Opinions Libres
le blog d'Olivier Ezratty

Les avancées de l'intelligence artificielle – 2

Après une **petite introduction** sur ce qu'est ou pas l'intelligence artificielle, passons à un côté plus terre à terre en faisant un petit inventaire approximatif des techniques de l'IA. Il s'agit toujours de vulgarisation et d'une restitution de mon processus de découverte du sujet au fil de l'eau ! Nous évoquerons en partie la question du matériel, notamment pour les réseaux de neurones. Le reste le sera dans la dernière partie de cette série d'articles.

Des hivers au printemps de l'IA

L'histoire moderne de l'intelligence artificielle a démarré comme nous l'avons vu dans la première partie en 1957. S'en est suivi une période de recherche fondamentale importante, notamment au MIT AI Lab, à l'origine notamment du langage **LISP** (1958) qui servit pendant deux à trois décennies à développer des solutions logicielles d'IA. Ces recherches étaient financées par l'ARPA, l'agence de recherche du Pentagone devenue ensuite la DARPA, l'équivalent de la DGA française, mais évidemment bien mieux financée. La recherche sur l'IA était principalement financée par les deniers publics, notamment aux USA et au Royaume-Uni. Encore aujourd'hui, une très grande partie des recherches les plus avancées sur l'IA aux USA le sont par l'omniprésente DARPA.

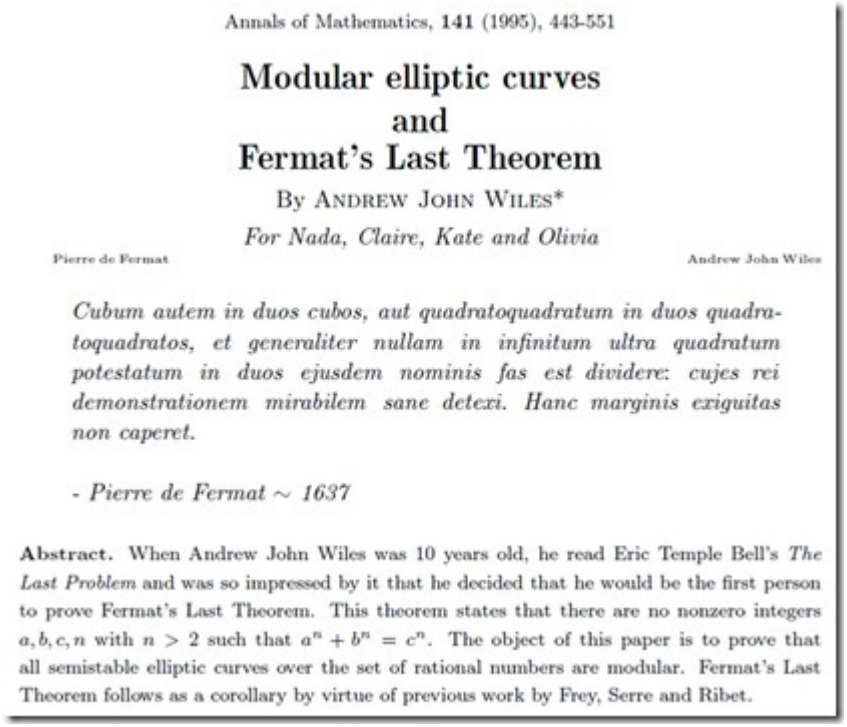
L'IA connu son premier "hiver" avec une réduction d'une bonne part de ces budgets à partir de 1973, tant au Royaume-Uni qu'aux USA. C'était la conséquence de la publication du **Rapport Lighthill** destiné à l'organisme public britannique **Science Research Council** – équivalent de l'Agence Nationale de la Recherche française – qui remettait en cause le bien fondé des recherches de l'époque en robotique et en traitement du langage. Cet hiver a duré jusqu'en 1980.

En cause, des promesses un peu trop optimistes des experts du secteur. Comme souvent, les prévisions peuvent être justes sur tout ou partie du fond mais à côté de la plaque sur leur timing. Cette **histoire de l'IA** en fait un

inventaire intéressant. **Herbert Simon** (*ci-dessus*) et **Allen Newell** prévoyaient en 1958 qu'en 10 ans, un ordinateur deviendrait champion du monde d'échecs et un autre serait capable de prouver un nouveau et important théorème mathématique. 30 ans d'erreur pour la première prévision et autant pour la seconde sachant qu'elle est toujours largement en devenir pour être générique ! Cet écueil est le même dans les prévisions actuelles autour de la singularité et du transhumanisme (l'ordinateur plus intelligent que l'homme en 2030, l'immortalité pour nos enfants, etc).

Le chercheur d'IBM Herbert Gelernter avait réussi en 1958 à utiliser un logiciel de démonstration de théorèmes de géométrie fonctionnant en chainage arrière (de la solution jusqu'au problème) sur un IBM 704 et à partir d'une base de 1000 règles. Cela relevait d'une combinatoire plutôt simple. Il en va autrement du théorème d'incomplétude de **Godel** qui dit que "*dans n'importe quelle théorie récursivement axiomatisable, cohérente et capable de « formaliser l'arithmétique, on peut construire un énoncé arithmétique qui ne peut être ni prouvé ni réfuté dans cette théorie"* ou encore du dernier théorème de **Fermat** (x^n + y^n = z^n impossible pour un entier n>2) qui n'ont jamais été démontrés via de l'IA.

Le théorème de Fermat a été démontré au milieu des années 1990 et après des années d'efforts de plusieurs mathématiciens dont **Andrew Wiles**. Sa démonstration publiée dans les annales de mathématiques fait 109 pages et fait appel à de nombreux concepts ! Un **défi a été lancé en 2005** par un certain Jan Bergstra pour démontrer le théorème de Fermat avec un ordinateur et il reste toujours à relever. A vous de jouer si cela vous tente !



Annals of Mathematics, **141** (1995), 443-551

Modular elliptic curves and Fermat's Last Theorem

By Andrew John Wiles*

For Nada, Claire, Kate and Olivia

Pierre de Fermat — Andrew John Wiles

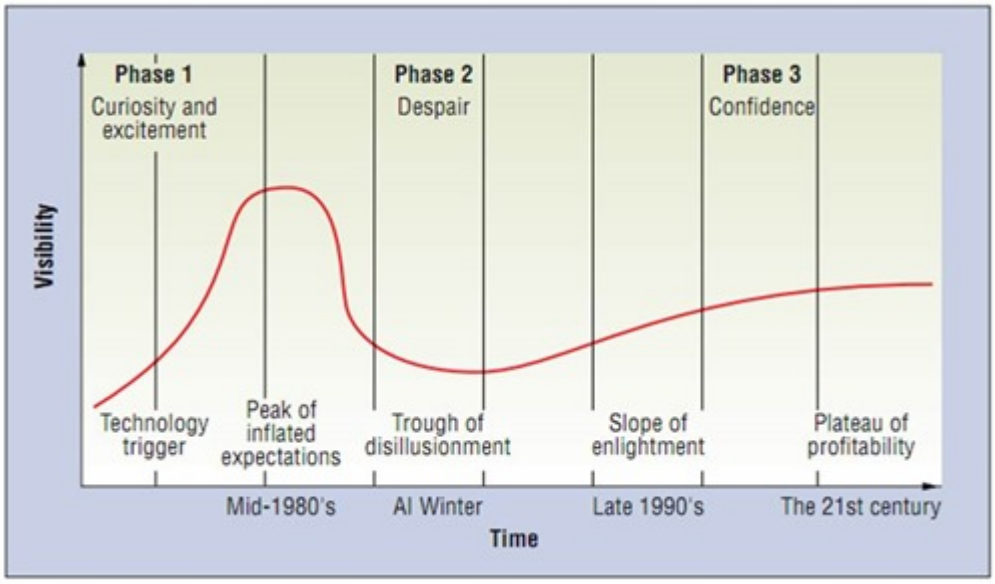
Cubum autem in duos cubos, aut quadratoquadratum in duos quadratoquadratos, et generaliter nullam in infinitum ultra quadratum potestatum in duos ejusdem nominis fas est dividere: cujes rei demonstrationem mirabilem sane detexi. Hanc marginis exiguitas non caperet.

- Pierre de Fermat ~ 1637

Abstract. When Andrew John Wiles was 10 years old, he read Eric Temple Bell's *The Last Problem* and was so impressed by it that he decided that he would be the first person to prove Fermat's Last Theorem. This theorem states that there are no nonzero integers a, b, c, n with $n > 2$ such that $a^n + b^n = c^n$. The object of this paper is to prove that all semistable elliptic curves over the set of rational numbers are modular. Fermat's Last Theorem follows as a corollary by virtue of previous work by Frey, Serre and Ribet.

Herbert Simon prévoyait aussi – toujours en 1958 – qu'en 1978, les machines seraient capables de réaliser toutes les activités intellectuelles humaines. Et la loi de Moore n'existait pas encore puisqu'elle a été énoncée après, en 1965 ! En 1967, Marvin Minsky pensait qu'en une génération, tous les problèmes liés à l'IA seraient résolus. Deux générations plus tard, on en discute encore. Il prévoyait aussi qu'au milieu des années 1970, les ordinateurs auraient l'intelligence d'un homme moyen. Reste à savoir ce qu'est un homme moyen. Les retards étaient aussi manifestes dans la traduction automatique et dans la reconnaissance de la parole. Notons qu'Herbert Simon a été récompensé en 1978 par le Prix Nobel d'économie, pour ses travaux sur les rationalités de la prise de décision, après avoir gagné la fameuse médaille de Turing en 1975. Il n'existe pas de prix Nobel de la prévision ! Il faudrait en général les attribuer à des personnes déjà décédées !

S'en est suivie une période d'enthousiasme au début des années 1980 alimentée notamment par la vague des systèmes experts. Le langage **Prolog** du français Alain Colmerauer a contribué à cette vague. Une nouvelle vague de désillusions s'en est suivie autour des années 1990. L'une des raisons était que le matériel n'arrivait pas à suivre les besoins de l'IA, notamment pour traiter deux besoins clés : la reconnaissance de la parole et celle des images, très gourmandes en puissance de calcul (cf la **source du schéma** ci-dessous).



Lors des années 1980 avaient été lancés divers gosplans d'ordinateurs "de la cinquième génération" dédiés aux applications de l'IA. Cela a commencé avec celui du **MITI Japonais,** lancé en 1981 avec des dépenses d'un milliard de dollars, puis avec le projet anglais **Alvey** lancé à ?350 million et enfin le **Strategic Computing Initiative** de la DARPA. Tous ces projets ont capoté et ont été terminés discrètement. Le projet du MITI visait à faire avancer l'état de l'art côté matériel et logiciel. Ils cherchaient à traiter le langage naturel, à démontrer des théorèmes et même à gagner au jeu de Go. Le projet a probablement pâti d'une organisation trop traditionnelle, linéaire et centralisée. La fin des années 1980 a aussi connu l'effondrement du marché des **ordinateurs dédiés au langage LISP**.

Pendant les années 1990 et 2000 ont émergé de nombreux projets de **HPC** (high-performance computing) très éloignés de l'IA et focalisés sur la puissance brute et les calculs en éléments finis. Ils étaient et sont encore utilisés pour de la simulation, notamment d'armes nucléaires, d'écoulements d'air sur les ailes d'avion ou pour faire des prévisions météorologiques. Les HPC de **Cray Computers** avaient été créés pour cela ! Cette société existe **toujours**. C'est l'une des rares survivantes des années 1970 !

Depuis le début des années 2000, l'IA a été relancée grâce à diverses évolutions :

- L'augmentation de la **puissance du matériel** qui a permis de diversifier la mise en œuvre de nombreuses méthodes jusqu'alors inaccessibles. Et en particulier, l'usage de méthodes statistiques pouvant exploiter la puissance des machines autant côté calcul que stockage et puis, plus récemment, les réseaux neuronaux.

- L'atteinte de diverses **étapes symboliques** marquantes comme la victoire de Deep Blue contre Kasparov en 1997 puis d'IBM Watson dans Jeopardy en 2011. Enfin, il y a quelques jours, la victoire de DeepMind au jeu de Go.

- L'**Internet** qui a créé de nouveaux besoins comme les moteurs de recherche et permis la mise en place d'architectures massivement distribuées.

- La disponibilité de très **gros volumes de données**, via les usages de l'Internet et des mobiles, des objets

connectés ou la génomique, qui permet d'associer les méthodes de force brute et les réseaux neuronaux et autres machine learning ou méthodes statistiques.

- Les **besoins** dans la robotique, dans la conquête spatiale (Curiosity, Philae…), dans les véhicules à conduite assistée ou autonome, dans la sécurité informatique, la lutte contre la fraude et les scams.
- Les **nombreuses applications commerciales** de l'IA croisant le machine learning, les objets connectés, la mobilité et le big data.
- L'adoption de **méthodes** scientifiques et pragmatiques – basées sur l'expérimentation – et transdisciplinaires, par les chercheurs et industriels.

Comme tout domaine scientifique complexe, l'IA n'a jamais été un terrain d'unanimité et cela risque de perdurer. Diverses écoles de pensée se disputent sur les approches à adopter. On a vu s'opposer les partisans du connexionnisme – utilisant le principe des réseaux de neurones et de l'auto-apprentissage – face à ceux du computationnisme qui préfèrent utiliser des concepts de plus haut niveau sans chercher à les résoudre via des procédés de biomimétisme.

On retrouve cette dichotomie dans la **bataille entre "neats" et "scuffies"**, les premiers, notamment John McCarthy (Stanford), considérant que les solutions aux problèmes devraient être élégantes et carrées, et les seconds, notamment Marvin Minsky (MIT) que l'intelligence fonctionne de manière plus empirique et pas seulement par le biais de la logique. Comme si il y avait un écart entre la côté Est et la côte Ouest !

Ces débats ont leur équivalent dans les sciences cognitives, dans l'identification de l'inné et de l'acquis pour l'apprentissage des langues. Burrhus Frederic Skinner est à l'origine du comportementalisme linguistique qui décrit le conditionnement opérant dans l'apprentissage des langues. Noam Chomsky avait remis en cause cette approche en mettant en avant l'inné, une sorte de pré-conditionnement du cerveau des enfants avant leur naissance qui leur permet d'apprendre facilement les langues. En gros, le fonctionnement de l'intelligence humaine est toujours l'objet de désaccords scientifiques ! On continue d'ailleurs, comme nous le verrons dans le dernier article de cette série, à en découvrir sur la neurobiologie et le fonctionnement du cerveau.

D'autres débats ont cours entre les langages de programmation déclaratifs et les moteurs d'inférences utilisant des bases de règles. Sont arrivées ensuite les méthodes statistiques s'appuyant notamment sur les réseaux bayésiens et les techniques d'optimisation. A ce jour, les méthodes les plus couramment utilisées sont plutôt des domaines mathématiques et procéduraux, mais les méthodes à base de réseaux neuronaux et d'auto-apprentissage font leur chemin. L'intelligence artificielle intégrative qui se développe vise à exploiter conjointement toutes les approches.

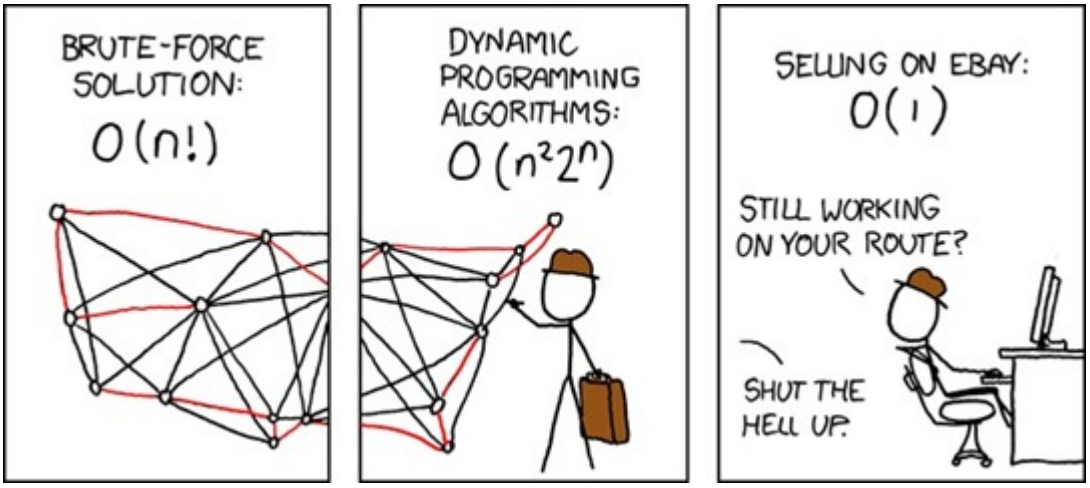
Aujourd'hui, l'IA est aussi l'objet d'un débat de société, philosophique, économique (sur le futur de l'emploi) et donc politique. Les débats ont tendance à trop sortir de la sphère scientifique et technique, au point que, parfois, on ne sait plus de quoi l'on parle ! L'IA est un vaste machin ou tout est mis dans le même sac. On y anthropomorphise à outrance l'IA en imaginant qu'elle imite, remplace et dépasse l'homme. C'est l'une des raisons d'être de ces papiers que d'essayer de remettre quelques pendules à l'heure !

Sur ce, je vais maintenant partir des couches d'abstraction les plus basses (systèmes experts, réseaux neuronaux, machine learning, méthodes statistiques, …) pour ensuite monter dans les couches plus hautes qui font généralement appel aux couches basses, comme dans la reconnaissance de la parole ou des images. Pour chacune de ces techniques, je vais évoquer si besoin est leur ancienneté, les progrès les plus récents, les applications phares ainsi que quelques acteurs des marchés correspondants.

Force brute

La force brute est un moyen courant de simuler l'intelligence humaine ou de la dépasser. Pour un jeu comme les échecs, elle vise à tester toutes les possibilités et à identifier les chemins les plus optimums parmi des zillions de combinaisons. Cela peut fonctionner si c'est à la portée de la puissance de calcul des machines. Ces mécanismes peuvent être optimisés avec des algorithmes d'élagage qui évacuent les "branches mortes" de la combinatoire ne pouvant aboutir à aucune solution. C'est plus facile à réaliser aux échecs qu'au jeu de Go ! La force brute a été utilisée pour gagner aux premiers avec l'ordinateur **Deeper Blue** d'IBM en 1997, calculant 200 millions de positions par seconde. Des réseaux neuronaux ont été exploités pour gagner au Go récemment avec la solution créée par **DeepMind**, la filiale en IA de Google.

La force brute est utilisée dans de nombreux domaines comme dans les moteurs de recherche ou la découverte de mots de passe. On peut considérer que de nombreux pans de l'IA l'utilisent, même lorsqu'ils s'appuient sur des techniques modernes de réseaux neuronaux ou de machine learning que nous traiterons plus loin. Elle ne fonctionne que si la combinatoire reste dans l'enveloppe de puissance de l'ordinateur. Si elle est trop élevée, des méthodes de simplification des problèmes et de réduction de la combinatoire sont nécessaires.



(**source de l'image**)

La force brute s'est aussi généralisée parce que la puissance des ordinateurs le permet : ils tournent plus vite, sont distribuables, le stockage coûte de moins en moins cher, les télécommunications sont abordables et les capteurs de plus en plus nombreux, des appareils photo/vidéo des smartphones aux capteurs d'objets connectés divers.

Moteurs de règles et systèmes experts

Les débuts des moteurs de règles remontent à 1957 quand Alan Newell et Herbert Simon développaient le General Problem Solver (GPS), un logiciel de résolution de problèmes utilisant des règles modélisant les inférences possibles d'un domaine et résolvant un problème en partant de la solution attendue et en remontant vers les hypothèses.

Les moteurs de règles s'appuient sur la notion de raisonnement contraint par des règles. On fournit au moteur un ensemble de règles pouvant par exemple représenter le savoir des experts dans un domaine donné. Avec des règles proches de la programmation logique du genre "si X et Y sont vrais, alors Z est vrai" ou "X entraine Y". On peut alors interroger le système en lui posant des questions genre "est-ce que W est vrai" et il va se débrouiller pour exploiter les règles enregistrées pour répondre à la question. Les moteurs de règles utilisent la théorie des graphes et la gestion de contraintes.

Cette branche de l'IA a été introduite par John McCarthy en 1958. Elle aboutit dans les années 1970 aux travaux de Robert Kowalski de l'Université d'Edinbourg, d'Alain Colmerauer et Philippe Roussel qui sont à l'origine du langage de programmation **Prolog** qui connut ses heures de gloire dans les années 1980. Le **LISP** a été aussi utilisé dans ce domaine. Il s'est même développé une petite industrie avec les ordinateurs spécialisés de **Lisp Machines** et **Symbolics** (1979-2005), et des logiciels d'**Intellicorp** (créé en 1980 et maintenant spécialisé dans les logiciels de gestion d'applications pour SAP).

Les moteurs de règles sont employés dans les systèmes experts, un domaine et un marché qui s'est développé depuis les années 1980. Les systèmes experts ont été notamment théorisés dans le cadre du **Stanford Heuristic Programming Project** en 1980. Ils répondent à des questions dans des domaines spécifiques dont on a codifié la connaissance. Cela permit à l'IA de se rendre utile dans des domaines spécifiques, comme dans la santé. L'approche se heurtait cependant à la difficulté de capter la connaissance des experts. Cela explique son déclin dans les années 1990. Dans de nombreux domaines, la force brute s'est imposée en lieu et place de la logique et de la captation manuelle de connaissances. Cela se retrouve dans le traitement du langage, la traduction automatique, la reconnaissance des images ou les moteurs de recherche. Même IBM Watson utilise la force brute pour exploiter de gros volumes de bases de données de connaissances non structurées.

Un système expert s'appuie sur deux composantes clés : une base de connaissance, générée souvent manuellement ou éventuellement par exploitation de bases de connaissances existantes, puis un moteur d'inférence, plus ou moins générique, qui va utiliser la base de connaissance pour répondre à des questions précises. Les systèmes experts peuvent expliquer le rationnel de leur réponse. La traçabilité est possible jusqu'au savoir codifié dans la base de connaissances.

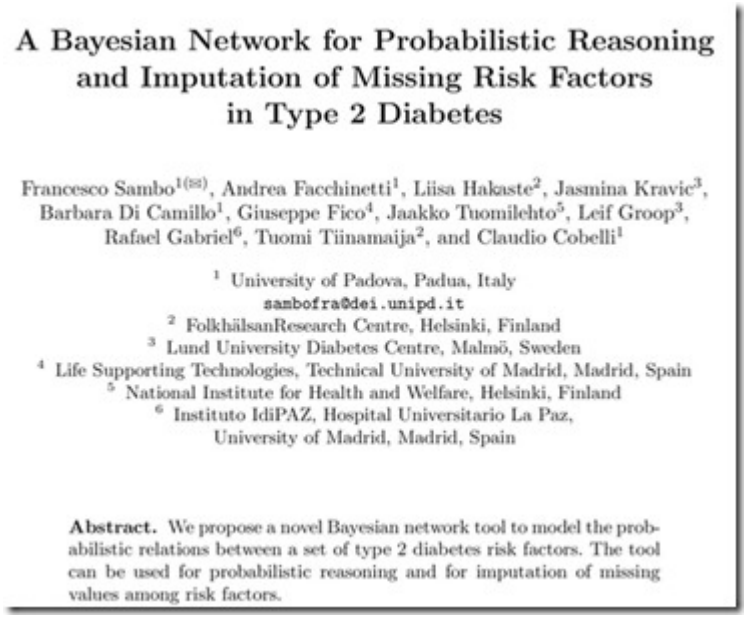
On compte encore des outils et langages dans ce domaine et notamment l'offre du français **ILOG**, acquis en 2009 par IBM et dont les laboratoires de R&D sont toujours à Gentilly près de Paris. Le moteur d'inférence ILOG Jrules est devenu **IBM Operational Decision Manager**. De son côté, ILOG Solver est une bibliothèque C++ de programmation par contraintes, devenue IBM ILOG CPLEX CP Optimizer. Une stratégie de branding moins efficace que celle de IBM Watson, comme nous le verrons dans le prochain article de cette série.

Méthodes statistiques

Les méthodes statistiques et notamment bayésiennes permettent de prévoir la probabilité d'événement en fonction de l'analyse d'événements passés. Les réseaux bayésiens utilisent des modèles à base de graphes pour décrire des relations d'interdépendances statistiques et de causalité entre facteurs.

Les applications sont nombreuses comme la détection de potentiel de fraudes dans les transactions de cartes bancaires ou l'analyse de risques d'incidents pour des assurés. Elles sont aussi très utilisées dans les moteurs de recherche au détriment de méthodes plus formelles, comme le rappelle Brian Bannon en 2009 dans **Unreasonable Effectiveness of Data**.

La plupart des études scientifiques dans le domaine de la biologie et de la santé génèrent des corpus sous forme de résultats statistiques comme des gaussiennes d'efficacité de nouveaux médicaments. L'exploitation de la masse de ces résultats relève aussi d'approches bayésiennes. Le cerveau met d'ailleurs en œuvre une logique bayésienne pour ses propres prises de décision, notamment motrices, les centres associés étant d'ailleurs situés dans le cervelet tandis que dans le cortex cérébral gère la mémoire et les actions explicites (source : **Stanislas Dehaene**).



A Bayesian Network for Probabilistic Reasoning and Imputation of Missing Risk Factors in Type 2 Diabetes

Francesco Sambo[1(✉)], Andrea Facchinetti[1], Liisa Hakaste[2], Jasmina Kravic[3], Barbara Di Camillo[1], Giuseppe Fico[4], Jaakko Tuomilehto[5], Leif Groop[3], Rafael Gabriel[6], Tuomi Tiinamaija[2], and Claudio Cobelli[1]

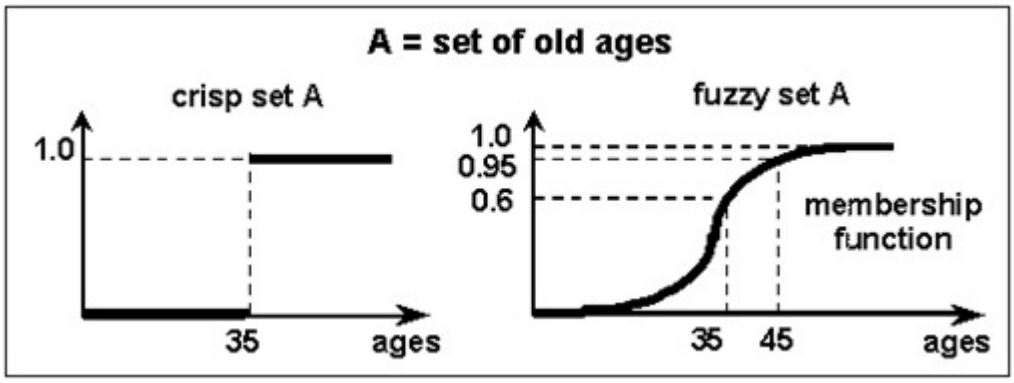
[1] University of Padova, Padua, Italy
sambofra@dei.unipd.it
[2] FolkhälsanResearch Centre, Helsinki, Finland
[3] Lund University Diabetes Centre, Malmö, Sweden
[4] Life Supporting Technologies, Technical University of Madrid, Madrid, Spain
[5] National Institute for Health and Welfare, Helsinki, Finland
[6] Instituto IdiPAZ, Hospital Universitario La Paz, University of Madrid, Madrid, Spain

Abstract. We propose a novel Bayesian network tool to model the probabilistic relations between a set of type 2 diabetes risk factors. The tool can be used for probabilistic reasoning and for imputation of missing values among risk factors.

Logique floue

La logique floue est un concept de logique inventé par l'américain Lofti Zadeh ("Fuzzy Logic") en 1965. J'avais eu l'occasion de l'entendre la présenter lors d'une conférence à l'Ecole Centrale en 1984, lorsque j'étais en option informatique en troisième année. Ca ne nous rajeunit pas !

La **logique floue** permet de manipuler des informations floues qui ne sont ni vraie ni fausses, en complément de la logique booléenne, mais à pouvoir faire des opérations dessus comme l'inversion, le minimum ou le maximum de deux valeurs. On peut aussi faire des OU et des ET sur des valeurs "floues".



Quid des applications ? Elles sont relativement rares. On les trouve dans le contrôle industriel, dans des boites de vitesse chez Volkswagen (pour tenir compte de l'intention "floue" du conducteur), pour gérer des **feux de circulation** et maximiser le débit, dans la reconnaissance de la parole et d'images, le plus souvent, en complément du bayésien. Des **dizaines de milliers de brevets** auraient été déposés pour protéger des procédés techniques utilisant la théorie de la logique floue.

Réseaux de neurones

Les réseaux de neurones visent à reproduire approximativement par bio mimétisme le fonctionnement des neurones vivants avec des sous-ensembles matériels et logiciels capables de faire des calculs à partir de quelques données en entrées et de générer un résultat en sortie. Combinées en grand nombre, les neurones artificiels permettent de créer des systèmes capables par exemple de reconnaître des formes. Les réseaux

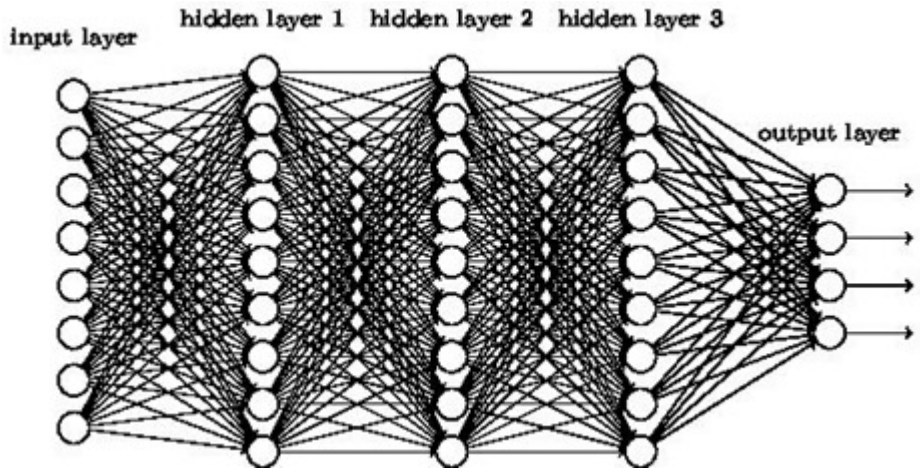
neuronaux les plus intéressants sont ceux qui peuvent faire de l'auto-apprentissage. Attention cependant, les réseaux neuronaux visent l'efficacité algorithmique et ne prétendent pas être des reproductions fines du système nerveux biologique à bas niveau.

Le concept de base est né en 1943 des travaux de Warren McCullochs et Walter Pitts. Donald Hebb ajouta le principe de modulation la connexion entre neurones en 1949, permettant aux neurones de mémoriser de l'expérience. La connaissance est acquise via les interconnexions entre neurones et via un processus d'apprentissage. Elle est matérialisée sous la forme de poids de connexions synaptiques entre neurones qui varient en fonction de l'expérience acquise, par exemple dans la reconnaissance d'images.

Le premier réseau de neurones matériel fut créé par Marvin Minsky et Dean Edmons en 1950. Le SNARC simulait 40 neurones avec 3000 lampes à tubes ! Frank Rosenblatt, un collègue de Marvin Minsky, créa ensuite le concept du **perceptron** en 1957 qui était un neurone assez simple dans son principe. Le premier perceptron était un réseau de neurones artificiels à une seule couche tournant sous forme de logiciel dans un **IBM 704**, le premier ordinateur du constructeur doté de mémoires à tores magnétiques. C'était un outil de classification linéaire utilisant un seul extracteur de caractéristique.

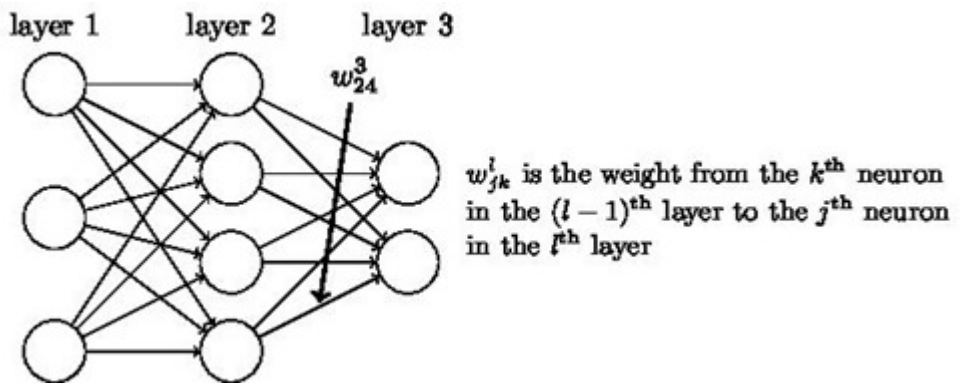
En 1969, Marvin Minsky publia avec Seymour Papert le livre "Perceptrons" qui critiquait sévèrement les travaux de Frank Rosenblatt. D'ailleurs, sur un point très spécifique portant sur les portes logiques XOR des perceptrons. Ce livre mit un coup d'arrêt à ces développements, un peu comme le rapport de Lightfill quelques années plus tard. Toujours, dans la dynamique de la rivalité des *neats* vs *scuffies*. Ce coup d'arrêt fit perdre un temps considérable à l'ensemble des recherches en IA, ce d'autant plus que les réseaux neuronaux sont devenus, depuis, un pan fondamental des progrès dans tous les étages de l'IA. Marvin Minsky reconnu toutefois son erreur dans les années 1980, après le décès de Frank Rosenblatt.

Depuis une vingtaine d'années, les réseaux neuronaux sont mis à toutes les sauces, la dernière étant la victoire de **DeepMind** contre un champion de Go à la mi-mars 2016. Les réseaux neuronaux ont progressé pas à pas, avec la création d'innombrables variantes conceptuelles pour améliorer leurs capacités d'apprentissage et de mémorisation. L'IA progresse d'ailleurs régulièrement et de manière plutôt décentralisée, avec des dizaines de chercheurs contribuant à faire avancer l'état de l'art. Les dernières années ont cependant vu les efforts de recherche passer des travaux dans la logique de base vers ses applications.



L'un des points clés des réseaux de neurones actuels est la technique de la **rétropropagation du gradient** (back propagation) qui corrige les défauts des Perceptrons identifiés par Papert et Minsky. Elle a vu le jour dans les **années 1960** puis, pendant et après le second hiver de l'IA, a repris son essor vers 1986. Elle permet de modifier le poids des liaisons synaptiques entre neurones en fonction des erreurs constatées dans les évaluations précédentes, par exemple dans la reconnaissance d'images. Comment fonctionne cette boucle d'apprentissage ?

C’est un apprentissage soit assisté, soit automatique en comparant les résultats avec la bonne réponse, déjà connue. C’est un des débats clés d’aujourd’hui : est-on réellement capable de créer des réseaux doués de facultés d’auto-apprentissage ? Il semblerait que l’on en soit encore loin.

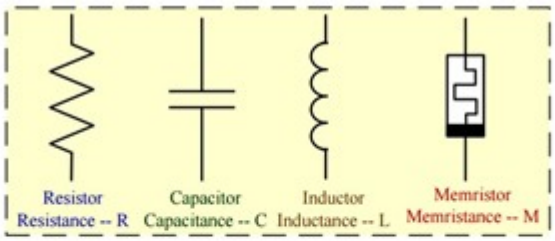


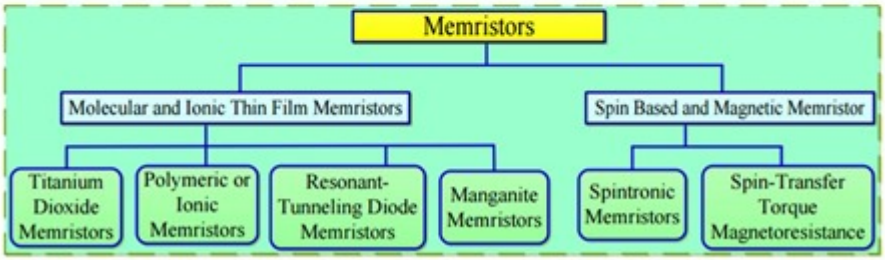
20 ans après la renaissance des réseaux neuronaux, en 2006, le japonais Osamu Hasegawa créait les réseaux neuronaux auto-organisés incrémentalement (“Self-Organising Incremental Neural Network” ou SOINN), utilisables dans des réseaux neuronaux auto-réplicables et capables d’auto-apprentissage. En 2011, son équipe développait un robot utilisant ces SOINN capable d’auto-apprentissage (**vidéo**), illustrant magistralement les applications des réseaux neuronaux. Nous sommes 10 ans plus tard, et on constate que les robots autonomes sont encore loin du compte, même si les sociétés telles que Boston Dynamics, filiale de Google, nous ébaubissent avec des robots très souples dans leur démarche et résistant à l’adversité.

Les réseaux neuronaux ont aussi progressé grâce à leur mise en œuvre dans des architectures matérielles spécialisées permettant de bien paralléliser leurs traitements comme le fait le cerveau. Le composant électronique idéal pour créer un réseau de neurones est capable d’intégrer un très grand nombre de pico-unités de traitement avec entrées, sorties, logique de calcul si possible programmable et mémoire non volatile. Il faut par ailleurs que les connexions entre neurones (synapses) soient les plus nombreuses possibles. En pratique, les connexions se font avec les neurones adjacents dans les circuits.

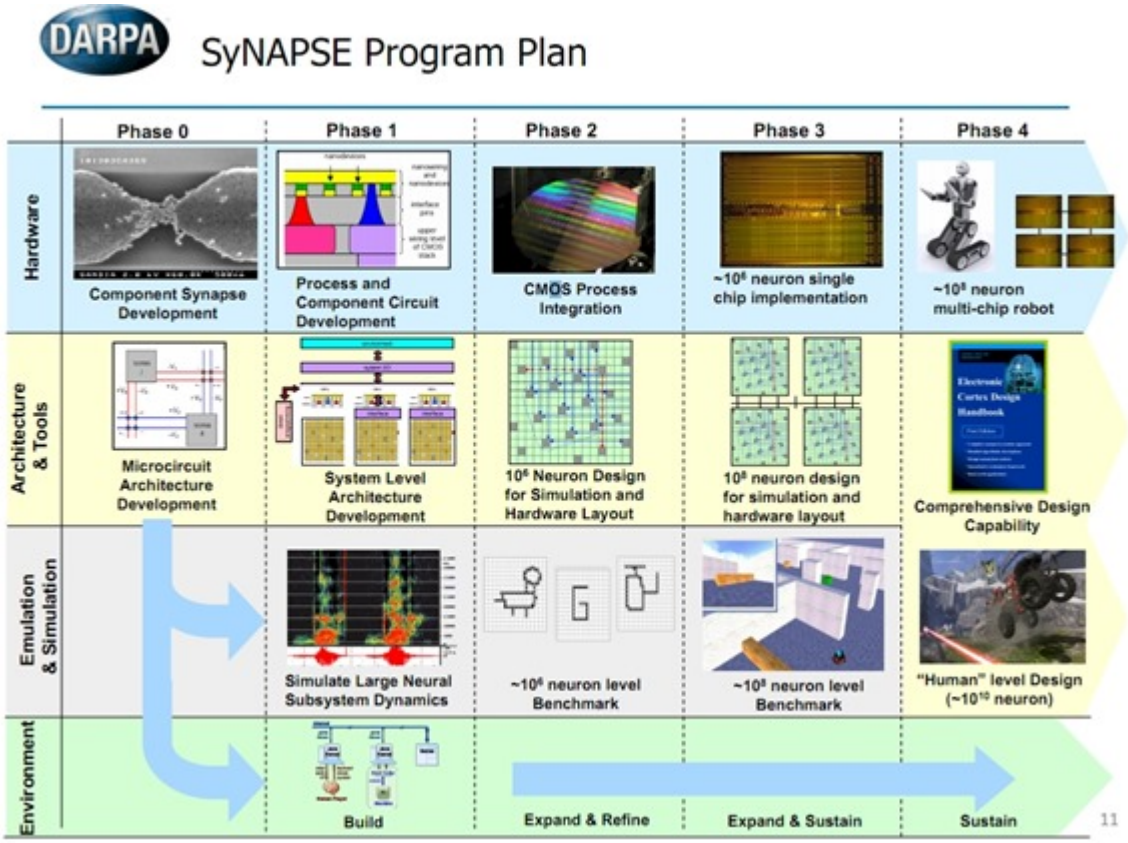
Les **memristors** ont fait son apparition en 2008 chez HP après avoir été conceptualisée en 1971 par le sino-

américain **Leon Ong Chua**. Ce sont des composants électroniques capables de mémoriser un état en faisant varier leur résistance électrique par l'application d'une tension. Un peu comme les cristaux liquides bistables qui servent dans (feu) les liseuses électroniques. La valeur modifiable de la résistance permet de stocker de l'information. Les memristors peuvent aussi être intégrés au côté de composants actifs classiques dans des unités de traitement. C'est très bien expliqué dans **Memristor: From Basics to Deployment** de Saraju Mohanty, publié en 2013, d'où sont extraits les deux schémas ci-dessous. Le second présente les différents types de memristors actuellement explorés. Ces composants sont intégrables dans des puces au silicium utilisant des procédés de fabrication plus ou moins traditionnels (**nanoimprint lithography**), en ajoutant une bonne douzaine d'étapes dans la production, et avec des matériaux rares comme les oxydes de titane.





Les memristors ont été développés dans le cadre des projets de recherche du programme **SyNAPSE** de la DARPA. **HP** a été le premier à en prototyper en 2008, avec de l'oxyde de titane. Il en existe de plusieurs types, pouvant généralement être fabriqués dans les lignes de productions de chipsets CMOS traditionnelles, mais avec des procédés spécifiques de dépôt sous vide de couches minces de matériaux semi-conducteurs. HP a même lancé un partenariat avec le fabricant de mémoires **Hynix**, mais le projet a été mis en veilleuse **en 2012**. Le taux de rebus serait trop élevé lors de la fabrication. C'est un paramètre clé pour pouvoir fabriquer des composants en quantité industrielle et à un prix de vente abordable. De plus, le nombre de cycles d'écriture semblait limité pour des raisons chimiques, dans le cycle de libération/captation d'oxygène pour les memristors en oxydes de titane.



En octobre 2015, HP et **SanDisk** ont cependant annoncé un partenariat pour fabriquer des mémoires volatiles et non volatiles à base de memristors, censées être 1000 fois plus rapides et plus endurantes que les mémoires flash traditionnelles.

D'autres laboratoires de recherche et industriels planchent aussi sur les memristores et les réseaux de neurones matériels :

- **IBM** planche avec l'**ETH** de Zurich (le CNRS suisse) sur des ordinateurs à base de memristors. Ce même ETH développe un **memristor** capable de stocker trois états à base de pérovskite (titanate de calcium) de 5 nm d'épaisseur. Cela pourrait servir à gérer de la logique floue.

- Des chercheurs de l'Université Technologique du Michigan ont **annoncé début 2016** avoir créé des memristors à base de bisulfite de molybdène qui ont un comportement plus linéaire.

- Des **chercheurs du MIT** ont annoncé début 2016 leurs travaux sur le chipset Eyeriss utilisant des neurones spécialisés réparties dans 168 cœurs dotés de leur propre mémoire. Mais visiblement sans memristors. L'application visée est la reconnaissance d'images. Le projet est financé par la DARPA.

- Le projet **Nanolitz** aussi financé par la DARPA dans le cadre des projets Atoms to Product (A2P) et s'appuie sur des fils microscopiques pour connecter plus efficacement des cœurs et neurones dans des circuits spécialisés.

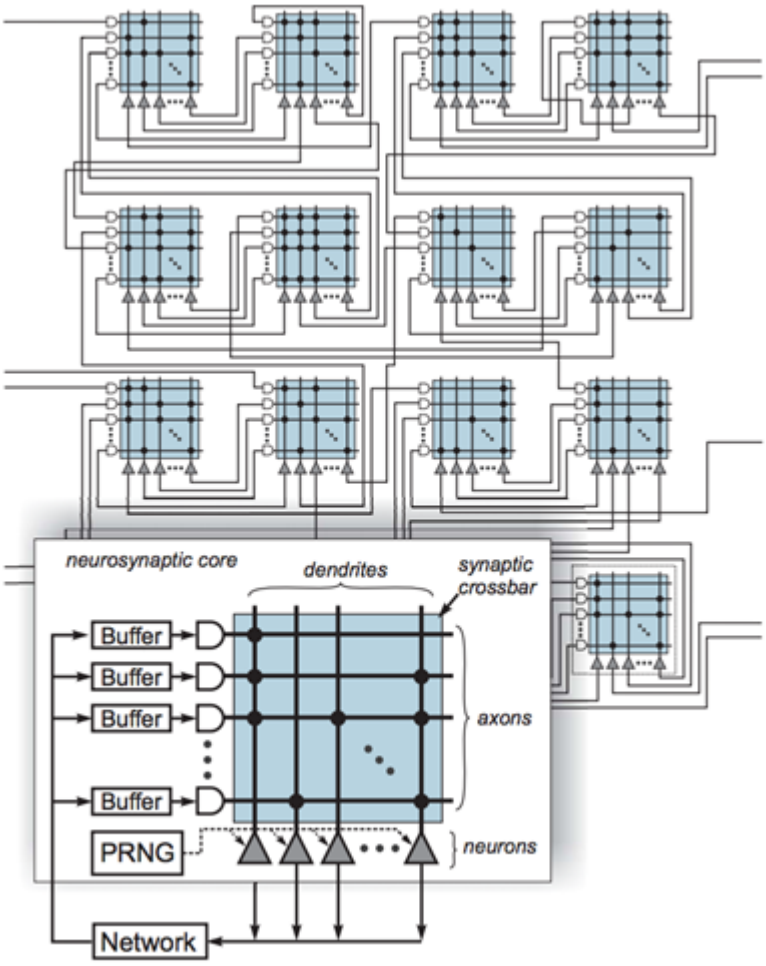
- L'ANR a financé le projet collaboratif **MHANN** associant l'INRIA, l'IMS de Bordeaux et Thalès pour créer des memristors ferriques. Le projet devait être terminé en 2013 et avait bénéficié d'une enveloppe de 740 K€. Difficile de savoir ce qu'il en est advenu en ligne.

- Enfin, la start-up californienne **Knowm** a lancé le **premier composant commercial** à base de memristors,

fabriqué en partenariat avec la Boise State University, à base d'argent ou de cuivre et au prix de $220. Il est destiné en premier lieu aux laboratoires de recherche en réseaux neuronaux.

Le programme SyNAPSE de la DARPA a en tout cas aboutit en 2014 à la création par IBM de ses processeurs neuronaux **TrueNorth** capables de simuler un million de neurones artificiels, 256 millions de synapses reliant ces neurones et exécutant 46 milliards d'opérations synaptiques par secondes et par Watt consommé. Le tout avec 4096 cœurs. Le chipset a été fabriqué par Samsung en technologie CMOS 28 nm et avec une couche d'isolation SOI (issue du français SOITEC !) permettant de diminuer la consommation électrique et d'accélérer les traitements. Le chipsets comprend 5,4 milliards de transistors en tout et fait plus de 4 cm2 de surface. Et surtout, il ne consomme que 70 mW, ce qui permet d'envisager d'empiler ces processeurs en couches, quelque chose d'impossible avec les processeurs CMOS habituels qui consomment beaucoup plus d'énergie. A titre de comparaison, un processeur Intel Core i7 de dernière génération (Skymake) réalisé en technologie 14 nm consomme entre 15 W et 130 W selon les modèles, pour 1,7 milliards de transistors.

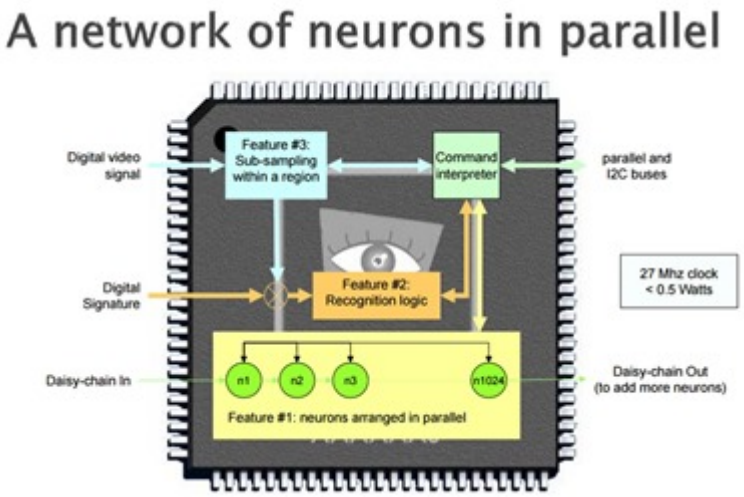
Le but d'IBM est de construire un ordinateur doté de 10 milliards de neurones et 100 trillions de synapses, consommant 1 KW et tenant dans un volume de deux litres. A titre de comparaison, un cerveau humain contient environ 85 milliards de neurones et ne consomme que 20 Watts ! Le biologique reste encore à ce stade une machine très efficace d'un point de vue énergétique !



Il existe d'autres projets d'ordinateurs synaptiques à base de réseaux de neurones. On peut notamment citer le projet de Jeff Hawkins, le fondateur de Palm, celui de Stanford, qui travaille sur le chipset **Neurocore** intégrant pour l'instant 65536 neurones et fonctionnant à très basse consommation.

Il y a aussi le projet **SpiNNaker** de Steve Furber (Université de Manchester, UK), qui vise à créer un chipset de un milliard de neurones. Il s'appuie cependant sur une architecture matérielle classique, avec 18 cœurs 32 bits ARM par chip. On est plus dans l'architecture massivement parallèle avec des milliers de processeurs de ce type que dans les processeurs véritablement synaptiques.

Enfin, dans le domaine commercial, le **CogniMem** CM1K est un chipset ASIC intégrant un réseau de 1024 neurones qui sert aux applications de reconnaissance des formes. Ne coutant que $94, il est notamment utilisé dans la **BrainCard**, issue d'une start-up française.



Plus récemment, **Nvidia** a présenté au CES 2016 sa carte PX2 pour l'automobile qui intègre deux processeurs X1 comprenant 256 GPU. Les GPU Nvidia sont utilisés pour simuler des réseaux de neurones. C'est bien mais probablement pas aussi optimal que de véritables réseaux de neurones et de synapses artificiels comme le TrueNorth d'IBM. Qui plus est, la carte PX2 doit être réfrigérée par eau car elle consomme plus de 200 W. Comme l'explique **Tim Dettmers**, un GPU n'est utilisable pour des réseaux de neurones que si la mémoire est facilement partagée entre les cœurs de GPU. C'est ce que propose justement Nvidia avec son architecture **GPUDirect** RDMA.

Il y aussi **Movidius** qui propose ses chipsets Myriad à base de réseaux neuronaux exploitant des processeurs vectoriels, dédiés au traitement de l'image. Ils en ont récemment lancé une version qui tient sur une clé USB, la Fathom Neural Compute Stick.

On peut donc constater que tout cela bouillonne, plutôt au niveau des laboratoires de recherche à ce stade, et que l'industrialisation prendra encore un peu de temps, mais que les réseaux neuronaux matériels ont probablement un bel avenir devant eux.

Support Vector Machines

Il faudrait aussi citer les **SVM (**Support Vector Machines) ou “Machine à vecteurs de support”, une autre technique d’apprentissage supervisé plus adaptée à certaines classes de problèmes que les réseaux neuronaux. Les SVM “scalent” mieux que ces derniers dans des architectures distribuées.

Ils peuvent servir par exemple à générer une classification automatique d’échantillons de données complexes, comme une segmentation client ou la classification de termes textuels. On les utilise aussi pour prédire la valeur numérique d’une variable. L’approche concurrence celle des réseaux bayésiens.

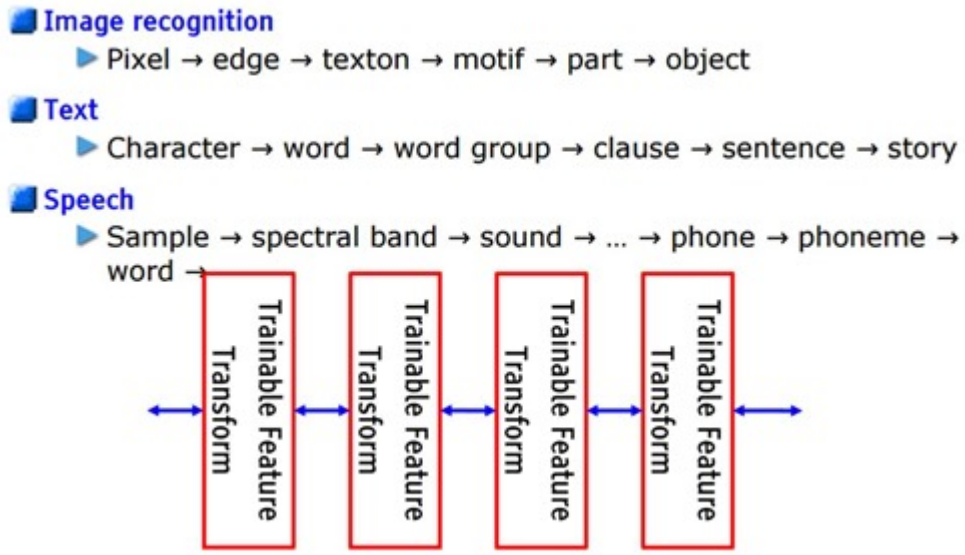
Machine learning et deep learning

Le vaste domaine du machine learning, ou apprentissage automatique, vise à faire des prédictions à partir de données existantes. C’est un domaine qui est intimement relié à celui des réseaux de neurones, qui servent de substrat pour les traitements. En effet, les outils de machine learning et de deep learning s’appuient sur différentes variantes de réseaux de neurones pour leur mise en œuvre pratique, notamment des réseaux neuronaux à plusieurs niveaux. Ces réseaux sont supervisés ou pas selon les cas.

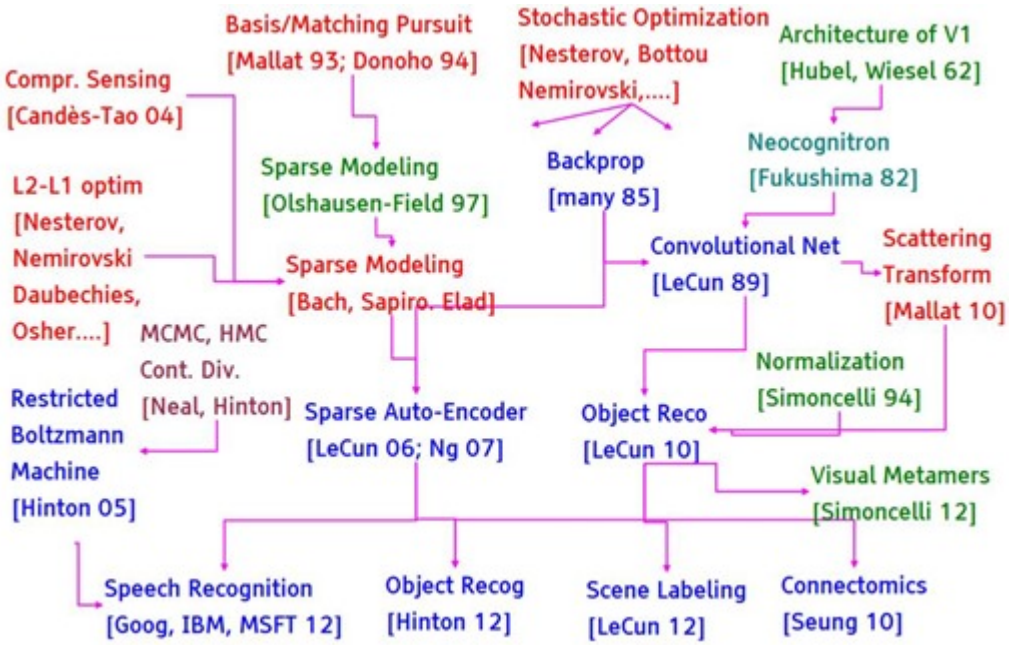
Le machine learning est surtout utilisé aujourd’hui pour la reconnaissance des formes dans les images et celle de la parole, donc dans les sens artificiels. Il peut aussi servir à exploiter des données non structurées et à gérer des bases de connaissances. IBM liste quelques-unes de ces applications dans **son marketing**. On y retrouve des études de cas dans l’éducation pour créer des MOOC auto-adaptatifs, dans le retail avec un assistant d’achats, dans la santé avec la personnalisation de traitements contre certains cancers ou encore dans l’analyse de diverses données dans la smart city.

Les réseaux neuronaux ont connu un renouveau en 2006 avec les travaux des canadiens Geoffrey Hinton et Simon Osindero et du singapourien Yee-Whye Teh publiés dans **A Fast Learning Algorithm For Deep Belief Nets** qui optimisent le fonctionnement des réseaux neuronaux multicouches. Le concept du machine learning a été ensuite formalisé par Geoffrey Hinton en 2007 dans **Learning multiple layers of representation**. Il s’appuyait lui-même sur les travaux du français **Yann LeCun** (en 1989) qui dirige maintenant le laboratoire de recherche en IA de Facebook et de l’allemand **Jürgen Schmidhuber** (1992) dont deux des anciens étudiants ont créé la start-up **DeepMind** maintenant filiale de Google. Petit monde ! Geoffrey Hinton travaille pour **Google** depuis 2013, pas loin du légendaire **Jeff Dean**, arrivé en 1999 et qui planche maintenant aussi sur le deep learning**.**

Pour comprendre le fonctionnement du deep learning, il faut avoir beaucoup du temps et un bon bagage mathématique et logique ! On peut commencer par parcourir **Deep Learning in Neural Networks** de ce Jürgen Schmidhuber, publié en 2014 qui fait 88 pages dont 53 de bibliographie ou bien **Neural Networks and Deep Learning**, un livre gratuit en ligne qui expose les principes du deep learning. Il explique notamment pourquoi l’auto-apprentissage est difficile. Bon, cela fait tout de même plus de 200 pages en corps 11 et on est largué à la cinquième page, même avec un bon background de développeur ! Il y a aussi **Deep Learning Methods and Applications** publié par Microsoft Research (197 pages) qui démarre en vulgarisant assez bien le sujet. Et puis **Artificial Intelligence A Modern Approach**, une somme de référence sur l’IA qui fait la bagatelle de 1152 pages et qui ne serait que le B-A-BA pour les étudiants en informatique. J’ai enfin trouvé cette présentation plutôt synthétique **A very brief overview of deep learning** de Maarten Grachten en 22 slides ! Ouf ! Bref, il faut se taper l’équivalent de plusieurs **Rapports du CES de Las Vegas** !



Quid du **deep learning** ? C'est une variante avancée du machine learning qui s'appuie sur des architectures en couches utilisant des "Restricted Boltzmann Machines" qui s'alimentent les unes les autres. Une machine de deep learning possède plus de couches dites cachées dans ses réseaux de neurones qu'une architecture de machine learning classique. Le deep learning (apprentissage profond) permet d'élever le niveau d'abstraction du machine learning en exploitant des concepts de plus haut niveau, comme indiqué dans le schéma ci-dessus (source : l'excellent slideshow de **LeCun – Ranzato**, qui fait 204 slides, source également de l'historique du deep learning ci-dessous ! Voir aussi la **conférence inaugurale** de Yann LeCun au Collège de France en février 2016 où il excelle dans la vulgarisation). Ce fonctionnement imite d'ailleurs celui du cerveau humain qui utilise plusieurs niveaux d'abstraction. Il permet en théorie de mettre en place des méthodes d'auto-apprentissage sans base de données d'entrainement. Pour l'instant, cependant, cela semble surtout concerner les sens artificiels.



Cette complexité du sujet m'intrigue particulièrement. J'ai pu récemment explorer des sujets aussi variés que les réseaux M2M, la biologie moléculaire ou la génomique et ils m'ont semblé bien plus abordables que l'IA, même en parcourant des ouvrages de spécialistes ! C'est dire ! Cette difficulté d'appréhender la science derrière l'IA a probablement un lien avec les fantasmes qui lui sont associés. On imagine à fois le meilleur et le pire de ce que l'on ne peut pas expliquer avec son espace restreint de connaissances.

Cela permet aussi de mettre l'IA à toutes les sauces dans le marketing. L'appellation d'IA est encore utilisée pour valoriser certaines offres mais le machine learning l'est tout autant maintenant. Derrière le marketing, il reste à comprendre ce que le fournisseur a réellement produit : a-t-il assemblé des briques logicielles existantes (souvent en open source), a-t-il créé des briques spécifiques, a-t-il juste entraîné un modèle, la solution est-elle une simple application directe de techniques existantes ? C'est ce que nous verrons dans l'épisode suivant !

Reconnaissance de la parole

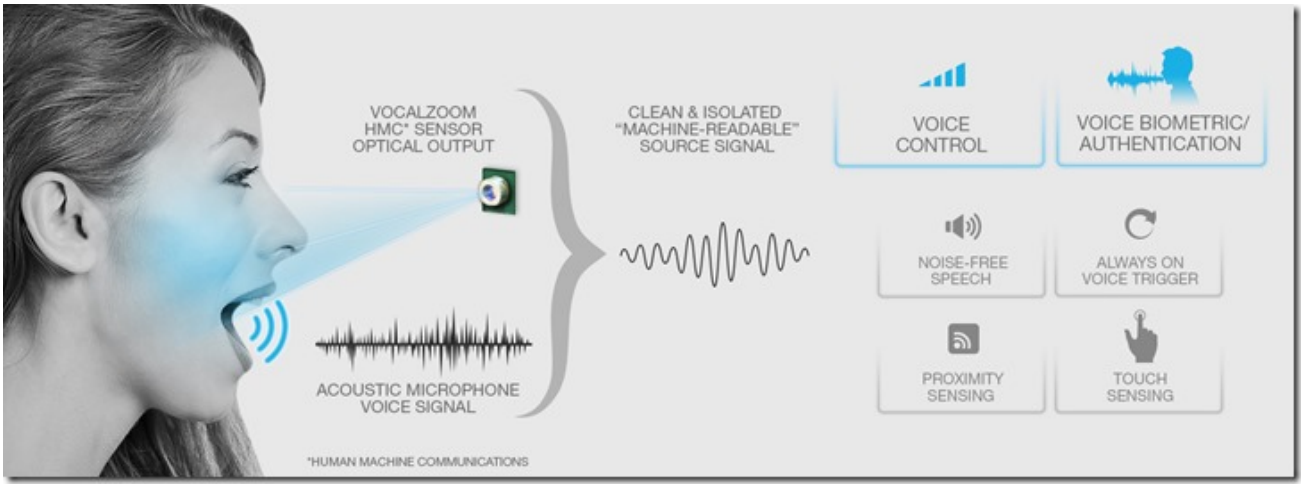
C'est une technologie commune et disponible dans les applications grand public. Le marché est dominé par de grands acteurs américains (OK **Google**, **Microsoft** Cortana, **Apple** Siri, **Amazon** Alexa) et, en OEM, par **Nuance** qui vend sa solution un peu partout. Apple a fait l'acquisition de la start-up **VocaliQ** en 2015 et **Sensory** fait avancer l'état de l'art de manière indépendante depuis plus de 20 ans.

La reconnaissance de la parole s'appuyait au départ sur des techniques statistiques et notamment bayésiennes. Elle a fait des progrès continus grâce à l'intégration de techniques différentes telles que le deep learning, le big data, les réseaux neuronaux et des modèles de Markov à base de statistiques. Elle profite aussi de l'augmentation régulière de la puissance des processeurs et notamment des processeurs mobiles. **Nvidia** propose depuis peu d'exploiter les fonctions GPU de ses chipsets pour mettre en œuvre des techniques de deep learning, bénéficiant du fort parallélisme des nombreux GPU disponibles. Les solutions de reconnaissance vocale ont souvent besoin d'accéder à des bases de données de référence, surtout s'il fonctionne sans apprentissage de la voix de l'utilisateur. Cela nécessite un aller et retour avec les serveurs du service, ce qui se sent si on utilise un smartphone. D'où l'intérêt de la 4G et de son débit comme de son faible temps de latence pour les allers et retours avec les serveurs.

Pour en savoir plus, voir cet historique de la recherche en reconnaissance de la parole : **Survey of Technical Progress in Speech Recognition by Machine over Few Years of Research** parue en 2015. Ce sujet se retrouve à la fin de cet article car comme les suivants, il intègre de nombreuses branches du savoir issu de plusieurs décennies de recherches dans l'IA.

On est encore **loin de la solution parfaite**, notamment parce que les logiciels manquent d'informations sur le contexte des conversations (voir aussi **Why our crazy smart AI still sucks in transcribing speech** paru dans Wired en avril 2016). Le taux de fiabilité n'est jamais de 100%. Il ne l'est d'ailleurs jamais pour l'homme également ! Par exemple, Microsoft Cortana atteint un taux d'erreur d'environ 8%, soit le double de celui de l'homme. Ce taux d'erreurs de l'IA diminuerait de 25% par an. Microsoft prévoit d'atteindre le taux d'erreur humain d'ici quelques années. Et encore, c'est pour l'anglais ! Le taux d'erreur est toujours plus élevé dans d'autres langues comme le chinois. D'où l'intérêt de la récente publication en open source de la solution **Deep Speech 2** de **Baidu**.

Le taux d'erreur est particulièrement élevé s'il y a du bruit ambiant, comme dans la rue, dans un endroit où il y a du monde et même dans sa voiture. Des techniques de captation du son et d'élimination du bruit ambiant existent aussi. Certaines portent sur l'analyse spectrale et le filtrage de fréquences. D'autres utilisent la captation stéréophonique pour séparer le bruit proche (différentié) du bruit lointain (qui l'est moins). J'avais même vu la start-up israélienne **VocalZoom** au CES 2015 qui utilisait un laser pour capter les vibrations des lèvres. Il faut juste trouver où placer le laser, ce qui est plus facile sur les installations fixes.

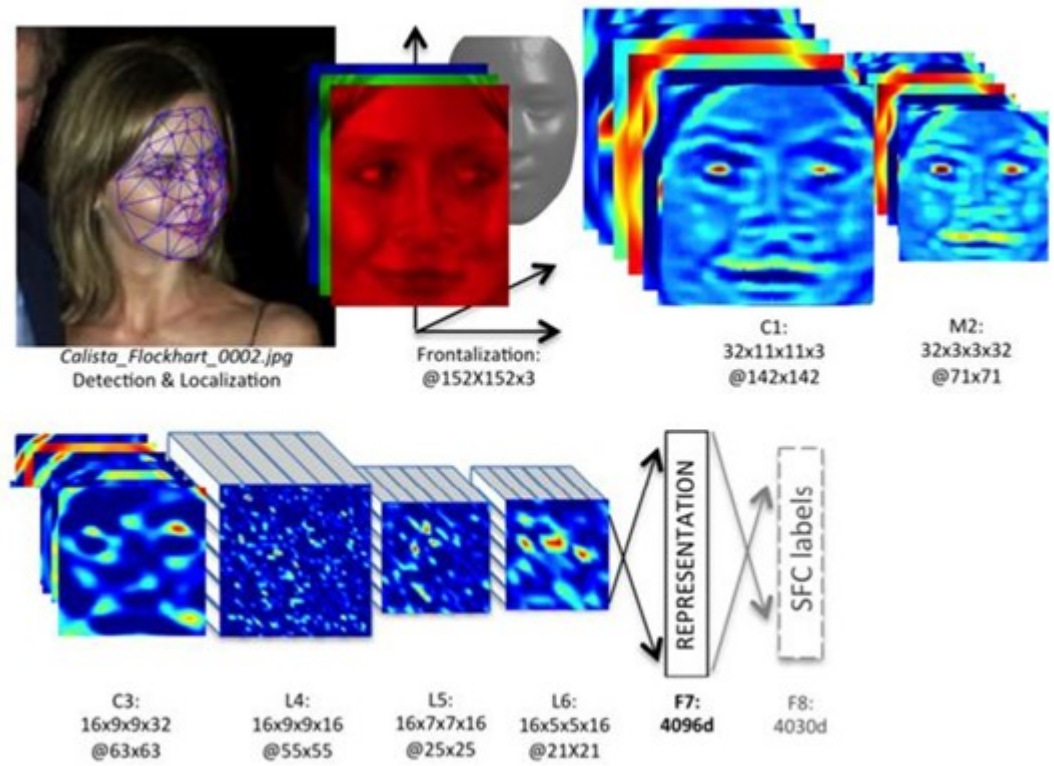


Reconnaissance d'images

Cette fonction est devenue aussi critique que la reconnaissance de la parole et notamment dans les moteurs de recherche et certains réseaux sociaux, pour identifier des visages, des expressions ainsi que des lieux. Elle est aussi présente depuis des décennies dans les logiciels d'OCR pour reconnaitre les textes, images et schémas de documents scannés. Nous avions même un leader en France dans le domaine avec la société **LTU**, rachetée par le japonais Jastec en 2005.

La reconnaissance d'images est l'une des principales applications des réseaux neuronaux, du machine learning et du deep learning comme nous l'avons vu précédemment. L'un des objectifs de la recherche est d'élever au maximum le niveau sémantique de la reconnaissance, pour identifier les personnes et objets sur les images. Par exemple, dans le cas des solutions de **Nvidia** ou **Mobileye** pour la conduite assistée, il s'agit de détecter au pixel près ceux qui correspondent à des piétons, des cyclistes, des véhicules, de la signalisation au sol et des panneaux de signalisation.

Google est évidemment friand de ce genre de technologies qu'ils utilisent dans Google Image et Google Photo. Google Image est capable (avec le glisser-déplacer) d'identifier des images similaires à celle que l'on fournit. Cela utilise probablement une méthode simple de création de hash-code sur les photos et de recherche dans l'index d'une grande base de données. Dans leur projet FaceNet, Google annonce avoir atteint un taux de réussite de détection de visage de 99,63%. Voir **FaceNet: A Unified Embedding for Face Recognition and Clustering**. publié en juin 2015. Le tout s'appuie sur un réseau neuronal à 22 couches.



De son côté, **Facebook** et son projet DeepFace s'appuie sur la technologie issue d'une start-up israélienne **face.com**. Son taux de réussite serait de 97,25% pour vérifier qu'une personne sur une photo est la même sur une autre, quel que soit l'angle de la prise de vue et l'éclairage. C'est juste en-dessous du taux de reconnaissance humain qui serait évalué à 97,5%.

On trouve de la détection de visages dans plein de solutions du marché comme avec la fonction Faces de **Apple** iPhoto. Elle provient peut-être de la start-up suédoise **Polar Rose** acquise par Apple en 2010. De manière peu surprenante, Apple a aussi acquis, début 2016, la start-up **Emotient**, spécialisée dans la reconnaissance d'émotions faciales à base de machine learning. Le matching de visages est une chose, mais détecter les émotions en est une autre et on peut s'attendre à ce qu'Apple utilise cette fonctionnalité dans les évolutions de ses solutions, notamment dans la visioconférence Facetime. Les APIs en cloud proposées par Microsoft Research dans le cadre de son **projet Oxford** apportent des services équivalents aux développeurs d'applications. **Google** fait de même avec ses **Cloud Vision APIs**. Cette abondance des offres rappelle que les technologies de l'IA, une fois au point, deviennent rapidement des commodités. Les méthodes sont sur la place publique. Il faut ensuite les mettre en œuvre avec du logiciel et du matériel. La différence se situe dans l'implémentation et aussi dans le marketing.

La reconnaissance des visages est évidemment un sujet chaud pour les services de sécurité. On en voit dans tous les films et séries TV ! En quelques secondes, les suspects sont identifiés. Est-ce comme cela dans la vraie vie ? Probablement pas. Cela explique pourquoi le **FBI** a lancé son projet NGI (Next Generation Identification) en 2009 et maintenant opérationnel. Il était pourvu à hauteur de la bagatelle de $1B et réalisé par Lockheed Martin.

Le marché de la reconnaissance faciale est aussi prolixe en solutions diffusées en OEM, comme **imagga** (seulement $300K de levés) et ses API en cloud de tagging automatique d'images en fonction de leur contenu, **Cognitec** qui vise surtout les marchés de la sécurité, **Cortexica** ($6,6m de levés) et son logiciel findSimilar en cloud qui met en œuvre ces techniques pour le retail et **Slyce** qui cible aussi le marché du retail ($37m de levés + IPO en avril 2015).

A noter un autre domaine connexe, celui de la reconnaissance de l'écriture manuscrite à partir d'encre digitale, saisie par exemple avec un stylet comme sur les tablettes. Ce marché est moins connu que pour la reconnaissance vocale ou d'images. Et nous y avons un champion français, la société **MyScript**, anciennement Vision Objects, qui est basée à Nantes et qui vend notamment son logiciel à Samsung.

Reconnaissance de vidéos

La reconnaissance de vidéos est une évolution naturelle de la reconnaissance d'images, à ceci près que les vidéos fournissent plus d'information. Elle elle est utile dans tout un tas de contexte, notamment pour les voitures à conduite automatique, un domaine où l'ordinateur **peut maintenant dépasser l'homme**.

De son côté, **Facebook** sait reconnaitre un sport dans une vidéo en s'appuyant sur des réseaux neuronaux. Quant à **Google** Brain, il est capable d'identifier des chats dans des vidéos mais avec un taux d'erreurs encore très élevé, de l'ordre de 25%. La reconnaissance des visages est précise à 81,7% près (**source**). Il faut un début à tout !

On trouve des solutions de reconnaissance de visage dans les vidéos chez **Kairos** qui savent aussi analyser les émotions et quantifier les foules, chez **KeyLemon** ($1,5m de levés) qui propose une solution en cloud, chez **Clarifai** ($10m de levés) qui permet notamment de faire de la curation de contenus photo et vidéo, ou chez le japonais **NEC**. Il faut aussi citer **OpenCV**, une solution open source de détection de visages. Voir **cette liste** de solutions pour développeurs de détection de visages dans les vidéos.

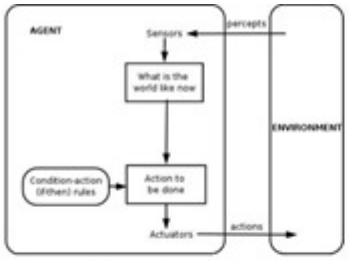
Agents intelligents et réseaux d'agents

Dans ce concept apparu dans les années 1990, les agents intelligents permettent de résoudre des problèmes dans des architectures distribuées. Conceptuellement, un agent est un logiciel ou un matériel qui capte de l'information, décide d'agir rationnellement en fonction des données récupérées et déclenche un action pour optimiser ses chances de succès. Si c'est du matériel, il comprendra des capteurs et des actuateurs. Mais il peut n'être que du logiciel et obtenir des données brutes en entrées et générer des données en sortie. Un agent réagit donc en fonction de l'environnement et en temps réel. Les agents intelligents sont intégrés dans des systèmes distribués dénommés systèmes multi-agents avec des agents autonomes, mais reliés et collaborant entre eux.

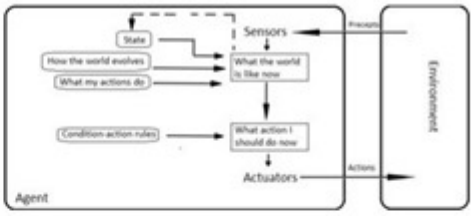
On compte notamment les **Distributed Problem Solving** (DPS) qui découpent un problème en sous-problèmes qui sont résolus de manière coopérative entre plusieurs agents reliés les uns aux autres. Ces systèmes sont conçus pour résoudre des problèmes bien spécifiques.

Les agents sont classifiés par Russell & Norvig dans **Artificial Intelligence – A Modern Approach** (2003-2009) en **types distincts** selon leur niveau d'autonomie et leur mode de prise de décision :

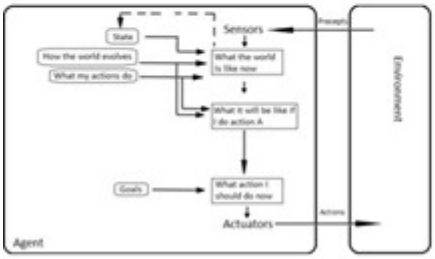
- Les **simple reflex agents** qui comprennent des capteurs, des règles indiquant quelle action mener et des actuateurs pour les déclencher. Ils travaillent en temps réel.
- Les **model based reflex agents** qui ajoutent un moteur d'état capable de mémoriser dans quel état se trouve l'objet et qui évaluent l'impact des actions pour changer d'état.
- Les **goal-based agents** qui prennent leur décision en fonction d'un objectif et déterminent une action pour l'atteindre.
- Les **utility-based agents** qui prennent leur décision en fonction d'un but à atteindre qui est plus général.
- Les **learning agents** qui contiennent une fonction d'auto-apprentissage.



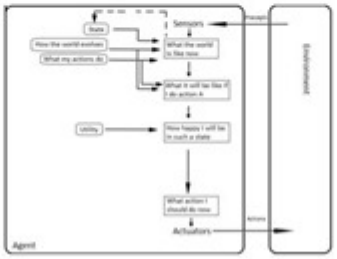
Simple reflex agent



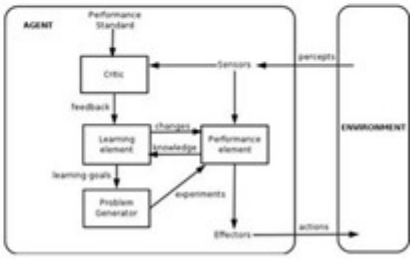
Model based reflex agent



Model-based, goal-based agent



Model-based, utility-based agent



General learning agent

Vu de haut, les réseaux d'agents ressemblent aux réseaux de neurones mais leur mode de fonctionnement est différent. Un agent peut très bien être lui-même individuellement construit avec un réseau de neurones pour réaliser une tâche spécifique comme la reconnaissance de la parole ou d'images. Un autre agent va utiliser le texte généré par la reconnaissance puis appliquer un processus de reconnaissance sémantique, puis un autre va traiter la question, fouiller dans une base de données ou de connaissance, récupérer des résultats, un autre va formuler une réponse et la renvoyer à l'utilisateur. Idem pour un système de traduction automatique qui va d'abord analyser la parole avec un premier agent, puis réaliser la traduction avec un second, puis utiliser un troisième agent de "text to speech" pour transformer le résultat de manière audible.

Le robot Nao d'Aldebaran a une belle capacité de mouvement grâce à une mécanique de bon niveau. Il interagit en parlant avec l'utilisateur, mais de manière encore limitée. Son grand frère Pepper, est doté d'une capacité à capter les émotions des humains qu'il a en face de lui mais sa capacité de dialogue est encore approximative.

Un robot autonome est aussi un condensé de nombreux agents qui gèrent différents niveaux d'abstraction avec de nombreux capteurs, de la mécanique, des systèmes permettant au robot de savoir où il est, avec quoi il interagit, etc. Un robot est particulièrement complexe à mettre au point car il cumule des défis au niveau des capteurs, de la mécanique pour se mouvoir, de la batterie pour son autonomie, et dans l'intelligence artificielle pour piloter l'ensemble et éventuellement interagir à la fois mécaniquement, visuellement et oralement avec des humains.

Le niveau d'abstraction des réseaux d'agents est plus élevé que celui des réseaux de neurones. D'où le fait que j'en termine par là sur cette partie !

Les agents sont notamment utilisés dans les systèmes de call centers. Une start-up française s'était lancée (parmi d'autres) sur ce créneau : **Virtuoz**. Elle a été acquise en 2013 par l'américain **Nuance**. Il existe même un concours du **meilleur agent de service client en ligne**, lancé en 2016 en France avec une trentaine de candidats ! Quid des outils de développement associés ? **Il y en a plein**, et notamment en open source.

Dans la **troisième partie** de cette série, nous étudierons l'étude de cas d'IBM Watson pour évoquer les méthodes de commercialisation et de marketing des solutions d'intelligence artificielle.

Vous pouvez consulter tous les épisodes de ce roman fleuve de printemps sur l'intelligence artificielle :

Episode 1 : **sémantique et questions clés**
Episode 2 : **histoire et technologies de l'intelligence artificielle**
Episode 3 : **IBM Watson et le marketing de l'intelligence artificielle**
Episode 4 : **les startups US de l'intelligence artificielle**
Episode 5 : **les startups acquises par les grands du numérique**
Episode 6 : **les startups françaises de l'intelligence artificielle**
Episode 7 : **la modélisation et la copie du cerveau**
Episode 8 : **évolutions de la loi de Moore et applications à l'intelligence artificielle**
Episode 9 : **la robotisation en marche des métiers**

Cet article a été publié le 13 mars 2016 et édité en PDF le 15 mars 2024.