; sans pour autant prendre part à la définition d'un « chat ». Néanmoins, la présence de ces liens des composants vers l'étiquette fera en sorte que l'observation de ces composants évoquera le concept de « chat ».

### 5.3 L'activation, en bref

Nous avons vu dans ce chapitre le processus d'activation, où les concepts sont *activés* selon une énergie qui suit leur degré d'implication dans l'expérience courante. Pendant que les concepts *détectés* via la conceptualisation sont ceux qui représentent l'expérience en cours de la façon la plus spécifique ; les concepts *évoqués* sont ceux implicitement activés considérant les liens d'abstraction (par exemple : un félin est implicitement détecté si un chat est détecté) et considérant les expériences antérieures via les liens d'agrégation (un poil, qui fut observé sur un chat, évoquera le chat).

Puisque les concepts décrivent des types d'expériences avec un certain niveau d'abstraction (qui implique une certaine capacité de généralisation), ils peuvent être utilisés pour des fins de reconnaissance de formes. La façon d'effectuer la reconnaissance de formes à partir du réseau conceptuel est une problématique de recherche ouverte. J'ai proposé quelques solutions, dont celle d'ajouter des *prémices-étiquettes* qui apparaîtraient dans les listes d'agrégats des concepts détectés. Le lien d'agrégation d'un concept détecté vers une prémice étiquette serait donc enforcie, avec l'énergie du concept détecté, à chaque expérience d'entraînement impliquant la classe liée à la prémice étiquette. Nous testerons cette stratégie tout en étudiant les

propriétés des réseaux conceptuels au prochain chapitre (chapitre 6) avec des applications de reconnaissance de chiffres écrits et prononcés, ainsi qu'avec l'application de reconnaissance du locuteur.



## CHAPITRE 6

### PREUVE DE CONCEPT

Ce chapitre explore l'utilité des réseaux conceptuels pour des applications classiques de reconnaissance de formes, soit la reconnaissance de chiffres écrits représentés sur des images, la reconnaissance des chiffres prononcés oralement et la reconnaissance du locuteur. Les exemples qui sont présentés sous-tendent trois objectifs plus particuliers :

1. Vérifier la praticabilité de la reconnaissance de formes à partir des réseaux conceptuels ;
2. Présenter les propriétés des réseaux conceptuels ;
3. Exposer les limites pratiques de la théorie proposée et introduire les prochaines étapes de recherche.

#### **Avant propos**

Étant donné les objectifs cités plus haut, il est important d'accepter que ce chapitre ne vise pas à proposer une approche révolutionnaire pour la reconnaissance de chiffres écrits ou prononcés ; contrairement à ce dont on pourrait s'attendre d'une recherche qui serait plus appliquée. Les performances des systèmes de reconnaissance de formes dépendent de nombreux paramètres, incluant notamment : la configuration des systèmes (par exemple pour un réseau de neurones : le nombre de couches, de neurones sur chaque couche, de type des fonctions d'activation, la façon dont sont connectés les neurones), les caractéristiques utilisées en entrées, les données d'entraînement (leur quantité, leur type et la façon qu'elles sont utilisées pour l'entraînement) et des particularités du problème de classification ou régression. La recherche contemporaine en intelligence artificielle s'intéresse en particulier, pour chaque application d'intérêt, à optimiser les paramètres susmentionnés afin d'optimiser les performances. Ce travail d'optimisation est en aval de la réalisation de systèmes d'apprentissage théoriquement viables.

Développer un nouveau système pour l'apprentissage machine est un travail de longue haleine. Pour chacun des systèmes d'apprentissage machine précédents (tels que les SVM, réseaux de

neurones, arbres de décisions...), de nombreuses recherches s'étalant sur des décennies ont été nécessaires avant d'observer les performances actuelles (Bishop & Nasrabadi, 2006), révisant et améliorant les fondements théoriques ainsi que la façon d'exploiter ces systèmes dans les applications cibles. Le système proposé dans cette thèse ne fera pas, en toute évidence, exception à cette exigence de recherche. Ainsi, les performances de base obtenues dans ce chapitre servent pour l'instant qu'à valider la plausibilité de la théorie et donner un avant-goût des capacités des réseaux conceptuels afin de motiver de futures recherches théoriques, ou encore appliquées dans lesquelles des comparaisons de performances avec d'autres systèmes seront adéquates.

Nous réalisons la preuve de concept avec les applications de reconnaissance : (i) de chiffres écrits (via des images); (ii) de chiffres prononcés (via des fichiers audio); (iii) et du locuteur ayant prononcé le chiffre.

### 6.1 Reconnaissance d'images : chiffres écrits

Nous utilisons un réseau conceptuel pour encoder et classifier des images de chiffres pouvant prendre différentes valeurs de 0 jusqu'à 9.

L'expérience d'un chiffre est le type «  $v^{chiffre}$  », où chaque prémice encode un chiffre spécifique, comme illustré à la figure 6.1 où la racine de la hiérarchie d'abstraction des prémices «*chiffre*» admet tous les chiffres.

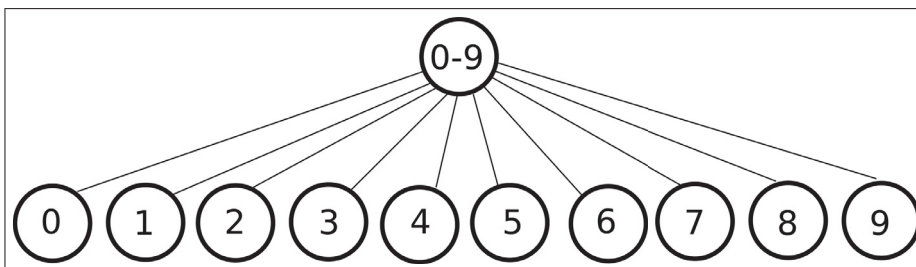


Figure 6.1 Hiérarchie d'abstraction de prémices représentant des chiffres

La reconnaissance de chiffres est d'abord effectuée à partir d'une image, soit une grille de pixels dont l'intensité varie sur une échelle 0-1 (noir-blanc). Les plages d'intensité admissibles sont encodées par des prémices d'intensité (type  $v^{intensité}$ ) ayant une hiérarchie d'abstraction comme celle présentée à la figure 3.1. Chaque feuille du transpositif d'entrée  $\Gamma^{intensité}$  correspond à un pixel de l'image (comme illustré à la figure 4.2). Avant la conceptualisation, chaque prémice nodale est ainsi actualisée selon l'intensité du pixel.

### 6.1.1 Entraînement

Différentes façons d'entraîner le réseau peuvent être développées et étudiées.

Pour chaque donnée d'entraînement (c'est-à-dire chaque illustration de chiffre), il faut effectuer une conceptualisation homogène et générative sur le transpositif d'entrée actualisé avec l'image.

Pour que le réseau puisse classer les chiffres, le problème est de relier les concepts liés à l'image (de type  $v^{intensité}$ ) aux prémices de l'étiquette (de type  $v^{chiffre}$ ), de sorte que ces dernières soient activées lors de la conceptualisation avec une énergie cohérente avec la nature du chiffre sur l'image. Voici quelques stratégies envisagées à l'heure actuelle :

1. *1 concept hétérogène pour chaque donnée d'apprentissage.* Pour chaque donnée d'apprentissage, nous créons un concept hétérogène qui contient la prémice du chiffre et l'ensemble des concepts racines détectés<sup>10</sup> lors de la conceptualisation. En prédiction, l'étiquette retenue est celle dont la prémice-étiquette a l'énergie maximale.
2. *1 concept hétérogène pour chaque chiffre.* Deux sous-variantes :
  - a. *Toute détection* : un concept hétérogène agrège la prémice d'un chiffre et l'ensemble des concepts racines détectés lors des expériences de ce chiffre.
  - b. *Toute détection discriminante* : Un concept hétérogène agrège la prémice d'un chiffre et l'ensemble des concepts racines détectés majoritairement lors des expériences de ce chiffre. *Majoritairement* : on établit un seuil d'exclusivité. Par exemple, un concept

---

10. *Concept racine* : soit un concept qui n'est pas agrégé par l'agrégat d'un noeud parent.

détecté à plus de 50% lors d'expériences du chiffre 7 sera dans la composition hétérogène du chiffre 7.

3. *Prémice-étiquette agrégat*. Dans cette stratégie, la prémice-étiquette définissant la classe de l'expérience correspond à un agrégat de chaque concept détecté lors de la consolidation. Ainsi un lien d'agrégation est créé ou renforcé de chaque concept détecté vers la prémice étiquette, avec l'énergie d'activation du concept détecté. En prédiction, l'étiquette retenue est celle dont la prémice-étiquette a l'énergie, normalisée par le nombre de consolidations ayant eu lieu pour cette étiquette, maximale.

La première stratégie présente l'inconvénient de créer potentiellement de nombreux concepts hétérogènes, ce qui alourdit le réseau et ralentit tous les traitements sur le réseau. La stratégie *1 concept hétérogène pour chaque chiffre* a l'avantage de créer peu de concepts hétérogènes, et avec moins de composants (plus particulièrement pour l'option *toute détection discriminante*). La dernière stratégie *Prémice-étiquette agrégat* ne crée pas de concept hétérogène, et est en somme la plus légère. Les performances obtenues avec cette dernière sont similaires aux précédentes pour les corpus maison et supérieures avec MNIST. C'est donc la stratégie utilisée par défaut pour la suite. Ces quelques stratégies donnent un aperçu de la pluralité de solutions envisageables, qu'il serait pertinent d'étudier lors de recherches dédiées.

### 6.1.2 Prédiction

La prédiction d'une image consiste à :

1. Effectuer la conceptualisation en mode non génératif
2. Effectuer l'évocation, c'est-à-dire propager l'énergie des concepts détectés vers les autres concepts du réseau.
3. Ensuite, on évalue quelle prémice de chiffres a le plus d'énergie : il s'agit de la classe retenue.



### 6.1.3 Tests sur des corpus maison

Afin de valider la praticabilité de l'approche proposée, nous effectuons d'abord quelques tests sur des petits corpus maison. Quatre valeurs d'intensité (également dispersée entre 0 et 1) sont considérées à la base de la hiérarchie de prémices d'intensité. Pour chaque expérience, la première moitié des données est utilisée pour l'entraînement et la seconde moitié pour la prédiction.

Le tableau 6.1 résume certaines expériences réalisées sur chaque corpus, en présentant : une description du corpus, la profondeur maximale de conceptualisation (Prof); le nombre d'observations utilisées pour la génération de concepts ( $N^{genese}$ ); le nombre de prédictions correctes (Correct); le nombre d'entraînements (Entr.); le nombre de concepts (Concepts), de liens d'abstraction (Abstr.) et de liens d'agrégation (Agr.) générés lors de l'entraînement.

Tableau 6.1 Résultats de prédiction pour la reconnaissance d'images de corpus maison

Corpus	Prof	$N^{genese}$	Correct	Entr.	Concepts	Abstr.	Agr.
18 motifs translétés {-, +,  }, (5x5)	5	1	7/9	1	296	449	735
			3/9	2	402	608	1107
			6/9	1	165	235	96
			7/9	1	223	344	167
			9/9	1	278	416	235
32 chiffres 0, 1, 2, 3, (7x7)	7	2	16/16	1	982	1425	1024
			12/16	2	1490	2162	2005
			15/16	3	1850	2698	2848
			16/16	4	2103	3078	3506
			15/16	5	2296	3355	4048
48 chiffres 0, 8, 1, 7, (10x10)	10	2	21/24	1	4282	6150	6371
			22/24	2	6651	9540	11692
			23/24	3	8807	12621	17126
			24/24	4	10882	15655	22864
			22/24	5	12588	18123	28037
32 chiffres 0, 8, 1, 7 (14x14)	14	2	14/16	1	12038	17105	20930
			13/16	2	20320	28369	38277
			11/16	3	27606	38445	55509
			13/16	4	34803	48341	72937
			13/16	5	41897	58189	91599
	12	2	14/16	1	7152	10063	11852
			12/16	1	11812	16520	21190

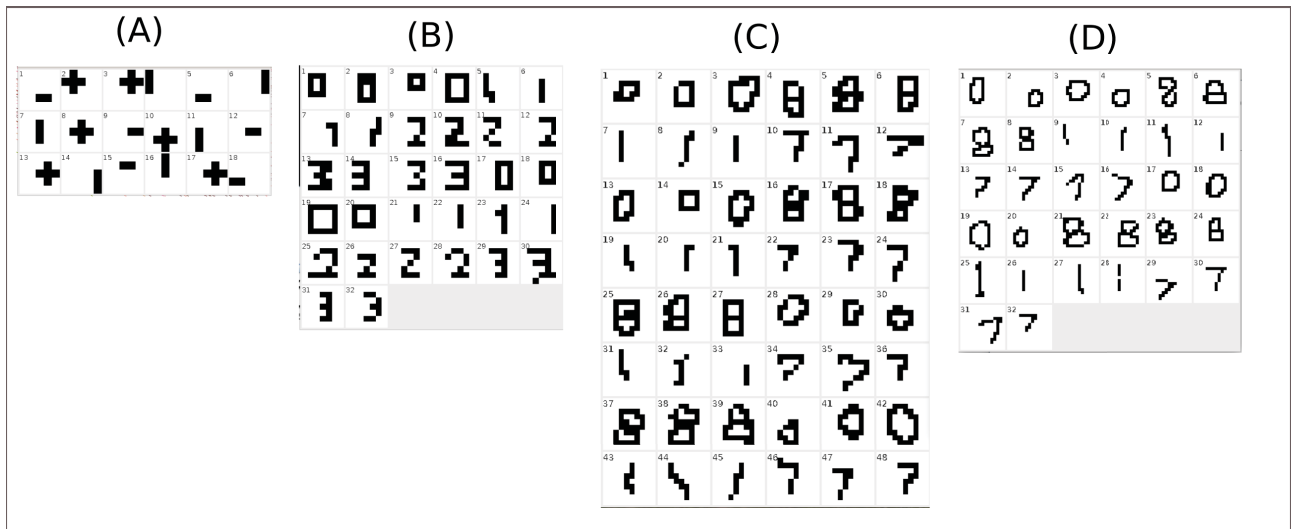


Figure 6.2 Corpus maison utilisé pour la reconnaissance des chiffres écrits. A : Premier corpus de test : 18 images de 5x5 pixels représentant trois symboles à différentes positions. B : Chiffres 0, 1, 2 et 3 sur 32 images de 7x7 pixels. C : Chiffres 0, 1, 7 et 8 sur 48 images de 10x10 pixels. D. Les nombres 0, 1, 7 et 8 sur 32 images de 14x14 pixels

### 6.1.3.1 Motifs translétés

Le premier test a été réalisé sur les 18 images de la figure 6.2A, qui inclut trois motifs simples dans une grille de 5 par 5 pixels : un trait horizontal de 3 pixels, un trait vertical de 3 pixels et un « plus ». Les motifs de même type sont identiques, mais translétés d'une image à l'autre. L'entraînement est réalisé sur les 9 premiers motifs et la prédiction sur les 9 suivants.

Si nous voulons une représentation exacte avec  $N^{genesis} = 1$ , la prédiction est de 7/9 après un entraînement ; de 3/9 avec deux entraînements. Les concepts générés de plus grande profondeur encodent précisément la position du motif dans la zone, et n'admettent donc pas la translation du motif. Par conséquent, les nouvelles expériences activent peu de structures de réseau existantes. Avec la technique « 1 concept hétérogène pour chaque chiffre - toute détection », on a 6/9 bonnes détections au premier entraînement, 7/9 au 2e puis 8/9 pour les suivantes. Si les résultats sont préférables pour cet exemple, ils restent cependant semblables pour les exemples suivants, et nous présentons donc, pour simplifier, que les résultats avec la technique basée sur la prémice-étiquette agrégat.

En optant plutôt pour un réseau avec une certaine plasticité en passant de  $N^{genesis} = 1$  à  $N^{genesis} = 2$ , la prédiction est de 6/9 au premier entraînement, 7/9 au deuxième et au troisième, puis 9/9 au suivant.

La figure 6.3A donne un aperçu en trois dimensions du réseau après l'entraînement. La quantité de concepts et de liens entre ceux-ci (liens d'agrégation ou d'abstraction) forme un « pain » d'information qu'il serait difficile de décortiquer de façon analytique même en explicitant les détails de la figure. Cette difficulté est encore plus drastique pour un réseau construit à partir d'images à plus haute résolution, tel que celui de la section 6.1.3.4, illustré à la figure 6.3D. L'utilité d'une telle tentative de représentation semble davantage d'ordre artistique que pratique. Il s'agit ici d'une distinction importante par rapport aux réseaux de neurones, dont la structure fixée a priori (typiquement par le concepteur) reste relativement simple et facile à décortiquer.

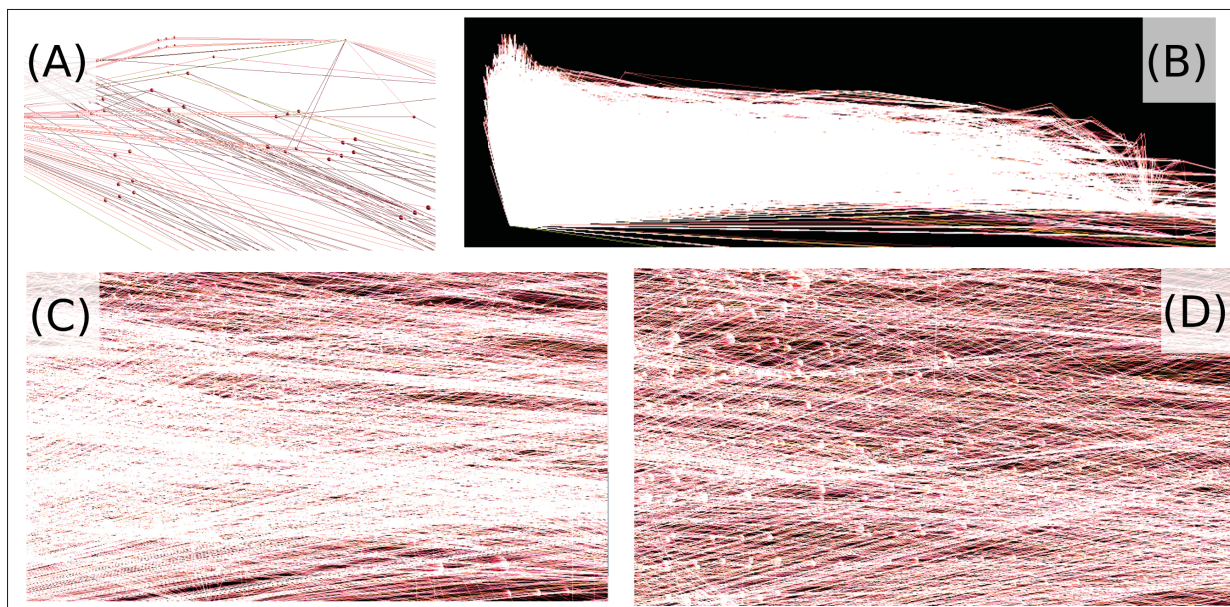


Figure 6.3 Aperçu du réseau conceptuel généré par l'entraînement sur un corpus. A : Corpus 6.2A. B : Corpus D (avec 4949 concepts). C/D : Agrandissement 2x/4x du corpus D

Ce premier exemple introduit deux propriétés importantes d'un réseau conceptuel :

1. Avec un faible nombre d'observations par concept généré (ici  $N^{genese} = 2$ ), l'entraînement génère beaucoup d'information avec peu de données d'entraînement ;
2. La complexité de la structure du réseau inféré suit celle des expériences phénoménologiques encodées. Ce faisant, le réseau devient rapidement trop complexe pour pouvoir être analysée par inspection, tel que sur une représentation graphique.

### 6.1.3.2 Reconnaissance des chiffres 0, 1, 2, 3 sur 7x7 pixels

Nous essayons maintenant de reconnaître les chiffres 0, 1, 2 et 3 réalisés à main levée dans une grille de 7 par 7 pixels, lesquels sont présentés à la figure 6.2B. Par rapport à l'exemple précédent, le transpositif d'entrée pour la grille de pixels possède maintenant deux couches de profondeur additionnelles. De plus, les chiffres sont réalisés à main levée dans la grille et ne sont donc pas identiques. Les 16 premières images sont utilisées pour l'entraînement et les 16 dernières pour la prédiction.

La prédiction donne un résultat de 16/16 bonnes classifications. Lors du réentraînement du réseau, la performance fluctue entre 12/16 et 16/16 d'un entraînement à l'autre. Les performances sont meilleures avec un seul entraînement, où le réseau présente une bonne capacité de discrimination/généralisation avec un nombre minimal d'unités.

Cet exemple nous permet de constater une autre propriété importante de l'utilisation des réseaux conceptuels : *la capacité de généralisation nécessite peu d'exemples*. Si nous examinons les images de la figure 6.2B, la variabilité des figures présentées est importante : les images sont déformées de l'une à l'autre, de tailles différentes, d'épaisseurs de lignes variables, de positions différentes. Avec seulement 16 exemples (4 de chaque chiffre), le réseau parvient à construire une représentation complexe de 982 concepts, ayant une plasticité suffisante pour la reconnaissance de nouvelles représentations de chiffres.

### 6.1.3.3 Chiffres 0, 8, 1, 7 sur 10x10 pixels

Dans cet exemple, nous testons le système pour différencier les chiffres 0, 8, 1 et 7. Ce choix de chiffres nous permet de vérifier si le système confond 0 et 8, puis 1 et 7. Les 32 images à main levée de 10x10 pixels sont présentées dans la figure 6.2C. Les 24 premières images sont utilisées pour l'entraînement et les 24 dernières pour la prédiction.

Après le premier entraînement, nous avons 21/24 succès, donc trois erreurs : l'image 26 (chiffre "8") est prise pour un "0" ; l'image 40 (chiffre "0") est prise pour un "8" ; l'image 44 (chiffre "1") est prise pour un "7". Il y a en effet une certaine confusion entre "0" et "8" puis entre "1" et "7". Réentraîner le réseau avec les mêmes exemples permet de générer des concepts plus profonds, ce qui améliore les performances respectivement de 22/24, 23/24 et 24/24 d'un entraînement à l'autre ; pour ensuite redescendre à 22/24 au 5e entraînement.

### 6.1.3.4 Reconnaissance des chiffres 0, 8, 1, 7 sur 14x14 pixels

Dans cet exemple, nous considérons à nouveau le problème de classifier correctement les chiffres 0, 1, 7 et 8, mais en utilisant des images de 14 par 14 pixels avec le corpus de la figure 6.2D. La prédiction donne un score de 14/16 et ne s'améliore pas avec la poursuite de l'entraînement. En diminuant la profondeur maximale de conceptualisation à 12, le score augmente alors à 14/16. Le réseau généré est plus simple (avec environ le 2/3 des concepts et des liens) et généralise mieux les images de l'ensemble de prédiction. Cette observation s'applique également avec le corpus MNIST.

## 6.1.4 MNIST

Le corpus MNIST (Modified National Institute of Standards and Technology database) (LeCun *et al.*, 2010) contient 60 000 images d'entraînement et 100 000 images de test. Chaque image, de 28 x 28 pixels, représente un nombre de 0 à 9 écrit à main levée en noir et blanc. Dans cette section, nous analysons la complexité, ainsi que la capacité de généralisation via la tâche de reconnaissance de 10 chiffres écrits  $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ , d'un réseau conceptuel construit

selon différentes profondeurs maximales de conceptualisation et selon différentes quantités de données d'entraînement extraites du corpus MNIST (LeCun *et al.*, 2010).

À chaque test, la moitié des données sont utilisées pour la construction du réseau (conceptualisation générative) et l'autre moitié pour la prédiction (où une conceptualisation non générative est effectuée). Le même nombre d'exemples pour chaque chiffre est considéré dans la première moitié des données *d'entraînement*. Si nous prenons par exemple un corpus de 100 images, 50 images sont utilisées pour la construction du réseau (5 par chiffre) et 50 images pour la prédiction.

La figure 6.4A présente le taux de bonnes prédictions obtenues selon la profondeur maximale de conceptualisation et le nombre de données considérées dans le corpus. En augmentant la profondeur de conceptualisation maximale, nous augmentons en quelque sorte la résolution de l'image conceptualisée. Or, cette conceptualisation des détails améliore les performances pour tous les corpus jusqu'à la profondeur 7. Ensuite, l'augmentation de la profondeur maximale de conceptualisation augmente en moyenne de plus en plus légèrement, impliquant également certaines baisses locales de performance.

On observe sur la figure 6.4B que le nombre de concepts générés augmente de façon exponentielle avec la profondeur de conceptualisation.

Les performances apparaissent liées au nombre de concepts dans le réseau. En effet à la Figure 6.4C, nous visualisons les performances obtenues en fonction du nombre de concepts générés<sup>11</sup>. La figure présente des performances similaires, pour un même nombre de concepts, surtout pour les réseaux générés à partir de 200 données d'entraînement ou plus. La performance du réseau entraîné avec le corpus de 100 données, et comportant moins de 10'000 concepts, est plus faible probablement parce que les données d'entraînement représentent mal les variations possibles pour chaque chiffre qui est entraîné avec seulement 5 exemples. Cependant, lorsque l'on considère les caractéristiques fines des chiffres avec une plus grande profondeur maximale

---

11. Pour un même nombre de concepts, il est à noter que la profondeur de conceptualisation sera d'autant plus importante que le nombre d'images du corpus est faible.

de conceptualisation, le corpus de 100 données a des performances similaires aux autres pour une taille de réseau d'environ 15'000 concepts.

Les performances stagnent à partir de 20 000 concepts. La meilleure performance est de 78% avec le corpus de 200 images (10 données d'entraînement pour chaque chiffre).

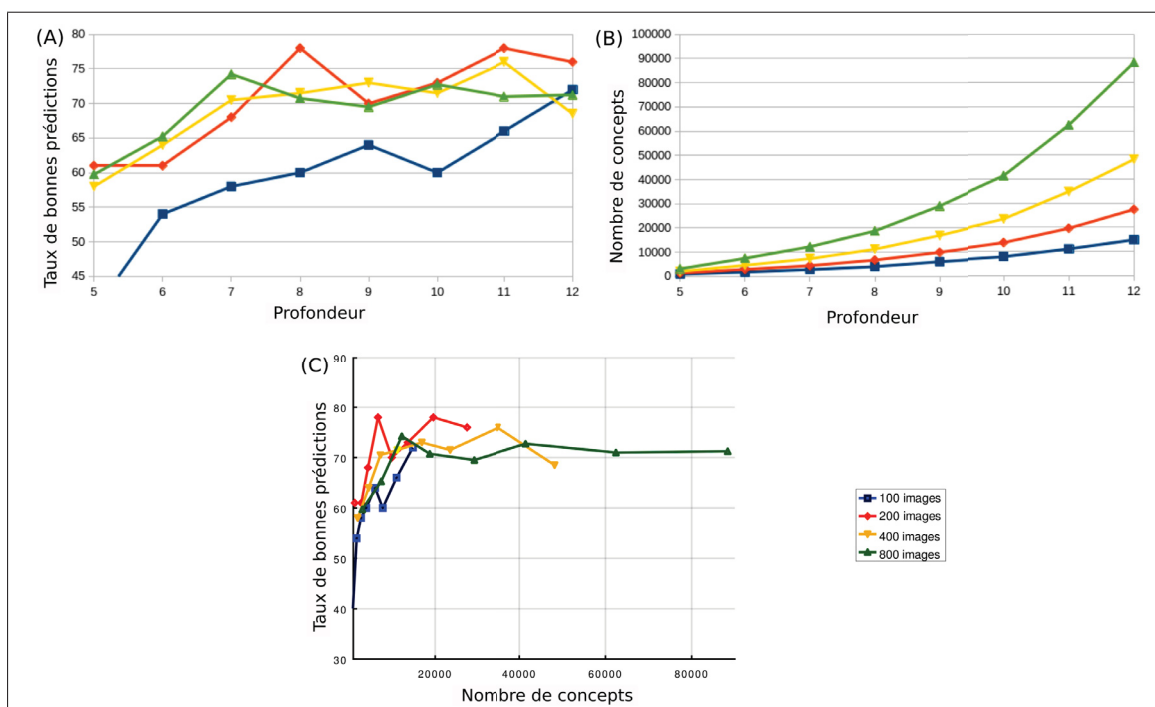


Figure 6.4 A : Taux de détection correcte en fonction de la profondeur maximale de conceptualisation, pour différentes tailles de corpus. B : Nombre de concepts générés en fonction du nombre de données utilisées du corpus MNIST et de la profondeur maximale de conceptualisation. C : Taux de bonnes détections en fonction du nombre de concepts générés

## 6.2 Reconnaissance de la parole : chiffres prononcés

### 6.2.1 Reconnaissance de la parole dépendante du locuteur

Maintenant, serait-ce possible de construire un réseau conceptuel qui encode l'expérience de la parole d'un groupe de locuteurs, puis de l'utiliser pour reconnaître des chiffres prononcés par un locuteur du groupe? Nous tentons cette expérience ici en utilisant le corpus *Free-Spoken-*

*Digit-Dataset* de Jackson *et al.* (2018). Ce corpus contient des enregistrements de 6 personnes prononçant les chiffres 0 à 9, 50 fois pour chaque chiffre. Il y a ainsi 3000 enregistrements au total. Le signal audio est échantillonné à 8KHz.

Les caractéristiques utilisées sont 30 coefficients MFCCs (Muda *et al.*, 2010) (Mel Frequency Cepstral Coefficients) extraits du signal audio sous une fenêtre de 32ms (256 données) et un pas de 15 ms.

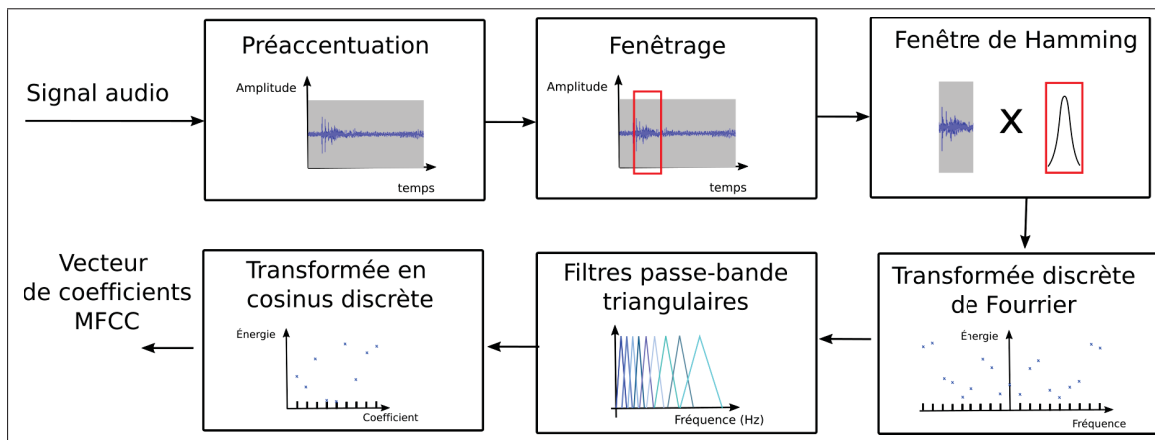


Figure 6.5 Extractions des MFCCs

La figure 6.5 décrit comment les coefficients MFCCs sont extraits. Le signal subit d'abord une préaccentuation des hautes fréquences, lesquelles sont très importantes pour l'identification des consonnes ; et naturellement de faible intensité dans le signal de la parole. Ensuite, le signal est considéré sur une certaine fenêtre à chaque instant où l'on s'intéresse à le caractériser dans le domaine fréquentiel. Afin de minimiser l'impact du fenêtrage dans l'évaluation du spectre, l'intensité du signal tout au long de la fenêtre est multipliée par une valeur correspondante d'une fonction de Hamming. Ces valeurs ainsi échelonnées sont donc utilisées afin d'évaluer la transformée de Fourier discrète. De ces valeurs, nous évaluons ensuite l'énergie de 30 bandes fréquentielles d'autant plus étroites que la sensibilité de l'oreille humaine aux différentes valeurs de la bande est grande. En appliquant une transformée discrète en cosinus sur ces trente valeurs, nous obtenons 30 coefficients dont l'information est décorrélée : soit les coefficients MFCCs. Les MFCCs sont actuellement évalués via la librairie de Six *et al.* (2014).



La valeur énergétique de chaque MFCC est encodée via un arbre binaire de prémices d'une profondeur ajustable  $N^{energies}$  qui correspond au nombre de feuilles (puisque l'arbre est binaire). Il y a donc un total de 30 arbres de prémices. Dans chaque arbre, les valeurs des prémices-feuilles sont déterminées de façon à bien se partager les valeurs d'énergie observées dans les données d'entraînement <sup>12</sup>.

L'expérience temporelle d'un MFCC est encodée dans un transpositif temporel où les  $L^{seq}$  feuilles sont actualisées selon la séquence des  $L^{seq}$  dernières prémices observées pour ce MFCC. La conceptualisation d'une fenêtre implique, conséquemment, celle des trente transpositifs temporels pour chaque groupe de valeurs consécutives des MFCCs. Chaque concept structurel généré lors de la conceptualisation encode une évolution temporelle caractéristique d'un MFCC spécifique.

Les chiffres de 0 à 9 sont encodés de la même façon que pour les chiffres écrits à la section précédente. Lors de l'entraînement, les concepts actifs de l'expérience sont consolidés avec les prémices-étiquettes décrivant la nature du chiffre. Lors de la détection, la conceptualisation est non générative. Le processus d'évocation est réalisé à partir des concepts actifs, ce qui donne une énergie à chaque prémisse-étiquette. Puisque le nombre de conceptualisations effectuées pour un chiffre prononcé dépend de la durée de l'enregistrement, nous distribuons l'énergie d'activation totale d'une expérience sur le nombre de conceptualisations consécutives à réaliser <sup>13</sup>. Ceci évite que l'expérience du réseau soit surreprésentée par les enregistrements les plus longs, qui impliquent davantage de conceptualisations et, de ce fait, de consolidation de concepts. La prémisse-étiquette retenue est alors celle dont la somme des énergies des conceptualisations consécutives (tout au long de l'expérience audio), normalisée sur ce nombre de conceptualisations, est maximale.

---

12. Plus spécifiquement : pour chaque MFCC, toutes les valeurs des énergies observées (incluant toutes les expériences d'entraînement) sont triées dans une liste, laquelle est divisée en  $N^{energies}$  sous-groupes. La valeur de chaque prémice est à la médiane du sous-groupe correspondant.

13. L'énergie maximale qu'un concept peut prendre lors d'une conceptualisation est  $1/(\text{nombre de conceptualisations})$ .

Pour la preuve de concept, nous réalisons d’abord quelques expériences avec 40 enregistrements audio par chiffre pour les trois premiers participants : 60% des enregistrements sont utilisés pour l’entraînement et 40% pour la prédiction. Plus en particulier, 24 enregistrements d’un chiffre d’un participant sont utilisés pour l’entraînement et 16 enregistrements pour la prédiction.

En choisissant  $N^{update} = 3$ ,  $N^{genesis} = 3$ ,  $L^{seq} = 10$  et  $N^{energies} = 20$ , nous obtenons un taux de succès de 320/420 (76%) avec un réseau contenant 1.0M de concepts, 1.6M de liens d’abstraction et 2.5M de liens d’agrégation. La matrice de confusion est présentée dans le tableau 6.2. Le taux de prédiction varie en fonction du chiffre prononcé, passant de 100% pour les chiffres 4 et 6 à 45% et 31% pour les chiffres 3 et 9 ; tous deux confondus en particulier avec le chiffre 6. Cette confusion dans les nombres ayant des phonèmes caractérisés par un bruit distribué sur le spectre ("s", "f", "R") peut s’expliquer par la difficulté à modéliser ces phonèmes, qui ont effectivement une grande variabilité structurelle.

Tableau 6.2 Matrice de confusion pour la prédiction des chiffres prononcés, avec  $N^{genesis} = 3$ ,  $L^{seq} = 10$  et  $N^{energies} = 20$

Vrai/prédit	0	1	2	3	4	5	6	7	8	9
0	31	0	2	0	1	0	8	0	0	0
1	0	34	0	0	4	0	2	0	0	2
2	0	0	41	0	1	0	0	0	0	0
3	4	1	3	19	0	0	14	0	1	0
4	0	0	0	0	42	0	0	0	0	0
5	0	0	0	0	3	35	4	0	0	0
6	0	0	0	0	0	0	42	0	0	0
7	0	5	0	0	1	1	6	29	0	0
8	0	2	0	0	0	0	6	0	34	0
9	0	9	0	0	5	0	15	0	0	13

En augmentant le nombre d’enregistrements de 40 à 45 pour chaque participant, et en augmentant le nombre de participants de 3 à 6, la tâche devient plus difficile. En gardant  $N^{update} = 3$ ,  $N^{genesis} = 3$ , alors le taux de succès baisse à 335/960 (34.90%) avec le réseau de 1.2M de concepts. Le réseau n’est donc pas en mesure de bien encoder la diversité des récurrences caractéristiques des participants. Cependant, en abaissant  $N^{update} = 3$  à  $N^{update} = 2$  tout en

conservant  $N^{genesis} = 3$ , la performance remonte à 692/960 (72%) avec 1.8M de concepts. En essayant  $N^{update} = 3$  et  $N^{genesis} = 2$ , le réseau comporte 3.2 millions de concepts et devient, ainsi, trop lent à opérer pour nos objectifs de test. Remarquons que dans l'ensemble des expériences de ce chapitre, les meilleurs résultats sont toujours obtenus avec  $N^{genesis} = 2$  au coût d'un réseau plus gros qu'avec  $N^{genesis} > 2$ .

## 6.2.2 Reconnaissance de la parole indépendante du locuteur

Si nous testions, maintenant, la capacité du système à reconnaître les chiffres indépendamment du locuteur? Nous entraînons alors le réseau avec les enregistrements des trois premiers locuteurs du corpus, puis effectuons la prédiction avec les enregistrements des trois derniers locuteurs du corpus.

Utilisant 45 données pour chaque chiffre d'un participant,  $N^{update} = 2$ ,  $N^{genesis} = 3$ ,  $L^{seq} = 10$  et  $N^{energies} = 20$ , nous obtenons un taux de succès de 248/600 (41%) avec un réseau comportant 1.81M de concepts. Le réseau n'est donc pas en mesure de bien modéliser la variabilité des chiffres prononcés. Cette lacune peut s'expliquer par le fait que les chiffres prononcés sont modélisés avec des prémices liées à des valeurs absolues d'énergies fréquentielles; pourtant sujet à changer d'un participant à l'autre. De ce fait, davantage de données d'entraînement sont requises afin que les chiffres soient modélisés selon différentes hauteurs possibles pour la voix humaine. Une piste de solution serait d'utiliser des prémices qui seraient liées, plutôt, à des énergies fréquentielles implicitement normalisées à chaque locuteur.

Pour l'instant cependant, en augmentant le nombre de participants utilisés pour l'entraînement de 3 à 4 (le nombre de participants pour la prédiction passe de 3 à 2), on augmente la performance à 180/280 (64.3%) avec un réseau de 1.95M de concepts. Fait intéressant, la performance augmente de 23% avec une augmentation de la taille du réseau de seulement 7.7%. Or, il est normal que la taille du réseau augmente de plus en plus tranquillement avec l'entraînement, puisque les nouvelles structures caractérisant les expériences se font de plus en plus rares. Néanmoins, celles découvertes apparaissent déterminantes pour la différenciation des

différents chiffres. Cette diminution du rythme d'augmentation de la complexité du réseau ne l'empêche donc pas, pour cette expérience, de modéliser la parole avec une précision qui suit la taille des données d'entraînement. Ceci suggère tout de même que la quantité de données d'entraînement est encore insuffisante <sup>14</sup>.

En augmentant le nombre de participants de 4 à 5, puis en répétant l'expérience six fois de sorte à permuter celui utilisé pour la prédiction, nous obtenons des performances de : 37/70, 54/70, 17/70, 27/70, 58/70 et 19/70, d'où une moyenne de 35.3/70 (50.5%) avec une variance de 14.3/70 (20.4%). Si on veut comparer aux cas précédant, on considère seulement les deux premiers résultats, pour une moyenne légèrement supérieure à un entraînement avec seulement 4 participants, soit 45.5/70 (65%). Cet exemple illustre que la capacité à reconnaître les chiffres varie largement selon le locuteur ; et confirme que trop peu de types de voix et façon de parler sont actuellement encodées par le modèle.

### 6.2.3 Reconnaissance de la parole : perspectives

Ces quelques expériences démontrent une capacité des réseaux conceptuels à modéliser les structures du signal de la parole, qu'il est ensuite possible d'utiliser dans l'objectif de prédiction par exemple de chiffres prononcés. Cette modélisation est réalisée sans égard à la tâche et sa complexité, ainsi les structures des phonèmes et triphones observés lors de la prononciation des chiffres 0 à 9 sont implicitement modélisées. Bien que cette modélisation s'avère un peu lourde pour notre objectif de distinction des chiffres prononcés (avec plus d'un million de concepts et de liens), elle prendrait tous son sens dans un contexte de reconnaissance de la parole au sens large.

Pour l'objectif de reconnaissance de la parole, au sens large, il importe de souligner que la technique proposée dans cette section n'est qu'un point de départ. Quels sont les entrées, les transpositifs d'entrée, les arbres de prémices les plus pertinents pour la tâche ? La façon d'utiliser les structures modélisant le signal audio pour l'objectif de reconnaissance est aussi à

---

14. En effet, lorsque l'entraînement est suffisant, le gain en performance ne suit plus l'augmentation de la quantité de données d'entraînement : de plus en plus de données sont nécessaires pour de faibles gains.

revisiter. Par exemple, consolider les concepts structuraux relatifs au signal audio directement avec la prémice-étiquette du mot auquel ils se rattachent n'est pas forcément la meilleure idée. Serait-ce pertinent, plutôt, de les consolider avec des prémices-étiquettes représentant les phonèmes et/ou triphones, qui seraient à leurs tours consolidés avec les prémices-étiquettes du mot relatif à l'expérience ? Bien d'autres solutions sont envisageables. La recherche est ouverte.

### 6.3 Reconnaissance du locuteur

Pour tester la reconnaissance du locuteur, nous répétons exactement l'expérience précédente de la section 6.2 (même corpus de données utilisé de la même manière). Cependant, nous ajoutons un arbre de prémisses dont la racine représente un locuteur au sens large, ayant comme enfants (les feuilles de l'arbre) les 6 locuteurs possibles. La façon de lier les concepts actifs aux prémisses est exactement la même que pour la reconnaissance des chiffres. Plus précisément, pendant l'entraînement, les concepts actifs de l'expérience sont consolidés avec la prémice-étiquette du locuteur ayant prononcé le chiffre. Pendant la prédiction (après le processus d'évocation des concepts actifs), l'étiquette retenue est celle liée à une prémice-étiquette d'énergie maximale, prenant en compte toutes les conceptualisations tout au long de l'expérience audio (exactement comme pour la reconnaissance des chiffres).

Pour l'expérience où  $N^{update} = 3$ ,  $N^{genesis} = 3$ , 40 expériences par participant et 3 participants, nous avons un taux de réussite de 325/420 (77.4%) pour la reconnaissance des chiffres et de 369/420 (87.9%) pour la reconnaissance du locuteur, toujours avec un réseau de 1.0M de concepts. En diminuant  $N^{genesis} = 2$ , nous avons une légère augmentation du taux de réussite de la reconnaissance des chiffres à 334/420 (79.5%) avec le réseau de 2.7M concepts, alors que le taux de réussite de la reconnaissance du locuteur est inchangé. Avec 45 expériences par participant et 6 participants,  $N^{update} = 2$  et  $N^{genesis} = 3$ , le taux de réussite de la reconnaissance des chiffres est de 692/960 (72.1% : identique à l'expérience équivalente sans reconnaissance du locuteur) et le taux de reconnaissance du locuteur est de 775/960 (80.7%) avec un réseau de 1.8M concepts.

Tableau 6.3 Matrice de confusion pour la prédiction des locuteurs

Vrai/prédit	Jackson	Lucas	Nicolas	Théo	Yweweler	George
Jackson	88	1	42	20	4	5
Lucas	0	149	10	1	0	0
Nicolas	1	0	158	0	1	0
Théo	4	2	10	138	5	1
Yweweler	10	27	23	5	94	1
George	5	1	2	1	2	149
Total	108	180	245	165	106	156

Le tableau 6.3 présente la matrice de confusion pour la reconnaissance du locuteur. On observe des performances assez différentes d'un locuteur à l'autre : 158/160 (98.8%) pour Nicolas, 149/160 (93.1%) pour George et Lucas, 138/160 (86.3%) pour Théo; 94/160 (58.8%) pour Yweweler et 88/160 (55%) pour Jackson. Jackson a été pris 42/160 (26.3%) pour Nicolas et 20/160 (12.5%) pour Théo; Lucas a été confondu en particulier avec Nicolas 10/160 (6.3%); Théo en particulier également avec Nicolas 10/160 (6.3%); Yweweler en particulier avec Lucas 27/160 (16.9%) puis avec Nicolas 23/160 (14.4%); Georges avec Jackson 5/160 (3.1%). Il n'y a pas de réciprocité dans les confusions : par exemple, Yweweler est pris pour Lucas 27/160, mais Lucas pour Yweweler 0/160. Cependant, certains locuteurs sont davantage représentés dans les prédictions comme on l'observe à la dernière ligne avec le total. L'énergie d'activation supérieure qu'obtiennent certains locuteurs, tels que Nicolas, Lucas et Georges, indique qu'ils sont davantage représentés par les concepts actuels du réseau. Davantage de concepts, impliquant un plus grand entraînement, seraient ainsi nécessaires pour encoder les caractéristiques des locuteurs tels que Jackson ou Yweweler. Il est intéressant de noter que la structure du réseau, qui n'est pas optimisée pour une tâche particulière, convient cette fois à deux tâches simultanées : reconnaître un locuteur et identifier le numéro qu'il a prononcé. L'exécution d'une tâche semble, pour l'instant, comme prévu, peu ou pas affectée par l'ajout de tâches tierces.

## CONCLUSION ET RECOMMANDATIONS

### 7.1 Contributions principales

J'ai proposé dans cette thèse l'idée que la structure d'un système d'apprentissage peut encoder la structure logique d'un phénomène observé. Dans cette perspective, j'ai introduit les *réseaux conceptuels* dont la structure est construite automatiquement avec l'expérimentation via un mécanisme de *conceptualisation*. L'utilisation du modèle pour la détection, la prédiction, la classification, le contrôle, etc., est néanmoins possible en considérant l'énergie des concepts via un *mécanisme d'activation*.

Le chapitre 5 a présenté une façon de réaliser la reconnaissance de formes à partir d'un réseau conceptuel. Au chapitre 6, nous avons donné des exemples : de prédiction de chiffres écrits en utilisant des corpus faits maison et le corpus MNIST ; puis de prédiction de chiffres parlés et de reconnaissance du locuteur en utilisant le Free-Spoken-Digit-Dataset (corpus de données libres portant sur des chiffres prononcés). Les résultats ont démontré la capacité du réseau conceptuel à se construire rapidement, avec peu de données, de sorte à pouvoir être utilisé avec succès pour la prédiction de nouveaux chiffres ou parlés. Ces tests ont démontré, en résumé : (1) l'utilité des réseaux conceptuels dans la tâche de reconnaissance de formes ; (2) les propriétés des réseaux conceptuels ; et (3) leurs limites qui encouragent la poursuite de la recherche.

### 7.2 Comparaison avec les autres approches d'apprentissage machine

L'objectif du modèle proposé est de *représenter* des expériences, et non de *prédire* des formes ou des valeurs, bien que le modèle puisse être utilisé en amont pour faire de la prédiction.

Par opposition, les modèles traditionnels (réseaux de neurones, SVM, arbre de décision) comportent une structure fixe durant l'apprentissage. Ils ne modélisent pas implicitement les expériences, mais plutôt implémentent une relation entrée/sortie. L'apprentissage consiste à inférer

les valeurs optimales des paramètres de la structure afin que cette dernière génère des sorties plausibles à partir des entrées.

Le paradigme proposé à plusieurs avantages.

Premièrement, le fait que la structure du réseau soit générée automatiquement lui permet d'avoir une complexité qu'il aurait été impossible de réaliser manuellement. Cette complexité apparaît utile dans la mesure où elle reflète celle des expériences du système.

Deuxièmement, puisque la structure se limite à représenter les expériences, sans être modulée par des objectifs de prédiction quelconques (ou éventuellement de contrôle dans les applications robotiques), cette dernière se présente comme une base pouvant servir à des objectifs variés, telle une sorte de représentation passive de l'univers sensible du système. Nous avons vu par exemple au chapitre 6 que l'ajout d'une tâche (reconnaître le locuteur) a eu très peu d'incidence sur la structure du réseau (qui était initialement utilisé pour reconnaître la parole). Nous pouvons faire l'hypothèse qu'une telle représentation globale des expériences sensibles est à la base de la pensée humaine, ayant effectivement la capacité de renouveler perpétuellement ses objectifs ; réorganiser ses conclusions et actions. Cette hypothèse impliquerait que la majorité des neurones du cerveau n'ont pas de sortie proprement dite, mais plutôt s'activent selon l'entité représentée dans le schéma de l'expérience sensorimotrice.

Troisièmement, le mécanisme de conceptualisation créer implicitement une structure dotée d'une capacité de généralisation avec très peu de données, ce qui peut être intéressant par exemple pour les applications où l'accès aux données est limité.

Cependant, le paradigme proposé à aussi des inconvénients.

Premièrement, puisque la structure du réseau conceptuel n'est pas optimisée pour un objectif de prédiction précis ; elle est sujette à être moins efficace qu'un modèle dédié et optimisé pour



cet objectif : en exigeant plus de paramètres et d'unités pour des performances de prédiction analogues.

Deuxièmement, la complexité du réseau augmente exponentiellement avec celle des entrées. Le cerveau humain comportant environ 100 milliards de neurones, nous ne pouvons prétendre que cette complexité n'est pas nécessaire à une bonne représentation des expériences sensori-motrices. Néanmoins, la capacité des systèmes informatiques actuels apparaît insuffisante pour traiter un aussi grand nombre d'unités. Dans nos expériences, les réseaux conceptuels comportant plus de 2M unités (50'000 fois moins qu'un cerveau humain) devenaient déjà difficiles à traiter. L'ordinateur contemporain pourrait ne pas être le meilleur support aux réseaux conceptuels.

### **7.3 Perspectives**

Les propriétés des réseaux conceptuels et les mécanismes proposés dans cette thèse (conceptualisation, activation, génération de concepts, consolidation...) sont offerts à la communauté scientifique en tant que « première ébauche ». Les propriétés et les mécanismes pourraient être revus, améliorés, remplacés, soustraits et/ou ajoutés afin d'améliorer la capacité de représentation du modèle, de faciliter sa mise en œuvre et d'accroître son potentiel pratique. D'autres modèles de représentation des phénomènes observés pourraient également être proposés à partir de zéro. Néanmoins, il est déjà possible d'étudier et d'utiliser le réseau conceptuel, tel que proposé, dans des applications pratiques. Cette thèse se veut, en somme, initier un axe de recherche en intelligence artificielle, qui serait porté sur la modélisation des expériences phénoménologiques, avec les réseaux conceptuels ou d'autres approches à venir.

Bien que cette recherche peut se poursuivre selon de multiples avenues, les recherches suivantes m'apparaissent tout particulièrement intéressantes pour élargir la praticabilité de l'approche :

1. *Réaliser un type d'apprentissage par renforcement à l'aide d'un réseau conceptuel.* Il serait possible de réaliser des transpositifs de sorties à même titre que des transpositifs d'entrées. Ces transpositifs auraient à leurs feuilles des prémices représentant des commandes motrices spécifiques. La conceptualisation temporelle permettrait ainsi de créer des représentations des schémas moteurs exécutés et de les lier aux autres types d'expériences sensorielles, incluant en l'occurrence des états cibles affinés aux objectifs du système. La recherche d'un état cible spécifique pourrait, conséquemment, se traduire par l'exploitation des concepts-*moteurs* (concepts qui représentent des schémas moteurs) qui leur sont les plus fortement liés. Remarquons que la forte plasticité des premiers concepts générés implique une variabilité des schémas moteurs qu'ils représentent. Par exemple, un concept-moteur désordonné permet qu'un groupe d'actions soit exécuté dans un ordre quelconque ; pendant qu'un concept-moteur très abstrait accepte des états potentiellement très différents (comme une prémice qui représente une vitesse de 0 à 100 km/h). Or, pour exécuter un concept-moteur plastique, il suffit de générer de façon aléatoire l'une de ses représentations, laquelle est traduite par une séquence spécifique de commandes motrices. Il y a donc une boucle correspondant à un *apprentissage par renforcement* où : les concepts-moteurs les plus liés aux états cibles sont sélectionnés pour générer des séquences motrices ; qui sont à leurs tours conceptualisées dans l'expérimentation du système de sorte à générer d'autres concepts-moteurs plus rigides ; ceux davantage liés aux états cibles sont à leur tour priorisés et le cycle recommence. D'un cycle à l'autre, les concepts-moteurs se raffinent d'autant plus qu'ils se lient aux états cibles. Il s'agit ici de l'idée préliminaire. Développer la théorie, tout en la testant dans une application réelle, est un projet de recherche.
2. *Systématiser l'interaction des concepts hétérogènes.* Devrait-on effectuer une conceptualisation hétérogène qui détecte et génère des concepts formés de sous-concepts de différents types ? Devrait-on plutôt simplement lier les concepts de différents types entre eux sans

qu'il y ait de concept hétérogène défini par leur regroupement ? De quelle façon devrait être réalisée, idéalement, la conceptualisation hétérogène, ou de quelle façon les concepts devraient être liés entre eux ? Quel est l'avantage de l'une ou l'autre stratégie dans des contextes cibles d'applications ? Il s'agit de questions que j'ai abordées dans la thèse, sans toutefois les résoudre dans toute leur complexité. La théorie d'interaction des concepts de différents types pourrait être développée dans une application, par exemple, où des images/vidéos et des sons seraient expérimentés simultanément. Un objectif de test serait de générer un son à partir d'une image/vidéo. Cette tâche est apparentée à celle de prédire une étiquette à partir d'une image (comme au chapitre 6). Cependant, les concepts liés à la sonorité étant générés automatiquement lors de la conceptualisation en même temps que les concepts des images, une stratégie plus sophistiquée et appuyée de fondements théoriques serait requise pour concrétiser cette interaction dynamique entre un grand nombre de concepts.

3. *Optimiser la rapidité de traitements.* Il serait intéressant d'étudier l'implémentation des algorithmes de conceptualisation et d'activation dans un système électrique dédié afin d'exploiter le parallélisme inhérent des traitements.

Les deux premiers points conviendraient, à mon avis, pour des étudiants au doctorat respectivement en robotique et en informatique (ou génie informatique), tandis que le dernier point se présente plus adapté à une équipe de recherche en microélectronique et/ou génie physique.



## ANNEXE I

### ÉVALUATION DE L'ABSTRACTION COMMUNE À PLUSIEURS CONCEPTS

L'objectif est d'évaluer l'abstraction la plus rigide pour représenter un ensemble de  $N^C$  concepts  $\{C_1, \dots, C_{N^C}\}$ . Nous proposons pour l'instant la stratégie suivante :

- *Idée* : chaque concept  $C_i \forall i = 1 \dots N^C$  ainsi que chacune de ses abstractions implicites est un candidat possible. On associe à chaque candidat un tableau de  $N^C$  valeurs initialisées à l'infinie, qui devra correspondre à la distance d'abstraction de chaque concept  $C_i$  avec le candidat. Le candidat dont la somme des éléments du tableau est minimale est considéré comme étant l'abstraction commune la plus rigide pour représenter l'ensemble des concepts, pour autant que cette somme soit finie. Autrement, aucun concept ne peut représenter à la fois tous les éléments  $C_i$  de l'ensemble. Remarquons qu'il y a aucune abstraction commune si les concepts  $C_i$  ne sont pas tous de même type. En opposition, les concepts d'un même type ont toujours la prémice racine du type pour abstraction commune.
- Pour chaque concept  $C_i$  de l'ensemble, on repasse récursivement ses abstractions en passant en paramètre la profondeur courante  $p$  de la récursion. La récursion partant de  $C_i$  est de profondeur 0 puisqu'elle visite  $C_i$  directement. Les abstractions de  $C_i$  sont de profondeur 1, puis leurs abstractions de profondeur 2, ainsi de suite. Lorsqu'un concept est ainsi visité, l'élément  $i$  du tableau lié au candidat est mise à jour selon la profondeur courante de la récursion.
- Une fois que tous les concepts candidats ont été visités partant de chaque élément  $C_i$  (point précédent), on évalue la somme des éléments du tableau de chaque candidat afin de trouver le candidat optimal.

*Note.* Il semble que cette stratégie pourrait être optimisée de sorte à converger vers le concept optimal sans évaluer toutes les possibilités assurément sous-optimales, telle que réalisée par exemple lors de la recherche d'une trajectoire optimale dans un graphe (Magzhan & Jani, 2013).



## ANNEXE II

### IDENTIFICATION DES ABSTRACTIONS ACTIVES

Supposant un réseau conceptuel donné, le problème de cette section est d'identifier les concepts de ce réseau qui admettent une composition détectée. Ces concepts sont dits « abstractions actives ».

Pour simplifier, nous considérons premièrement que tous les concepts sont ordonnés. Supposons qu'une composition ordonnée  $X^o = \{\chi_1, \chi_2, \dots, \chi_N\}^o$  est détectée. Les agrégats implicites de chaque élément  $\chi_i$  sont dénotés  $\{\Psi^1(\chi_i), \Psi^2(\chi_i), \dots, \Psi^M(\chi_i)\}$  où chaque agrégat  $\Psi$  comporte les composants  $\{\psi_1, \psi_2, \dots, \psi_N\}$ . Si un agrégat ordonné  $\Psi^o$  admet la composition  $X^o$ , alors chaque composant  $\psi_i$  admet l'élément  $\chi_i$  de  $X^o$  à la position correspondante  $i$ . L'agrégat est alors une abstraction de la composition observée et est dit « actif ». La figure II-1 présente l'idée. Les agrégats implicites de L41 sont  $\Psi^a$  (puisque L31 admet L41),  $\Psi^b$ ,  $\Psi^c$ ,  $\Psi^d$  et  $\Psi^e$ . Cependant, comme le premier composant  $\psi_1 = L42$  de  $\Psi^e$  n'admet pas L41,  $\Psi^e$  ne peut admettre la composition  $\{L41, L42, L43\}^o$ . On doit maintenant vérifier quels sont les agrégats possibles considérant aussi le deuxième élément de la composition, soit L42. Comme le deuxième élément de  $\Psi^a$ , L43, n'admet pas L42, alors  $\Psi^a$  est rejeté. Comme le deuxième élément de  $\Psi^d$ , L41, n'admet pas L42,  $\Psi^d$  est maintenant rejeté. Les deux candidats restants sont alors  $\Psi^b$  et  $\Psi^c$ . On doit vérifier lequel de ces candidats admettent L43 en troisième position : les deux. On peut donc conclure que  $\Psi^b$  et  $\Psi^c$  admettent la composition  $\{L41, L42, L43\}^o$  : il s'agit des abstractions implicites activées par la composition observée.

Dans le cas précédent où les agrégats sont présumés ordonnés, nous pouvions évaluer si un agrégat  $\Psi^o$  admet une composition  $X^o$  qu'en vérifiant que chaque élément admet celui à la position correspondante de la composition. Dans le cas où un agrégat est désordonné, la correspondance entre les éléments ne peut être établie à partir des positions. Dans ce cas, l'agrégat désordonné  $\Psi^o$  est actif s'il est possible d'identifier pour chaque composant de  $\Psi^o$  un abstracteur implicite d'un élément distinctif de  $X^o$ . Il s'agit ici d'un pairage entre les éléments de  $\Psi^o$  et ceux de  $X^o$ . Par exemple, la composition  $X^o = \{L44, L43, L43, L41\}^o$  est admise par l'agrégat

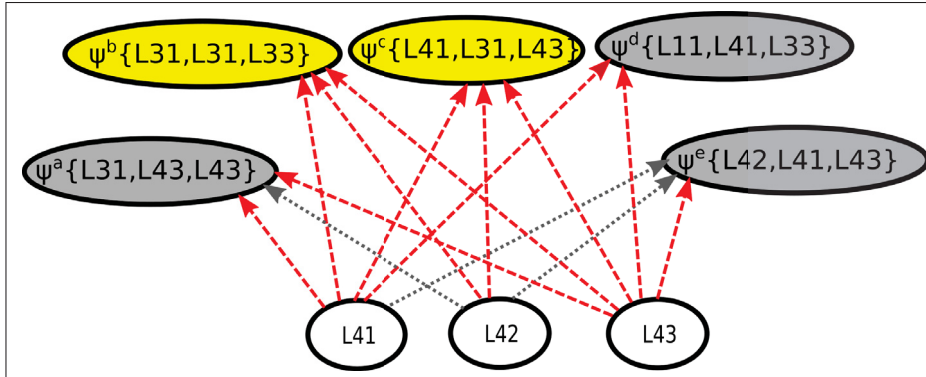


Figure-A II-1 Identification des abstractions implicites actives (et ordonnées) d'une composition ordonnée.

gat désordonné  $\Psi^d = \{L43, L31, L32, L33\}^d$  puisque : le 1<sup>er</sup> élément  $L43$  de  $\Psi^d$  admet le 2<sup>e</sup> élément  $L43$  de  $X^o$  ; le 3<sup>e</sup> élément  $L32$  de  $\Psi^d$  admet le 3<sup>e</sup> élément  $L43$  de  $X^o$  ; le 4<sup>e</sup> élément  $L33$  de  $\Psi^d$  admet le 1<sup>er</sup> élément  $L44$  de  $X^o$  ; le 2<sup>e</sup> élément  $L31$  de  $\Psi^d$  admet 4<sup>e</sup> élément  $L41$  de  $X^o$ . L'agrégat  $\Psi^d$  n'admet pas  $X^o$  s'il est impossible d'effectuer un pairage valide de ses éléments avec ceux de  $X^o$ . L'algorithme A-II.1 résume la stratégie employée selon l'ordonnance de la composition et des agrégats.

---

**Algorithm A-II.1** Détermination des abstractions implicites activées par une composition

**Result :** Déterminer les abstractions implicites activées par une composition  $X = \{\chi_1, \chi_2, \dots, \chi_N\}$  ordonné ou non.

- 1 Soit une méthode *pairageOrdonnée*( $\Psi, X$ ) qui retourne vrai si chaque élément  $\psi_i$  de  $\Psi$  admet l'élément correspondant  $x_i$  de  $X$  Soit une méthode *pairageDesordonnée*( $\Psi, X$ ) qui retourne vrai s'il est possible de jumeler les composants de  $\Psi$  avec ceux de  $X$ . Pour chaque paire  $p_n = \{\psi_i, \chi_j\}$  de  $N$  paires,  $\psi_i$  admet  $\chi_j$ ,  $i \neq n \forall n \neq i$  et  $j \neq n \forall n \neq j$ . Soit une liste d'abstractions actives On parcourt récursivement les agrégats implicites du premier concept structural de la composition, dénoté  $\chi_x$  **foreach** Agrégat implicite  $\Psi_j$  de  $\chi_x$  **do**
  - 2 | **if** ( $\Psi_j$  est ordonnée ET  $X$  est ordonnée ET *pairageOrdonnée*( $\Psi_j, X$ )) OU ( $\Psi_j$  est désordonnée ET *pairageDesordonnée*( $\Psi, X$ )) **then**
  - 3 | | On ajoute  $\Psi_j$  à la liste d'abstractions actives
  - 4 | **end**
  - 5 **end**
-



## BIBLIOGRAPHIE

- Aldrich, J. (1997). RA Fisher and the making of maximum likelihood 1912-1922. *Statistical science*, 12(3), 162–176.
- Allen, N. J. & Eroglu, C. (2017). Cell biology of astrocyte-synapse interactions. *Neuron*, 96(3), 697–708.
- Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M. & Farhan, L. (2021). Review of deep learning : Concepts, CNN architectures, challenges, applications, future directions. *Journal of big Data*, 8, 1–74.
- Barros, R. C., Basgalupp, M. P., de Carvalho, A. C. P. L. F. & Freitas, A. A. (2012). A survey of evolutionary algorithms for decision-tree induction. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(3), 291-312. doi : 10.1109/TSMCC.2011.2157494.
- Berger, J. O. (2013). *Statistical decision theory and Bayesian analysis*. Springer Science & Business Media.
- Bishop, C. M. & Nasrabadi, N. M. (2006). *Pattern recognition and machine learning*. Springer.
- Boucher, P. & Plusquellec, P. (2019). Acute stress assessment from excess cortisol secretion : Fundamentals and perspectives. *Frontiers in Endocrinology*, 10, 749. doi : 10.3389/fendo.2019.00749.
- Boucher, P., Dufour, P., Plusquellec, P., Dehak, N., Dumouchel, P. & Cardinal, P. (2016). PHY-SIOSTRESS : A multimodal corpus of data on acute stress and physiological activation. *In Workshop on Multimodal Corpora : Computer vision and language processing (MMC 2016)*, 45-48.
- Breiman, L. (2001). Random forests. *Machine learning*, 45, 5–32.
- Bro, R. & Smilde, A. K. (2014). Principal component analysis. *Analytical methods*, 6(9), 2812–2831.
- Caballero, D., Araya, R., Kronholm, H., Viiri, J., Mansikkaniemi, A., Lehesvuori, S., Virtanen, T. & Kurimo, M. (2017). ASR in Classroom Today : Automatic Visualization of Conceptual Network in Science Classrooms. *Data Driven Approaches in Digital Education*, pp. 541–544.

- Cordier, R. (1964). Chapter 5 - Sensory Cells. Dans Brachet, J. & Mirsky, A. E. (Éds.), *The Cell* (pp. 313-386). Academic Press. doi : <https://doi.org/10.1016/B978-0-12-123306-8.50013-X>.
- Dahl, J., Myhrhaug, B. & Nygaard, K. (1968). *Some features of the SIMULA 67 language*.
- Dehaene, S., Dehaene-Lambertz, G. & Cohen, L. (1998). Abstract representations of numbers in the animal and human brain. *Trends in Neurosciences*, 21(8), 355-361. doi : [https://doi.org/10.1016/S0166-2236\(98\)01263-6](https://doi.org/10.1016/S0166-2236(98)01263-6).
- Earp, F. S. (1996). Transactions, circuits, and identity : Proposing a conceptual network. *Journal of Economic Issues*, 30(2), 407-412. doi : 10.1080/00213624.1996.11505804.
- Elson, R. C., Selverston, A. I., Huerta, R., Rulkov, N. F., Rabinovich, M. I. & Abarbanel, H. D. I. (1998). Synchronous behavior of two coupled biological Neurons. *Physical Review Letters*, 81, 5692–5695. doi : 10.1103/PhysRevLett.81.5692.
- Fukushima, K. (1988). Neocognitron : A hierarchical neural network capable of visual pattern recognition. *Neural Networks*, 1(2), 119-130. doi : [https://doi.org/10.1016/0893-6080\(88\)90014-7](https://doi.org/10.1016/0893-6080(88)90014-7).
- Georgiev, G. V. & Nagai, Y. (2011). A conceptual network analysis of user impressions and meanings of product materials in design. *Materials & Design*, 32(8), 4230-4240. doi : <https://doi.org/10.1016/j.matdes.2011.04.024>.
- GhaviDel, S., Nezamdost, S. A. & Riahinia, N. (2020). Conceptual network evolution of cybernetic area in middle east countries. *International Journal of Information Science and Management (IJISM)*, 18(2), 97-114. Repéré à [https://ijism.ricest.ac.ir/article\\_698326.html](https://ijism.ricest.ac.ir/article_698326.html).
- Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J. & Chen, T. (2018). Recent advances in convolutional neural networks. *Pattern Recognition*, 77, 354-377. doi : <https://doi.org/10.1016/j.patcog.2017.10.013>.
- Hinton, G. E. (2007). Boltzmann machine. *Scholarpedia*, 2(5), 1668.
- Hochreiter, S. & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735–1780.
- Jackson, Z., Souza, C., Flaks, J., Pan, Y., Nicolas, H. & Thite, A. (2018). Jakobovski/free-spoken-digit-dataset : v1.0.8 (Version v1.0.8). Zenodo. doi : 10.5281/zenodo.1342401.

- Jadidinejad, A. H., Mahmoudi, F. & Meybodi, M. (2016). Conceptual feature generation for textual information using a conceptual network constructed from Wikipedia. *Expert Systems*, 33(1), 92–106.
- Korn, H. & Faure, P. (2003). Is there chaos in the brain? II. Experimental evidence and related models. *Comptes Rendus Biologies*, 326(9), 787 - 840. doi : <https://doi.org/10.1016/j.crv.2003.09.011>.
- LeCun, Y., Cortes, C. & Burges, C. (2010). MNIST handwritten digit database. Florham Park, NJ, USA.
- Magzhan, K. & Jani, H. M. (2013). A review and evaluations of shortest path algorithms. *International Journal of Scientific & Technology Research*, 2(6), 99–104.
- Moon, T. K. (1996). The expectation-maximization algorithm. *IEEE Signal processing magazine*, 13(6), 47–60.
- Muda, L., Begam, M. & Elamvazuthi, I. (2010). Voice recognition algorithms using mel frequency cepstral coefficient (MFCC) and dynamic time warping (DTW) techniques. *arXiv preprint arXiv :1003.4083*.
- Müller, B., Reinhardt, J. & Strickland, M. T. (2012). *Neural networks : an introduction*. Springer Science & Business Media.
- Murthy, S. K. (1998). Automatic construction of decision trees from data : A multi-disciplinary survey. *Data mining and knowledge discovery*, 2, 345–389.
- Nguyen, T. (2010). Total number of synapses in the adult human neocortex. *Undergraduate Journal of Mathematical Modeling : One+ Two*, 3(1), 26.
- O’Shea, K. & Nash, R. (2015). An introduction to convolutional neural networks. *arXiv preprint arXiv :1511.08458*.
- Rabiner, L. & Juang, B. (1986). An introduction to hidden Markov models. *iee assp magazine*, 3(1), 4–16.
- Reynolds, D. (2009). Gaussian Mixture Models. Dans Li, S. Z. & Jain, A. (Éds.), *Encyclopedia of Biometrics* (pp. 659–663). Boston, MA : Springer US. doi : 10.1007/978-0-387-73003-5\_196.
- Robert, C. (2005). *Le choix bayésien : principes et pratique*. Springer Science & Business Media.

- Rosenblatt, F. (1958). The perceptron : A probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6), 386.
- Rumelhart, D. E., Hinton, G. E. & Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, 323(6088), 533–536.
- Safavian, S. & Landgrebe, D. (1991). A survey of decision tree classifier methodology. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(3), 660-674. doi : 10.1109/21.97458.
- Schölkopf, B., Smola, A. J., Bach, F. et al. (2002). *Learning with kernels : support vector machines, regularization, optimization, and beyond*. MIT press.
- Sharma, S., Sharma, S. & Athaiya, A. (2017). Activation functions in neural networks. *Towards Data Science*, 6(12), 310–316.
- Six, J., Cornelis, O. & Leman, M. (2014). TarsosDSP, a real-time audio processing framework in Java. *Proceedings of the 53rd AES Conference (AES 53rd)*.
- Thom, H. C. (1958). A note on the gamma distribution. *Monthly Weather Review*, 86(4), 117–122.
- Tomičić, I. & Schatten, M. (2020). A conceptual network analysis of gamification practices in primary and secondary education. *2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO)*, pp. 1091-1097. doi : 10.23919/MIPRO48935.2020.9245225.
- Van Houdt, G., Mosquera, C. & Nápoles, G. (2020). A review on the long short-term memory model. *Artificial Intelligence Review*, 53, 5929–5955.
- Yarlikas, S. & Bilgen, S. (2012). Economic and technical assessment of cloud computing. *Closer*, pp. 167–172.
- Zhang, M.-L. & Zhou, Z.-H. (2007). ML-KNN : A lazy learning approach to multi-label learning. *Pattern recognition*, 40(7), 2038–2048.
- Zhao, Y., Zeng, J., Yang, Y. & Chen, L. (2008). A conceptual network based modeling framework for semantic representation of chinese news. *2008 Fourth International Conference on Natural Computation*, 6, 221-225. doi : 10.1109/ICNC.2008.27.