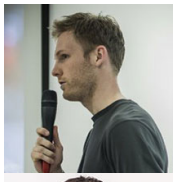
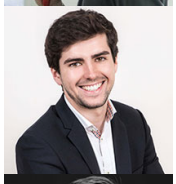
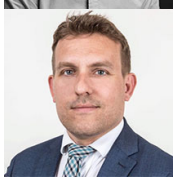
Deep Learning : des usages contrastés
Une contextualisation de l'ouvrage de Goodfellow, Bengio et Courville


Rémi ADON

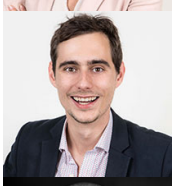

Florian ARTHUR


Guillaume BAQUIAST


Guillaume HOCHARD


Abdellah KAID GHERBI


Aurélia NÈGRE


Antoine SIMOULIN


Fouad TALAOUIT-MOCKLI

Quantmetry


Nicolas BOUSQUET[1]

EDF R&D - Laboratoire d'IA industrielle SINCLAIR & Sorbonne Université

TITLE

A critical analysis of the use of deep learning within the socio-economic world

RÉSUMÉ

Cet article propose une revue critique de la traduction française de l'ouvrage *Deep Learning*, par Ian Goodfellow, Yoshua Bengio et Aaron Courville (Goodfellow et al., 2016 ; MIT Press), publiée sous le titre *L'apprentissage profond* (Éditions Florent Massot, 2018). Celle-ci est devenue célèbre pour avoir été la première traduction scientifique d'envergure coproduite par une intelligence artificielle. Alors que l'apprentissage profond connaît une évolution rapide, cet ouvrage et les idées qu'il véhicule restent profondément d'actualité. Nourrie par de nombreux retours d'expérience portant sur l'usage réel de l'apprentissage profond au sein des entreprises, la mise en perspective de ce corpus de méthodes et d'outils vis-à-vis d'approches plus traditionnelles s'articule autour de trois thématiques-clés : le traitement d'images, l'analyse de séries temporelles et le traitement automatisé du langage naturel. Deux enjeux cruciaux pour une adoption massive mais éclairée y sont également discutés, qui nous semblent contextualiser utilement les apports techniques de l'ouvrage : l'intelligibilité de l'apprentissage profond et l'optimisation énergétique des ressources de calcul.

Mots-clés **:** *apprentissage profond, industrialisation, modèles et algorithmes, méthodes statistiques, analyse critique.*

1. nicolas.bousquet@sorbonne-universite.fr

ABSTRACT

This article provides a critical review of the French translation of the book *Deep Learning*, by Ian Goodfellow, Yoshua Bengio and Aaron Courville (Goodfellow et al., 2016; MIT Press), published under the title *L'apprentissage profond* (Éditions Florent Massot, 2018). This became famous for being the first large-scale scientific translation co-produced by an artificial intelligence. While deep learning is undergoing rapid evolution, this book and the ideas it conveys remain profoundly topical. Based on a large amount of feedback on the real use of deep learning within companies, the perspective of this corpus of methods and tools in relation to more traditional approaches revolves around three key themes: image processing, analysis of temporal data and automated processing of natural language. It also discusses two issues that are crucial for massive but mature adoption, and which seem to us to usefully contextualize the technical contributions of the book: the intelligibility of deep learning-based approaches and the energy optimization of computing resources.

***Keywords*:** *deep learning, industrialization, models and algorithms, statistical methods, critical analysis.*

1. Introduction

1.1. L'essor de l'apprentissage profond

L'ouvrage *Deep Learning*, paru fin 2016, revêt une importance particulière. À notre connaissance, il reste en effet l'un des seuls à offrir, après une introduction aux concepts mathématiques et aux bases de l'apprentissage automatique (*machine learning* ou ML), un panorama vaste et accessible de l'état de l'art scientifique en apprentissage profond (AP), domaine qui connaît aujourd'hui un engouement extraordinaire. La puissance de structuration des algorithmes d'AP, largement fondés sur l'emploi de réseaux de neurones artificiels, suggère de multiples champs applicatifs et les impose aujourd'hui comme les composants phares des intelligences artificielles *connexionnistes* (Schuman et al., 2017b), c'est-à-dire fondées sur l'exhibition de corrélations fines entre les phénomènes produisant les données. Ces outils se révèlent également si adaptés à la gestion des importants volumes de données générés par les activités numériques que des processeurs spécifiquement créés pour l'emploi optimisé de ces algorithmes commencent à être commercialisés à grande échelle (Esser et al., 2016; Schuman et al., 2017a). De ce fait, l'attractivité des métiers liés à l'exposition, la mise en forme et le traitement des données ne cesse de croître.

Quelques années après sa publication originale, cet ouvrage reste fondamental car il répond à l'exigence d'appréhender simultanément le problème de l'apprentissage automatique sous différents angles : ainsi, l'aspect informatique vise à construire des *pipeline* opérationnels de collecte, stockage et traitement des données, tandis que l'aspect statistique ambitionne de proposer un choix de *modèle* explicatif ou génératif sous-jacent à un problème de décision. La traduction française de cet ouvrage a notamment pour objectif de faciliter l'appropriation par les différents domaines professionnels concernés par l'exploitation de données massives ; comme l'écrit Cédric Villani, « *on pense toujours plus facilement [les sciences] dans sa langue natale* » (Villani, 2016).

1.2. Les risques potentiels d'un changement de paradigme

Si cette puissance des algorithmes d'AP couplés à des architectures et des méthodes de prétraitement de données spécialisées est indéniable, le réalisme de la mise en œuvre actuelle et l'utilité pratique de certains d'entre eux, en regard d'autres approches disponibles, sont ici questionnés. Le point de vue développé dans cet article est celui de praticiens expérimentés, confrontés à des problèmes de traitement de données relevant de métiers aussi variés qu'assureur, banquier d'affaire, industriel de l'énergie, professionnel de santé, logisticien, géologue ou chargé de ressources humaines. Pour certains métiers, la popularité de l'AP paraît leur imposer de se renouveler en profondeur, quitte à balayer un vaste ensemble de techniques et de méthodologies historiques. Un exemple frappant d'une telle injonction concerne ainsi le traitement de l'image, pour lequel des méthodes traditionnelles peuvent cependant rester très performantes pour la résolution de problèmes spécifiques, comme la *segmentation**[2] (Féron et Mohammad-Djafari, 2005; Pereyra et McLaughlin, 2015) – celle-ci pouvant justement favoriser l'usage de l'AP, puisqu'elle permet une labellisation[3] plus rapide (Zhang et Xu, 2018). De même, le problème du traitement des données censurées en étude de survie, généralement bien opéré par des algorithmes du type *Expectation-Maximization* (EM), est maintenant appréhendé par des outils d'AP sans que la compréhension du mécanisme probabiliste de

2. Les astériques présents dans le texte renvoie à des définitions proposées dans un lexique en fin d'article.
3. Voir § 3.1.1 pour une explication détaillée de ce terme.

reconstruction de ces données soit explicitée (voir par exemple Katzman et al., 2018). L'impératif permanent de rapidité, répondant à l'attente d'une audience fascinée par des résultats impressionnants en intelligence artificielle, y est sans doute pour quelque chose. Mais un tel dynamisme, illustré par l'augmentation exponentielle du nombre de contributions, risque – par la pression qu'elle place sur les chercheurs, notamment – d'aboutir à des pertes de qualité et de pertinence méthodologique (Bengio, 2020), potentiellement dommageables pour l'usage réel de l'AP et son enseignement (Lipton et Steinhardt, 2019).

De façon sous-jacente, nous percevons au contact des applications métiers que la très forte publicité donnée aux outils d'AP risque d'isoler des communautés scientifiques et de se priver de solutions utiles, éprouvées, moins gourmandes en données et en ressources. Notre conviction est qu'il est naturel, et souhaitable, que l'avancement de la science offre un nombre croissant de solutions à un problème donné, mais qu'il est capital de perpétuer la connaissance et l'usage d'approches traditionnelles. Celles-ci peuvent conserver un avantage comparatif par rapport aux dernières méthodologies proposées, ne serait-ce qu'en facilité d'interprétation[4]. Le développement et l'usage intensif de ces algorithmes s'accompagne de questionnements croissants sur leur intelligibilité, leur transparence, leur loyauté* et plus généralement sur la nature du produit informationnel résultant du traitement des données (Besse et al., 2017). Des modèles explicatifs, causaux, exhibant les propriétés principales d'un phénomène permettent un tel dialogue entre l'homme et la machine « boîte noire » (Bhatt, 2018).

Nous pensons donc que l'AP mérite encore de nombreux approfondissements théoriques ainsi que des retours d'expériences toujours plus variés, et qu'il est nécessaire de contextualiser son usage dans les applications métiers. L'objectif de cet article est de contribuer à ces réflexions.

1.3. Contributions de l'article

Un résumé des principaux contenus et messages de l'ouvrage suit cette introduction. Nous considérons ensuite trois champs d'application importants de l'apprentissage profond – le traitement d'image, celui de signaux temporels et de données textuelles – illustrés par des situations « de terrain ». Ces exemples permettent de percevoir les gains de performance, mais aussi les contraintes qu'amènent les méthodologies fondées sur les réseaux de neurones profonds, et les solutions traditionnelles utilisables en première intention. Les choix de solution répondent à des considérations pratiques multiples, et ne sont pas seulement liés à la disponibilité de machines puissantes ou à la taille des données disponibles. Une section de discussion est enfin consacrée à deux sujets d'ouverture peu développés dans l'ouvrage, au sujet desquels entreprises et société civile prêtent de plus en plus d'attention : l'intelligibilité des algorithmes d'AP et le coût environnemental de leur mise en œuvre au sein d'outils d'intelligence artificielle.

2. Résumé de l'ouvrage

Le monde de l'apprentissage automatique profond est abordé en trois parties. La première est consacrée aux bases d'un cursus de mathématiques appliquées et aux fondations statistiques de l'apprentissage. Des outils classiques (maximisation de vraisemblance, règle de Bayes, ...) jusqu'aux concepts les plus avancés (astuce du noyau, modèle de représentation), l'essentiel

4. Il est d'ailleurs assez révélateur de percevoir que les approches *post hoc* d'interprétation des modèles et algorithmes d'AP reposent essentiellement sur des métriques de comparaison avec des approches statistiques classiques ; voir § 4.2.

des idées de modélisation et d'estimation définissant les approches supervisées, non supervisées et par renforcement de l'apprentissage statistique est présenté selon une approche historique.

La notion de supervision renvoie à l'existence de données $(\mathbb{x}, \mathbb{y}) = (x_i, y_i)_{i\in\mathcal{I}}$, où $\mathcal{I}$ désigne un échantillon dit d'*apprentissage*. Si $\mathbb{x}$ décrit des caractéristiques observables du phénomène à l'étude, le *label*[5] $\mathbb{y}$ décrit une caractéristique décisionnelle vis-à-vis de ce phénomène, idéalement reliée à $\mathbb{x}$ par des relations causales inconnues : quelle action y doit-on préconiser, voire automatiser si on observe x, sachant qu'on connaît $(\mathbb{x}, \mathbb{y})$? Ce label $\mathbb{y}$ peut, typiquement, décrire une classe de phénomènes (approche par classification, voir Figure 1a) ou l'étalement d'une valeur ou d'un vecteur de valeurs d'intérêt (approche par régression). Une démarche *non supervisée*, privée de $\mathbb{y}$, exploite la géométrie du nuage de points $\mathbb{x}$ pour en déduire des modèles statistiques sur $\mathbb{x}$ (ex. : mélanges gaussiens) ou des partitionnements de ce nuage de points. Elle sous-tend donc qu'une décision $\mathbb{y}$ peut être formalisée à partir de cette classification (Figure 1b). Le *renforcement*, quant à lui, consiste à permettre une exploration progressive de l'environnement du couple $(\mathbb{x}, \mathbb{y})$ et s'approche de la réalité d'une prise de décision à effectuer séquentiellement dans un contexte où les incertitudes sont graduellement réduites (Figure 1c).

Ces trois approches délimitent les limites conceptuelles de ce qu'on nomme l'apprentissage automatique (on « apprend » en une fois ou progressivement les corrélations liant $\mathbb{x}$ à $\mathbb{y}$). Celui-ci comprend l'apprentissage machine (AM, ou *machine learning*) traditionnel et l'AP.

Dans la réalité des applications pratiques, les deux premières approches sont souvent mises en interaction[6]. L'apprentissage par renforcement présente cette particularité d'être encore réservé à des cas d'étude très délimités[7], en dépit de l'intérêt qu'il suscite. Sa faible maturité implique qu'il reste nettement moins étudié dans *L'apprentissage profond*, et le lecteur intéressé pourra plutôt se référer à Szepesvári (2010), l'une des bibles du domaine.

Transverses au domaine de l'apprentissage automatique, les enjeux cruciaux de régularisation, généralisation et de montée en échelle (ou *scalabilité*) – c'est-à-dire la limitation des risques de sur-apprentissage*, la pertinence de l'application des outils à de nouvelles données $\tilde{\mathbb{x}}$ et la gestion du fléau de la dimension* et de la taille croissante des données – permettent de saisir l'intérêt des approches par réseaux de neurones artificiels (RNN) dits *profonds*, qui font l'objet de la seconde partie du livre et qui constituent les outils fondamentaux de l'AP.

Les RNN constituent une classe de modèles de traitement du signal, qui s'inspirent vaguement du fonctionnement des neurones biologiques. Ils sont composés d'unités inter-connectées disposées en couches successives, décrites comme des neurones artificiels (Figure 2). Dans leurs formes les plus simples (dite de *propagation avant*), par le biais de techniques d'optimisation formelles[8] dites d'*entraînement* (un statisticien dirait d'*estimation*) , ces structures peuvent réaliser des approximations de correspondances *a priori* inconnues entre un ensemble d'entrées observées $\mathbb{x}$ et des sorties (labels) $\mathbb{y}$ connues. Étant donné une entrée $\mathbb{x} = (x_1, \dots, x_n)$, un neurone génère un signal de sortie

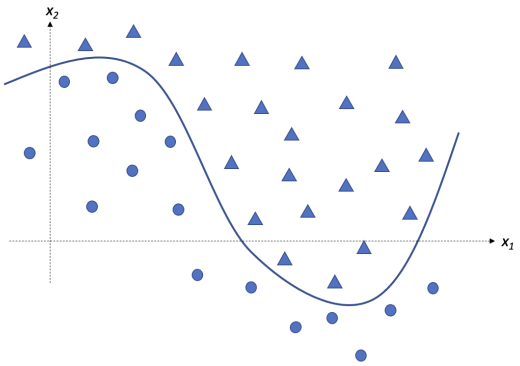
$$y = \sigma\left(\sum_{i=1}^{n} w_i x_i - b\right), \tag{1}$$

5. On parle aussi d'*étiquette* pour désigner $\mathbb{y}$.

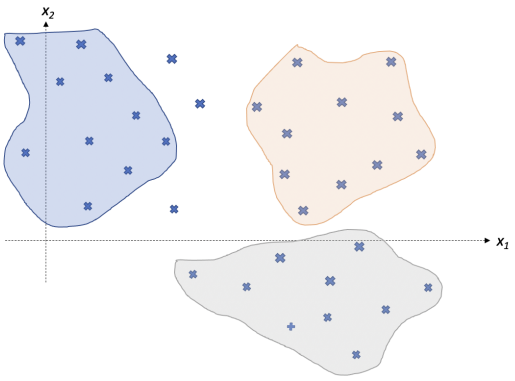
6. Ainsi, l'apprentissage *semi-supervisé*, d'usage courant, est un apprentissage qui dispose de labels pour une partie seulement des données d'entrée.

7. En particulier, on doit pouvoir opérer des simplifications fortes sur l'espace des configurations de l'environnement à explorer (ex. : son cardinal est fini et reste faible) afin d'éviter une explosion combinatoire, et/ou bénéficier d'une puissance de calcul exceptionnelle ; c'est en effet le cas des jeux de stratégie pour lequel ce type d'apprentissage a été mis en valeur.

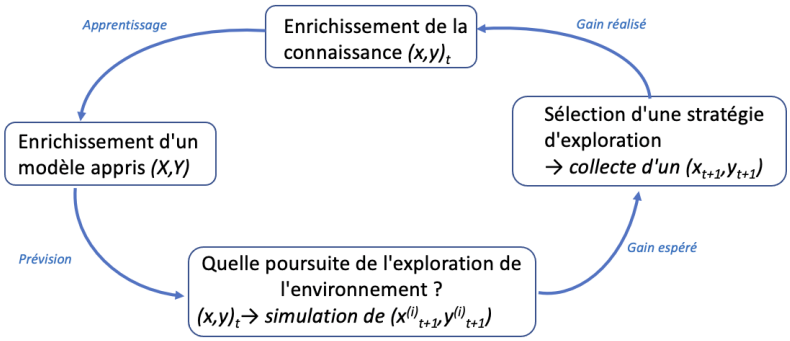
8. Fondées sur la connaissance explicite des gradients du modèle vis-à-vis de ses paramètres, et *rétropropageant* l'erreur entre prévision du modèle et observation des labels $\mathbb{y}$.



(a) *$\mathbb{y}$ est une famille binaire de formes (rond, triangle), le modèle à apprendre à partir d'un échantillon $(\mathbb{x}, \mathbb{y})$ connu est la surface de classification (en ligne pleine).*



(b) *Des regroupements de points géométriquement proches (clustering) sont opérés, qui pourront être étudiés par modélisation statistique. Certains peuvent être difficiles à classifier.*



(c) *L'apprentissage par renforcement est une approche itérative (markovienne) de la construction du lien entre une information $\mathbb{x}$ et un label $\mathbb{y}$, qui se fonde sur une fonction de décision (gain apporté par la connaissance des $\mathbb{y}$, tel que la diminution de l'erreur de modélisation $\mathbb{x} \to \mathbb{y}$). Son optimisation requiert la collecte d'un couple (x, y) le plus informatif possible, à chaque étape de l'approche, et se fonde sur le modèle courant de représentation de l'environnement $(\mathbb{x}, \mathbb{y})$. Dans ce schéma, t est une variable d'itération temporelle.*

FIGURE 1 – *Illustrations d'approches supervisée (a) et non supervisée (b) en classification, pour $\mathbb{x}$ de dimension 2. Illustration du principe d'apprentissage par renforcement (c).*

où la *fonction d'activation* σ est une transformation non linéaire, w_i est un poids associé à l'influence de la donnée x_i et b un biais. La succession de couches, chacune possédant plusieurs neurones traitant parallèlement le signal d'entrée, permet d'offrir des compositions de fonctions sur des partitions de ce signal. Cette grande flexibilité apportée par des fonctions d'activation simples permet théoriquement aux RNN de reproduire tout comportement reliant continûment $\mathbb{x}$ et $\mathbb{y}$ (Cybenko, 1989). En réalité, le choix multi-couches permet d'ôter de nombreuses hypothèses sur le choix de σ pour conserver cette propriété d'approximation universelle (Hornik, 1991; Hanin, 2019; Kidger et Lyons, 2020), cependant sans fournir d'indication forte sur l'architecture (tel un nombre optimisé de neurones par couche).

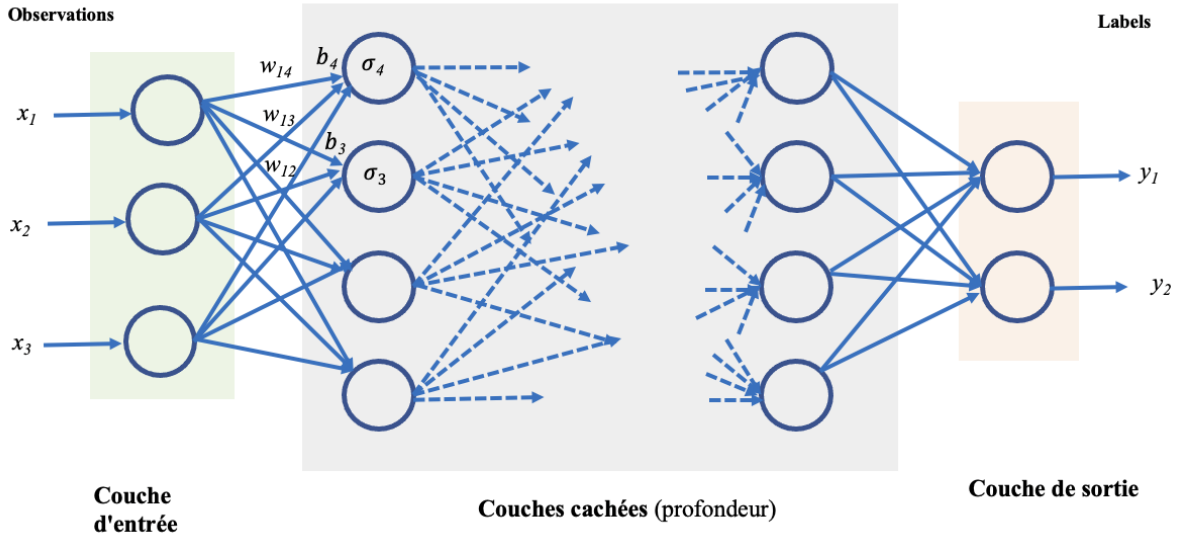


FIGURE 2 – *Illustration d'un réseau de neurones artificiels (RNN). Un RNN est défini par des couches successives, des biais et des fonctions d'activation, transformant un signal multivarié en une sortie (éventuellement) multivariée.* σ *est la fonction d'activation, apportant la non-linéarité à l'équation (1),* ω_{ij} *est le poids à l'observation* x_i *et au neurone* ij*, et* b *est un paramètre de biais.*

Les RNN dont le nombre de couches cachées (profondeur) est de 1 peuvent représenter aisément un grand nombre de transformations de $\mathbb{x}$ vers $\mathbb{y}$, et constituent un prolongement naturel de transformations non linéaires usuelles en statistique, comme la fonction logistique. Toutefois, il est nécessaire d'augmenter cette profondeur (et l'apprentissage est alors dit *profond*) pour apprendre les relations de corrélation (*motifs* ou *patterns*) les plus fins reliant signal d'entrée et signal de sortie, et améliorer la prévision y lorsqu'une nouvelle donnée x se présente. On augmente ce faisant le nombre de paramètres (ω, b) à estimer et celui des *hyperparamètres*[9] à optimiser.

Plus généralement, l'AP se différencie de l'apprentissage machine (AM) classique par le recours à des outils (modèles, algorithmes) construits spécifiquement pour rechercher des corrélations impossibles à déterminer *pratiquement* par des outils de l'AM ; ces outils mettent en jeu plus de deux couches cachées dans des réseaux de neurones, des paramètres et hyperparamètres beaucoup plus nombreux, et requièrent donc des données massives. L'AP se démarque également de l'AM par sa démarche de conception logicielle : il s'agit d'assembler des réseaux de blocs fonctionnels paramétrés en les calibrant à partir de données *via* une certaine forme d'optimisation fondée sur des gradients explicitement connus[10] : c'est une forme de *programmation différentiable* (Innes et al., 2018), à laquelle des langages de programmation spécifiques sont dédiés.

L'objectif des praticiens de l'AP est alors d'établir un compromis entre architecture de modèle et méthode d'optimisation afin de capturer un grand nombre de motifs présents dans les données, faiblement ou non directement accessibles au sens commun : texture et propriétés géométriques des images, structuration sous-jacente d'un texte, etc. De nombreuses méthodes de régularisation permettent de limiter le sur-apprentissage, telle l'augmentation du jeu de données en traitement d'image. Le lecteur trouvera donc dans cette seconde partie de

9. C'est-à-dire les paramètres de réglage des algorithmes d'entraînement.
10. D'après Yann Le Cun, Facebook, 5 janvier 2018.

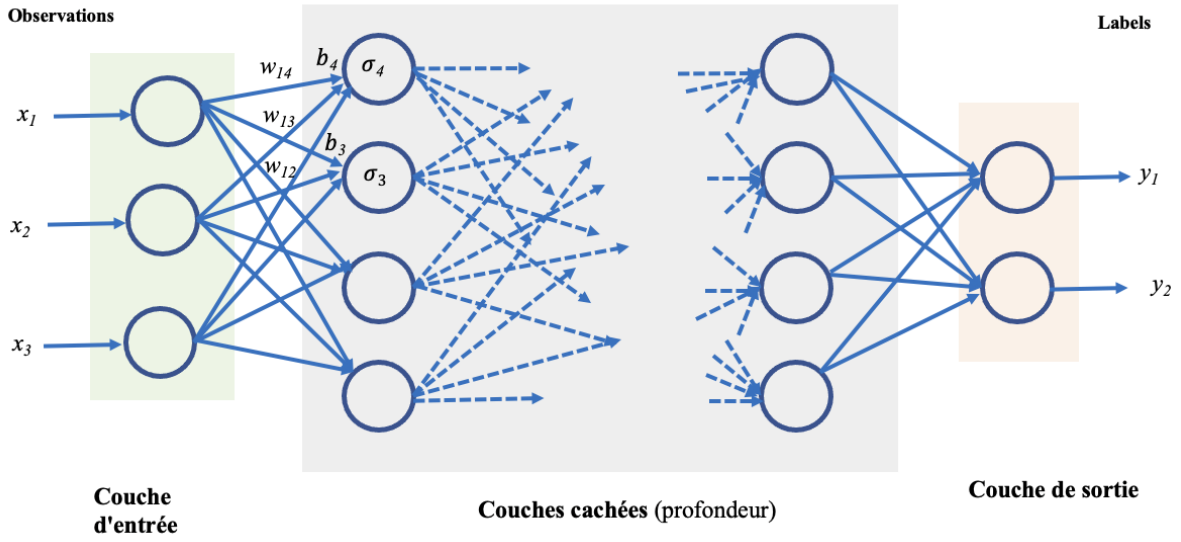


FIGURE 2 – *Illustration d'un réseau de neurones artificiels (RNN). Un RNN est défini par des couches successives, des biais et des fonctions d'activation, transformant un signal multivarié en une sortie (éventuellement) multivariée. σ est la fonction d'activation, apportant la non-linéarité à l'équation (1), ω_{ij} est le poids à l'observation x_i et au neurone ij, et b est un paramètre de biais.*

Les RNN dont le nombre de couches cachées (profondeur) est de 1 peuvent représenter aisément un grand nombre de transformations de $\mathbb{x}$ vers $\mathbb{y}$, et constituent un prolongement naturel de transformations non linéaires usuelles en statistique, comme la fonction logistique. Toutefois, il est nécessaire d'augmenter cette profondeur (et l'apprentissage est alors dit *profond*) pour apprendre les relations de corrélation (*motifs* ou *patterns*) les plus fins reliant signal d'entrée et signal de sortie, et améliorer la prévision y lorsqu'une nouvelle donnée x se présente. On augmente ce faisant le nombre de paramètres (ω, b) à estimer et celui des *hyperparamètres*[9] à optimiser.

Plus généralement, l'AP se différencie de l'apprentissage machine (AM) classique par le recours à des outils (modèles, algorithmes) construits spécifiquement pour rechercher des corrélations impossibles à déterminer *pratiquement* par des outils de l'AM ; ces outils mettent en jeu plus de deux couches cachées dans des réseaux de neurones, des paramètres et hyperparamètres beaucoup plus nombreux, et requièrent donc des données massives. L'AP se démarque également de l'AM par sa démarche de conception logicielle : il s'agit d'assembler des réseaux de blocs fonctionnels paramétrés en les calibrant à partir de données *via* une certaine forme d'optimisation fondée sur des gradients explicitement connus[10] : c'est une forme de *programmation différentiable* (Innes et al., 2018), à laquelle des langages de programmation spécifiques sont dédiés.

L'objectif des praticiens de l'AP est alors d'établir un compromis entre architecture de modèle et méthode d'optimisation afin de capturer un grand nombre de motifs présents dans les données, faiblement ou non directement accessibles au sens commun : texture et propriétés géométriques des images, structuration sous-jacente d'un texte, etc. De nombreuses méthodes de régularisation permettent de limiter le sur-apprentissage, telle l'augmentation du jeu de données en traitement d'image. Le lecteur trouvera donc dans cette seconde partie de

9. C'est-à-dire les paramètres de réglage des algorithmes d'entraînement.
10. D'après Yann Le Cun, Facebook, 5 janvier 2018.

l'ouvrage tous les concepts nécessaires à la conception, à la configuration et à l'optimisation d'un modèle de réseaux de neurones profonds.

La troisième et dernière partie de *L'apprentissage profond* présente enfin des pistes de recherches actives au moment de l'écriture de l'ouvrage – et qui le restent encore aujourd'hui. Différents modèles de réseaux, comme les auto-encodeurs ou les réseaux génératifs par antagonisme (GAN), qui agissent en binôme pour produire des données synthétiques difficilement discernables des données réelles, y sont présentés. Différentes méthodologies y sont détaillées, qui visent à étendre les domaines d'utilisation de l'apprentissage profond, notamment à de tels problèmes de génération de données et plus généralement d'apprentissage non supervisé. Les exemples évoqués sont nombreux, qui vont de la production de musique à l'augmentation de la résolution d'une image.

L'ouvrage est pensé pour un vaste public et abondamment illustré. Surtout, la présentation des principaux concepts se place en permanence dans un contexte *applicatif* : on ne peut parler des objets mathématiques *vecteur*, *matrice*, *tenseur* ou *opération de convolution* sans décrire la réalité physique de leur représentation dans l'espace mémoire ou la mémoire dynamique d'une machine. Différant formellement de l'approche théorique privilégiée par les statisticiens (telle qu'on peut la trouver dans l'ouvrage-phare de Hastie et al. (2001)), elle ne s'exonère jamais des contraintes posées par l'incarnation du calcul. L'exemple de la fonction d'activation $K-$dimensionnelle *softmax*[11]

$$\sigma(\mathbb{z})_j = \frac{\exp(z_j)}{\sum_{k=1}^{K} \exp(z_k)}$$

est révélateur : les auteurs rappellent au § 6.2.2 qu'elle possède autant de raisons pratiques que théoriques d'être utilisée pour approcher le comportement d'un vecteur de probabilité, d'où son usage intensif en classification. L'apprentissage se situe d'ailleurs au croisement de l'informatique, de la statistique, de l'optimisation, de l'analyse numérique et de la géométrie ; dans l'esprit des auteurs, il apparaît donc vain de privilégier un formalisme plutôt qu'un autre, et plus simple et naturel d'aborder le sujet sous l'angle de la mise en œuvre concrète. Ainsi, si pour un statisticien ou un expert en traitement du signal il n'existe pas de différence formelle entre un modèle d'AM et un modèle d'AP, comme indiqué précédemment, le second diffère du premier en ce sens qu'il tente non seulement d'élaborer plusieurs niveaux de représentations de l'information (produire un *vocabulaire*), mais aussi de les articuler entre elles, au moyen d'un nombre colossal de paramètres et par l'usage de briques logicielles différentiables.

Cet ouvrage permet enfin de saisir avec simplicité les difficultés actuelles de compréhension des mécanismes de l'AP – et au-delà, de l'IA dite *connexionniste*, tirant parti de motifs détectés dans les données. Il faut pour cela revenir aux fondations de l'apprentissage statistique.

Fondamentalement, celui-ci est bâti sur la définition puis la minimisation en θ d'une *fonction de coût* $L(\mathbb{x}, \mathbb{y}; \theta)$ entre des données représentant un phénomène d'intérêt Σ et un modèle $\mathcal{M}$ de ces données, paramétré par θ. Le vecteur de paramètres guide la forme, les propriétés essentielles, l'architecture, etc. de $\mathcal{M}$. Si l'on connaissait exhaustivement Σ (par exemple toutes les variétés de données-clients d'un assureur), on pourrait imaginer pouvoir prendre une décision optimale pour chaque nouvelle situation x se présentant (ex. : décider quel contrat d'assurance convient le mieux à tel client). Mais cette connaissance exhaustive étant inatteignable, le concepteur cherche à prendre la moins mauvaise décision possible conditionnellement à la connaissance des données disponibles. La fonction $L(\mathbb{x}, \mathbb{y}; \theta)$ résume donc, pour ce concepteur, les conséquences décisionnelles d'une erreur d'apprentissage liée au choix de $\mathcal{M}(\theta)$, où

11. où $\mathbb{z} = (z_1, \ldots, z_K)$ est typiquement un signal de sortie de l'avant-dernière couche d'un RNN utilisé pour une tâche de classification.

θ est estimé à partir des données disponibles.

Cette fonction de coût incorpore des métriques probabilistes, les données étant considérées comme des représentations de variables aléatoires. Plus généralement, cette fonction de coût dépend de l'objectif d'apprentissage et de la nature des données. L'apprentissage repose donc sur un choix de représentation (modèle), un choix de fonction de coût et une procédure d'optimisation. Nos choix de fonction de coût sont limités par notre capacité à comprendre l'espace des représentations possibles des données, et nous ne pouvons guère avoir de garantie sur l'exhaustivité de notre interprétation. Par ailleurs, les procédures d'optimisation sont définies de façon à combattre le manque de *(quasi-)convexité* globale en θ des fonctions de coût (Greenberg et Pierskalla, 1971). Or cette (quasi-)convexité est nécessaire pour obtenir un *optimum* global, et donc la garantie d'un apprentissage meilleur que tout autre fondé sur la même fonction de coût et les mêmes données (Figure 3).

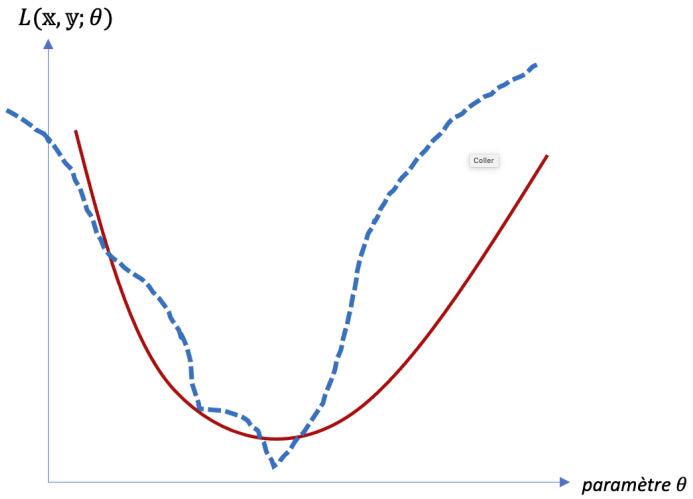


FIGURE 3 – *Pour un vecteur de paramètres θ réduit à la dimension 1, deux exemples de comportement souhaité de fonctions de coût : la recherche d'un minimum global en θ de ces fonctions est pertinent et réalisable par des algorithmes spécialisés si l'on sait ou l'on apprend qu'elles sont convexes (courbe pleine) ou quasi-convexes (courbe pointillée), cette dernière propriété étant bien plus générale que la convexité. Dans les deux cas, l'existence d'un minimum global (et donc d'une « meilleure » représentation possible étant donné le choix de modèle) est assurée.*

Dans le cas de l'AP, les algorithmes les plus avancés permettent d'obtenir des *optima* locaux, qui se révèlent proches des globaux sur de nombreux exemples bien connus, mais il est encore difficile d'en comprendre la raison ; la forme des fonctions de coût est en effet, en grande dimension, délicate à appréhender. Rajoutons également que nous n'avons encore guère, par ailleurs, de garanties théoriques sur le comportement exploratoire des algorithmes et leur atteinte réelle de ces optima, d'autant plus que nous savons que la plupart des réseaux de neurones ne peuvent être estimés d'une façon unique (ils manquent d'identifiabilité de par leur construction).

Dans son récent ouvrage de témoignage et de vulgarisation, Yann Le Cun (Le Cun, 2019, chap. 4) défend cependant l'idée que d'un point de vue pratique, le problème n'en est guère un. Un bon comportement largement prouvé de façon empirique suffirait en général à légitimer une IA fondée sur de l'AP. Pourtant, on peut présumer que pour des applications de l'IA en gestion de systèmes critiques*, l'une des ambitions majeures des programmes de re-

cherche internationaux actuels[12], les étapes de certification s'appuient aussi sur des garanties théoriques plus robustes.

Remercions donc les auteurs pour exhiber ainsi les limitations théoriques actuelles de ces méthodes *implémentées* ; ils font œuvre de salubrité publique en plaçant des mots clairs sur des incertitudes qui imprègnent encore la puissance de ces outils, et motivent ainsi les développements actuels de la recherche sur les propriétés de généralisation des IA connexionnistes. Quelques années après sa publication originale, en dépit de l'effervescence du domaine scientifique qu'il cherche à couvrir, cet ouvrage reste profondément d'actualité.

3. Analyse critique de trois types de cas d'études

Les principales applications industrielles de l'apprentissage profond portent actuellement sur l'analyse d'images, l'exploration de séries temporelles – et en particulier les séries de données produites par des capteurs – ainsi que le traitement automatisé du langage naturel. Dans cette section, ces cas d'usage sont analysés en considérant deux cas de figure distincts.

- Dans le cadre d'une **preuve de concept**, une forte contrainte de temps couplée à une contrainte sur la puissance du matériel à disposition s'exercent sur les ingénieurs, de façon récurrente, afin de produire des résultats. Le temps nécessaire à l'entraînement des réseaux de neurones profonds peut se révéler particulièrement problématique, et les arbitrages en faveur de modèles moins gourmands en ressources sont fréquents.

- Dans le cadre du **développement d'une solution logicielle** possédant une brique analytique, la réalisation est itérative ; le développement d'une application fonctionnelle et intuitive pour les utilisateurs est d'abord privilégié, sans forcément inclure des algorithmes complexes. Une seconde étape consiste en l'ajout de fontionnalités analytiques simples, permettant de fournir une première version de noyau d'intelligence. L'ajout d'une brique d'apprentissage profond est enfin étudié en fonction du cas d'usage et de la nécessité de raffiner les fonctionnalités des modèles.

En Annexe B, des typologies de cas d'usage, issus de la littérature ou que nous avons étudiés ces dernières années, sont détaillées afin de mieux illustrer les bénéfices et limitations des outils décrits dans les sous-sections suivantes. Par ailleurs, ces dernières comprennent des paragraphes techniques, comprenant de nombreuses références utiles et qui peuvent être sautés en première lecture, et des paragraphes plus généraux, incorporant conclusions et recommandations pratiques.

3.1. Détection et classification de situations dans des images

3.1.1. Cas d'étude, outils et apports de l'apprentissage profond

L'analyse d'images est certainement le domaine d'ingénierie qui a connu, grâce à l'apprentissage profond, la progression la plus spectaculaire depuis les années 1990 et les premières lectures automatiques de chèque (Jayadevan et al., 2011). Elle est très majoritairement fondée sur une classe particulière de RNN, les *réseaux de neurones convolutifs* (RNC, ou CNN en anglais), qui s'inspirent du comportement de captation de la structure d'une image par l'œil

12. Tels les projets *Explainable AI* (https://www.darpa.mil/program/explainable-artificial-intelligence), DEEL (www.deel.ai) ou *Partnership on AI* (https://www.partnershiponai.org).

en réduisant le signal d'entrée par une opération nommée convolution ; voir Le Cun et Bengio (1995) pour plus de détail, et la Figure 4.

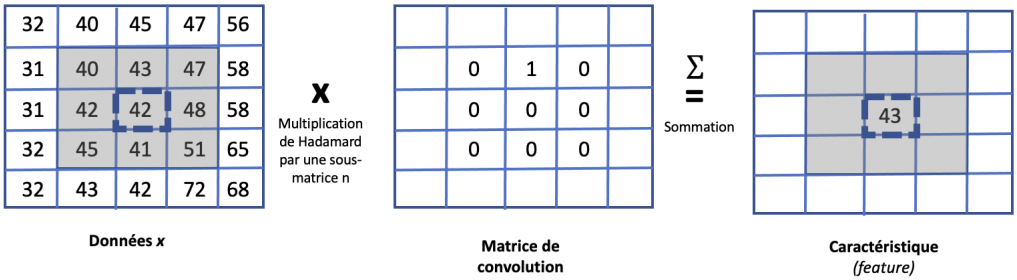


FIGURE 4 – *RNC prend en entrée une image x incarnée par une matrice de pixels auxquels sont associées des valeurs numériques entières codant des couleurs. Il opère un filtrage de sous-matrices de x (de dimension 3 x 3 dans cet exemple) en les multipliant (produit de Hadamard) par une matrice de poids ω (ici tous nuls sauf 1), puis en opérant une opération de convolution, qui revient à sommer sur ces poids. Le résultat de ce filtrage est l'activation d'un neurone portant une information résumant la variation des valeurs des pixels de la sous-matrice (dite* méga-pixel*) ; cette information résumée est appelée* caractéristique *ou* feature*.*

Les industries agroalimentaire (Liu et al., 2018), aéronautique (Kim et al., 2016), automobile (Li et al., 2018) et militaire (Chen et Wang, 2014), la santé (Litjens et al., 2017) et la sécurité (Akçay et al., 2016), le secteur bancaire, les administrations publiques (Anastasopoulos et Whitford, 2018) et la gestion environnementale en font un usage de plus en plus intensif. La *labellisation*, c'est-à-dire l'étiquetage des caractéristiques d'une image en vue de la classification d'un exemple type, puis la *classification* des images elles-mêmes en constituent les traitements les plus courants. Elles sont aussi bien utilisées pour tester la présence de défauts de production sur des produits en fin de chaîne industrielle (Wang et al., 2018) que les variations du couvert végétal, synonymes de déforestation sauvage (Zhao et al., 2017; Rakshit et al., 2018). La mise à disposition de banques d'images labellisées, telles qu'`ImageNet`, a permis de développer considérablement la classification supervisée. Toutefois, la grande variété de ces images limite la précision du pouvoir de classifieurs ainsi construits lorsqu'ils sont appliqués sur des images fortement contextualisées, et l'accroissement de la base de données s'avère généralement indispensable pour atteindre des performances proches de celles de l'humain. L'amélioration des techniques de reconnaissance optique de caractères (OCR) a ainsi fortement bénéficié des tests de sécurité CAPTCHA exécutés par les utilisateurs d'Internet depuis les années 2000, dont le résultat permet de labelliser des images de symboles (von Ahn et al., 2008).

L'AP appliqué aux images consiste surtout à localiser (segmenter) et analyser des objets au sein de ces images, généralement par le biais de RNC. L'utilisation des banques d'images permet de proposer aux utilisateurs des réseaux pré-entraînés[13], qu'il convient de réajuster (ou réentraîner) sur une banque d'images spécifiques au problème considéré. Certaines couches de ces réseaux permettent de capturer des particularités de l'image, telles un style de peinture sur la photo d'un tableau ; associé à des étapes de positionnement des objets, le réentraînement peut ainsi permettre de produire des algorithmes capables de reconnaître la proximité de style entre deux tableaux (Lesaffre, 2018). Au fur et à mesure du temps, les problèmes d'ingestion et de conservation des images et du traitement du temps réel ont conduit les chercheurs à

13. C'est-à-dire des R(C)NN qui ont déjà fait l'objet d'un premier entraînement sur des données présentant des ressemblances structurelles avec celles qui font l'objet de l'étude, et dont certains paramètres / hyperparamètres vont être conservés ; le second entraînement ne concerne donc qu'une partie des paramètres, ce qui permet d'utiliser des échantillons labellisés plus petits. Cette tâche de ré-entraînement est un exemple d'apprentissage par transfert*.

proposer des approches d'exploration rapides, telles Faster R-CNN (Ren et al., 2015) ou YOLO (Redmon et Farhadi, 2017), ainsi que la construction de banques de *masques*, permettant d'appliquer à des fractions d'image des algorithmes de reconnaissance et de séparation de motifs possiblement superposés (He et al., 2017). *Via* l'utilisation d'un dictionnaire de formes, il est par exemple devenu possible de détecter et dimensionner des toitures de bâtiments à partir de prises de vues aériennes, afin de calculer la surface potentiellement utilisable pour l'installation de panneaux solaires, ou encore de reconnaître des cellules d'un type particulier dans des amas cellulaires à partir d'images médicales (Ghouzam et Valverde, 2018).

En détectant et reproduisant les liens complexes entre les pixels d'une image à l'aide d'une hiérarchie de concepts, allant du pixel individuel aux motifs et aux formes, mais aussi en permettant de traiter des types d'image variés (2D, 3D (Maturana et Scherer, 2015), niveaux de gris, infrarouge (Gundogdu et al., 2016), etc.), les réseaux de neurones profonds ont ainsi permis une progression fulgurante de la performance des modèles d'analyse d'images depuis une vingtaine d'années. Depuis 2012, les vainqueurs des plus grandes compétitions mondiales de reconnaissance d'objet, telles que l'`ImageNet Large Scale Visual Recognition Challenge` (ILSVRC), utilisent systématiquement des réseaux de neurones profonds.

Les dernières années ont vu émerger enfin des possibilités d'applications très spécifiques, qui restent encore, pour la majeure partie d'entre elles, au stade de prototype avancé et relativement peu généralisable : compression d'image (Valenzise et al., 2018), cryptage (Dowlin et al., 2016), extraction et transfert de style (Gatys et al., 2016), colorisation (Zhang et al., 2016) et production de nouvelles images *via* des réseaux profonds dits *génératifs* faisant usage de mécanismes antagonistes (GAN) (Huang et al., 2018). Ces derniers, à l'origine des fameux *deepfakes*, sont aujourd'hui surtout considérés comme des aides prometteuses à la conception (par exemple de biens de consommation (Deverall et al., 2017), d'architectures (As et al., 2018) ou de composants industriels (Oh et al., 2019)), et des outils potentiellement intéressants pour produire des données anonymisées (Huang et al., 2018) – deux domaines en plein essor économique.

3.1.2. Limitations et approches alternatives

L'utilisation massive des RNC adaptés au traitement d'images est limitée en pratique par la disponibilité d'outils pré-entraînés, issus de grands laboratoires publics ou privés, et de banques d'images spécifiques. Ainsi, l'art des praticiens est de sélectionner des architectures de réseau par rapprochement avec des données et des contraintes opérationnelles connues, de conserver certains paramètres-clés issus des premiers entraînements (apprentissage par transfert*) et au contraire d'en réestimer d'autres.

Dès lors qu'il s'agit de travailler sur des données fortement structurées (ex. : IRM du cerveau, imagerie satellite à haute résolution) et/ou à caractère personnel (ex. : photographies d'identité, imagerie de santé) ou encore d'usage restreint par le secret industriel (ex. : images de drone survolant des installations), il devient difficile de se procurer en un temps raisonnable des jeux labellisés de grande taille. Si la recherche en labellisation rapide des éléments présents dans les images (*annotation*) – et non plus la simple classification de l'image – connaît un engouement important (He et al., 2018), leur disponibilité reste le problème principal des industriels et des institutions. Les limitations des jeux de données peuvent en effet aboutir à des biais importants, voire dangereux : des chercheurs ont récemment montré que des algorithmes de reconnaissance faciale par AP proposés par Microsoft ou IBM ont été entraînés avec des images de diversité trop faible et aboutissaient à des biais de genre et de couleur

de peau (Buolamwini et Gebru, 2018)[14]. Google a récemment rencontré d'autres difficultés en imagerie médicale, du fait d'un écart entre données d'entraînement d'excellente qualité et données de routine clinique (Heaven, 2020).

Dans un cadre semi-supervisé (voir § 2) ou non supervisé, actuellement plus réaliste pour ce type de données, on ne peut oublier des méthodes statistiques plus anciennes, visant à segmenter les images par le biais d'une modélisation par mélanges de l'information « cachée » (c'est-à-dire de la structure des objets que l'on recherche) (Aas et al., 2007). Un exemple typique d'information cachée est la finesse de séparation des objets au niveau pixel : est-elle nette, ou au contraire doit-on faire l'hypothèse d'un voisinage diffus ? Typiquement fondés sur une hypothèse de Markov et estimés à partir de techniques d'augmentation de données (Celeux et al., 2003), ces modèles statistiques ont notamment été appliqués avec succès à de nombreux problèmes en imagerie de santé (Féron et Mohammad-Djafari, 2005). Par ailleurs, des approches statistiques supervisées telles que les méthodes (ou machines) à noyaux graphiques (Harchaoui et Bach, 2007), qui souffrent certes du fléau de la dimension*, se révèlent cependant précieuses sur des images de faible dimension et requièrent moins d'images pour l'estimation ; elles mériteraient ainsi d'être employées en coopération avec des outils d'apprentissage profond pour approfondir la qualité d'une représentation particulière. Leur industrialisation au travers de langages informatiques de haut niveau (Python, Scala, etc.) reste cependant à étendre en regard des enjeux liés à l'automatisation croissante du traitement de l'information et d'accélération de diagnostic.

Préalablement au travail de segmentation, de nombreuses méthodes traditionnelles (voir Shapiro et Stockman (2003) pour une revue), aisément utilisables, permettent de produire des caractéristiques (*featuring*) à partir desquelles des modèles simples peuvent répondre à de nombreux besoins des industriels.

- L'*histogramme des couleurs* apporte en général assez d'information pour construire rapidement des modèles performants détectant des images obstruées par des nuages, ou la présence de tout autre élément ayant des signatures de couleurs spécifiques au sein des images.

- La *cascade de Haar* (Viola et Jones, 2001) permet par exemple de détecter des personnes et des visages en se fondant sur des caractéristiques construites en calculant des différences entre les sommes des valeurs des pixels de plusieurs zones. Faciles et rapides à calculer, elles peuvent être utilisées pour des cas d'usage de détection sur des flux vidéos, même sous contrainte de temps réel.

- L'*histogramme de gradient orienté* (Dalal et Triggs, 2005) génère des caractéristiques permettant aussi d'entraîner des modèles aux résultats intéressants sur la détection de formes. Dans ce cas, on construit des histogrammes d'orientation des gradients au sein de fenêtres de quelques dizaines de pixels.

Ces techniques ont été utilisées pour résoudre plusieurs cas concrets, portant notamment sur la labellisation multi-classes d'images satellites. Les performances d'un réseau pré-entraîné avec réentraînement des couches supérieures et celles de forêts aléatoires calibrées sur l'histogramme des couleurs ont été comparées. Ce dernier modèle a permis d'obtenir de bons résultats[15] très rapidement, mais de façon hétérogène selon le label, en fonction de la capacité du modèle à trouver une signature de couleurs propre au label. L'utilisation du réseau

14. Ajoutant à cela les difficultés éthiques posées par l'usage de ces outils, en l'absence de législation claire, ces entreprises ont décidé en 2020 de stopper leur activité de recherche en la matière.

15. Score F2 de 0.86 ; pour des précisions sur la définition et l'interprétation des scores de type $F\beta$ usuels en classification binaire, voir par exemple Powers (2011).

pré-entraîné permettait d'atteindre de meilleures performances[16], mais avec un coût en termes de développement et de temps d'entraînement bien supérieur : quelques minutes en regard de plusieurs jours. Il faut noter par ailleurs que l'entraînement d'un modèle non pré-entraîné durant un temps similaire ne permettait pas d'atteindre de semblables performances.

3.1.3. Recommandations pratiques et conclusions

Développer ou non une solution utilisant des réseaux de neurones profonds, dans un cadre supervisé, repose donc majoritairement sur les deux axes suivants.

- **La facilité à construire des caractéristiques explicatives**. Pour des problématiques simples de détection d'objets ou de classification d'images, des histogrammes des couleurs suffisent généralement pour entraîner un modèle simple d'apprentissage atteignant de bonnes performances.

- **La spécificité du problème**. Supposons que le problème et le format d'images soient communs, telle la classification d'images standard représentant des voitures ou des vélos. De nombreux modèles pré-entraînés très performants, de réutilisation rapide, sont déjà disponibles, et la phase très coûteuse de création d'un jeu de données n'est en général pas nécessaire. Si les images disponibles sont communes mais que le modèle est peu spécifique, et nécessite par exemple l'ajout de nouvelles classes d'objets, il est alors possible d'obtenir assez rapidement des résultats corrects à l'aide d'un apprentissage spécifique des dernières couches des modèles d'apprentissage profond pré-entraînés. Le gain de cet apprentissage par transfert résulte de la réutilisation de poids optimisés pour une tâche proche (Sharif Razavian et al., 2014). Il importe cependant de bien comprendre l'architecture du réseau, notamment en exhibant des cartes de caractéristiques* par couche. En cas d'images très spécifiques – des images microscopiques par exemple – alors l'entraînement complet d'un réseau de neurones est nécessaire et les architectures habituelles ne sont parfois pas adaptées. Des jeux de données d'apprentissage de très grande taille sont indispensables pour ce faire.

Rappelons également que la mise en œuvre de techniques d'apprentissage profond dépend fortement des ressources matérielles disponibles. Hors pré-traitement des images et développement d'un réseau profond, un entraînement complet sur des images de grande taille peut durer plusieurs semaines sur des processeurs standards.

Retenons enfin de ce parcours des cas d'usage que l'AP dédié au traitement automatique des images et des flux d'images est devenu globalement mature, à condition que la diversité des phénomènes qu'elles représentent soit bien délimitée, et que des données labellisées existent en grand nombre. L'absence de biais et la capacité de généralisation des outils d'AP en dépend, et il n'est pas évident de définir une typologie précise de ce qu'est le périmètre d'une image. Une image de chien prise sur un fond neutre n'apporte pas la même information qu'une image du même chien sur fond de verdure, ou de neige[17] Il est donc aisé d'introduire des biais préjudiciables par la sélection des données, tout en pouvant difficilement les contrôler.

Les réseaux de neurones convolutifs sont donc devenus les outils fondamentaux de ce type d'AP dans un cadre supervisé ; ils sont à présent bien compris et très largement utilisés, au-delà même du domaine du traitement d'image (nous les retrouverons notamment en traitement de série temporelle). Ils se spécialisent de plus en plus dans l'apprentissage des structures

16. Score F2 de 0.92.

17. En référence à un exemple célèbre de confusion entre un loup et un husky photographié sur fond neigeux, du fait que les images de loups disponibles dans la base d'entraînement avaient été réalisées dans un paysage enneigé ; voir par exemple Besse et al. (2019).

géométriques internes à ces images, mais deviennent d'autant plus gourmands en données d'entraînement labellisées, coûteuses par nature. Afin de proposer des architectures de réseaux pertinentes, sinon sobres et donc moins boulimiques en données, des outils de statistique classique permettent de produire des pré-traitements très efficaces. Par ailleurs, les approches statistiques markoviennes à état latent, qui permettent de mener des analyses non supervisées ou semi-supervisées, restent largement détachées des schémas algorithmiques d'AP ; il nous semble que leur capacité de labellisation et leur frugalité en termes de données sont encore insuffisamment connues des praticiens de l'AP, qui gagneraient à les étudier afin de les employer en interaction avec leurs propres outils.

3.2. Analyse de signaux temporels

3.2.1. Quelques cas d'usage fondamentaux

Les séries temporelles peuvent être générées par de nombreux processus (Forestier et al., 2017; Jones et Lorenz, 1986; Kegel et al., 2018) et nécessitent une analyse spécifique en raison de propriétés propres, notamment les possibles saisonnalités, auto-corrélations des séries et tendances (Shumway et Stoffer, 2017). La typologie des cas d'usage considérés majoritairement par les entreprises est la suivante.

- **Prévision.** Il est parfois crucial de connaître l'évolution d'une quantité à l'avance, afin de prendre des décisions ou anticiper d'éventuelles difficultés. La prévision de séries temporelles est particulièrement éprouvée dans le secteur de la finance (Sezer et al., 2020) et pour les séries économiques en général (Makridakis et al., 2009), ou bien encore dans le domaine de la vente, pour prévoir la demande ou un volume de vente à un horizon temporel donné (Fildes et al., 2019). L'incertitude autour de la valeur prévue peut être, selon les cas d'usage, aussi importante que la prévision en elle-même (Makridakis et al., 2009), et ce en particulier pour les problèmes à forte asymétrie (cas de la prévision de demande intermittente avec de forts volumes (Seeger et al., 2016)). L'état de l'art en matière de prévision converge aujourd'hui vers des modèles probabilistes, comme en témoigne la compétition de prévision M4 (Makridakis et al., 2018) et les dernières approches intégrant des réseaux de neurones visant à apprendre les paramètres d'une densité de probabilité (Salinas et al., 2019) ou d'une fonction quantile (Gasthaus et al., 2019).

- **Classification.** L'objectif est de réaliser une correspondance entre des segments de séries temporelles et un ensemble de catégories. Il peut s'agir de classifier des segments d'une même série (par exemple, identifier les périodes de sommeil au cours de la nuit d'un individu (Chambon et al., 2018)) ou de plusieurs séries (par exemple, la reconnaissance d'une personne en particulier par un assistant vocal (Längkvist et al., 2014)). La détection et l'identification de formes caractéristiques permettant l'attribution d'une série temporelle à une classe (Schäfer et Leser, 2020) peuvent se révéler, selon les cas d'application, particulièrement critiques – tel le suivi de la série temporelle des battements cardiaques d'un patient. *De facto*, la classification de cette série le plus tôt possible peut permettre un diagnostic plus précoce et une meilleure adaptation du traitement.

- **Segmentation.** On peut également souhaiter segmenter de façon non supervisée des séries temporelles afin d'identifier des groupes homogènes. Un exemple typique de cas d'usage est l'identification de profils clients, par exemple de consommateurs d'électricité pour un fournisseur d'énergie (Benítez et al., 2014). Dans le secteur de la santé, ce genre

de technique peut être utilisé pour classer des profils d'IRM fonctionnelles (Wismüller et al., 1998).

Par ailleurs, deux cas d'usage fréquents peuvent être associés à plusieurs des catégories ci-dessus :

– **Maintenance prévisionnelle.** Après la maintenance corrective, puis préventive, la tendance est aujourd'hui à la maintenance prédictive (ou prévisionnelle). Elle a pour objectif de limiter les coûts en cas de panne, les effets domino au sein d'un réseau ou encore la durée d'interruption d'un service. Ce sujet peut être abordé de deux manières : soit on s'intéresse à l'estimation du temps restant avant une panne (ce qui correspond traditionnellement à de l'analyse de survie (Talamo et al., 2019)), soit on cherche à opérer une classification à $t + x$ (panne / susceptibilité de panne / non-panne), x étant l'horizon temporel souhaité (Jahnke, 2015).

– **Détection d'anomalie.** Au-delà de la prévision d'une panne ou d'une fraude à partir d'un historique de données labellisées (Ferdousi et Maeda, 2006), on peut souhaiter identifier de nouveaux types d'anomalies au fil de l'eau : on parle alors d'apprentissage en ligne (*online*), à l'opposé de l'apprentissage par lot (*batch*), qui permet de détecter plus tôt les anomalies, en traitant les données et en mettant à jour le modèle en continu (Guo et al., 2016). C'est par exemple le cas dans les systèmes anti-fraude, où les typologies de fraude sont changeantes par nature (Seyedhossein et Hashemi, 2010). On les identifie alors comme des déviations par rapport à une norme[18] définie conjointement par le *data scientist* et les experts métiers. Ces anomalies peuvent être ponctuelles, contextuelles ou collectives (Choudhary, 2017).

3.2.2. Outils et apports de l'apprentissage profond

La nature particulière des séries temporelles implique des difficultés spécifiques. Deux points de mesure égaux à t n'engendrent pas forcément la même prédiction à $t + x$, en raison de la prise en compte des effets saisonniers et des tendances. De plus, les séries peuvent être très bruitées (en particulier les séries physiques issues de capteurs (Yao et al., 2017; Martí et al., 2015)) et peuvent être amenées à être appréhendées sous une forme multidimensionnelle (ainsi, des données issues de stations météorologiques et sismiques peuvent être utilisées conjointement pour la prévision climatique (Groves-Kirkby et al., 2006)). La stationnarité de la série temporelle est également un prérequis important pour l'utilisation de certains modèles (Dickey, 2005; Brockwell et Davis, 2016), en particulier les modèles statistiques de type ARMA, ARIMA (Woodward et al., 2017), ou le lissage exponentiel.

Enfin, une difficulté majeure est la nécessité assez récurrente de disposer de variables explicatives du phénomène étudié afin de caractériser ses différents cycles, le plus souvent construites manuellement. La connaissance métier apparaît donc primordiale afin de définir de telles variables *a priori*.

Les RNN profonds permettent de pallier certaines de ces difficultés. Ainsi, ils peuvent relativement aisément **prendre en compte des processus non-linéaires** dans la modélisation des données, ce qui permet de réduire le temps de pré-traitement (Schörgenhumer et al., 2019). Par ailleurs, ils permettent de **diminuer les hypothèses restrictives** sur les données, telle la stationnarité ou encore l'hypothèse des risques proportionnels, permettant là aussi de gagner du temps de pré-traitement (débruitage, suppression de la saisonnalité, etc. (Kusdarwati

18. Ou plus généralement une fonction de coût.

et Handoyo, 2018)). Les principaux RNN profonds qui, à présent, ont fait l'objet de multiples expérimentations réussies sont les suivants :

- Les *réseaux de neurones récurrents* (RNR), et en particulier les réseaux *Long Short Term Memory* (LSTM) (Gers et al., 1999), ont vocation à analyser des séquences et résolvent le problème de la dépendance à long terme inhérente à certaines séries temporelles, dans lesquelles deux points de mesure éloignés conservent une dépendance significative (Hochreiter et Schmidhuber, 1997). Les RNR sont construits sur un principe simple issu de l'étude des systèmes dynamiques : chaque élément de la sortie du réseau est une fonction des éléments précédents de la sortie. Le réseau permet alors d'approximer cette fonction de récurrence, soit indirectement en incorporant des connexions récurrentes entre unités cachées (voir Figure 5), soit directement via des relations de rétroaction entre la sortie et les couches cachées ; fonction que l'on traduirait, dans un cadre statistique classique, comme un opérateur markovien.

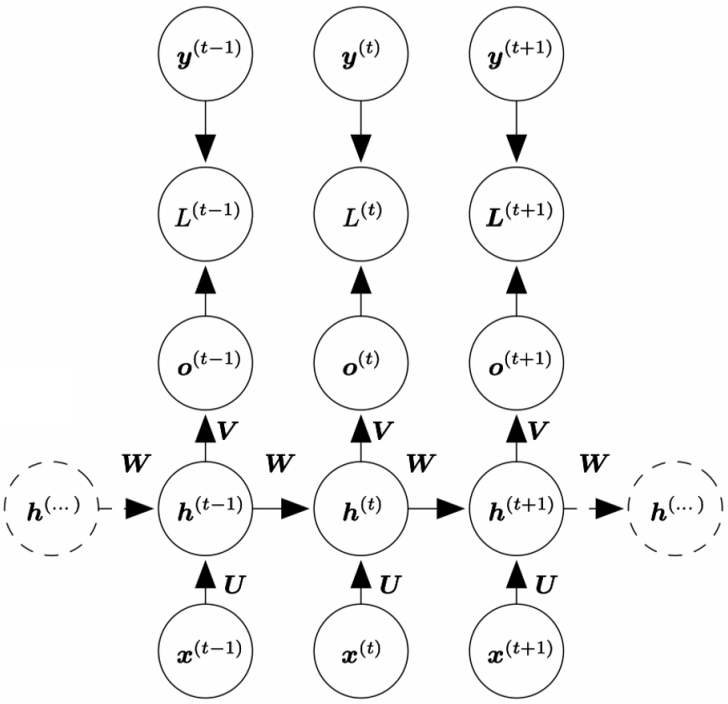


FIGURE 5 – *Illustration d'un réseau récurrent (RNR) à propagation avant présentant le calcul de la fonction de perte d'entraînement d'un réseau récurrent, qui associe une séquence d'entrée de valeurs* $\mathbb{x} = (\dots, x_{t-1}, x_t, x_{t+1}, \dots)$ *à une séquence correspondante de valeurs de sortie* $\mathbb{o} = (\dots, o_{t-1}, o_t, o_{t+1}, \dots)$*. Une perte* **L** *mesure une distance entre* $\mathbb{o}$ *et la cible (label) correspondante* $\mathbb{y} = (\dots, y_{t-1}, y_t, y_{t+1}, \dots)$*. Le RNR comprend des connexions cachées paramétrées par des poids* **U***, des connexions récurrentes entre couches cachées paramétrées par des poids* **W** *et dont l'ensemble des configurations est représenté par* $h(t)$*, et des connexions entre la sortie des couches cachées et les sorties du réseau* $\mathbb{o}$ *par des poids* **V***. Graphe inspiré de la Figure 10.3 de Goodfellow et al. (2016).*

Les LSTM octroient une grande liberté de paramétrage, ce qui, actuellement, peut parfois freiner leur utilisation en conditions réelles. Le temps d'entraînement reste également assez élevé par rapport aux méthodes d'apprentissage automatique plus traditionnelles. Une alternative peut être d'utiliser des *Gated Recurrent Units* (GRU) qui possèdent moins de paramètres que les LSTM et sont plus rapides à entraîner. Dans certains cas, tel que la prévision de flux routiers, les performances obtenues via des GRU sont mêmes

meilleures que via des LSTM (Fu et al., 2016).

- Les réseaux convolutifs (RNC) issus de l'analyse d'image se révèlent également intéressants pour la classification (Qian et al., 2020) et la prévision (Borovykh et al., 2017) de séries temporelles. Cette adaptation a notamment pour origine l'idée qu'une série temporelle peut être décrite comme une suite de valeurs qui peut être structurée, possiblement après transformation, sous la forme d'une image (Hatami et al., 2018)[19]. Généralement plus rapides à entraîner que les réseaux récurrents, les RNC ont néanmoins quelques inconvénients qui peuvent rendre difficile leur application aux séries temporelles (ex. : discordance de taille entre l'entrée et la sortie du réseau, entrée du réseau de taille fixe). Ils s'adressent donc à des applications très spécifiques.

- L'*auto-encodeur** (Sun et al., 2016) est également un RNN intéressant que nous avons utilisé dans plusieurs contextes, tel celui de la détection de défauts sur des séries temporelles (Lv et al., 2017). Sa capacité à extraire des variables latentes par reconstruction du signal original a fait ses preuves, ce qui en fait un candidat idéal pour la segmentation et la réduction de dimension, notamment pour les séries temporelles. Il peut être vu dans ce dernier cas comme une généralisation de l'analyse en composantes principales (ACP). L'auto-encodeur possède aussi des applications moins usuelles : il est par exemple possible de l'utiliser en amont d'un modèle, afin de contrôler la cohérence des données à différents moments.

3.2.3. Limitations et approches alternatives

D'une manière générale, l'expérience nous a appris que le déploiement de techniques d'AP, dans le cas spécifique du traitement des séries temporelles, doit être très précautionneux, et qu'en première intention, des approches plus classiques se révèlent robustes et compétitives.

Dans un contexte de **prévision**, les modèles paramétriques (ARMA et ses variantes de la famille Box-Jenkins (Brockwell et Davis, 2016), Holt-Winters (Chatfield, 1978) ou les modèles de Markov cachés (MacDonald et Zucchini, 1997)) ont fait leurs preuves depuis longtemps. Des approches non paramétriques, comme le krigeage par processus gaussiens (Espinasse et al., 2011) sont également indiquées. L'objectif est alors de trouver une fonction d'approximation des données par inférence bayésienne. Il y a donc, potentiellement, une infinité de paramètres en jeu.

Les outils de l'AM, par exemple les forêts aléatoires (Dudek, 2015) ou le *gradient boosting* (Taieb et Hyndman, 2014), restent également très compétitifs en pratique. Dans les cas de classification supervisée, la difficulté principale est de construire des variables explicatives (Deb et al., 2017). La transformée de Fourier (Elbir et al., 2018) et la transformée en ondelettes (Emanet, 2009) peuvent être d'une aide précieuse dans ce cas de figure. Lorsque la classification devient un problème de segmentation non supervisée (Zakaria et al., 2012), après avoir sélectionné une mesure de similarité adéquate (par exemple le *Dynamic Time Warping* (Cuturi et Blondel, 2017)), on procède de façon usuelle à une réduction de dimension (Keogh et Pazzani, 2000).

En **détection d'anomalie**, enfin, une technique traditionnellement utilisée est la modélisation par machines à vecteurs de support (SVM, ou *Séparateurs à Vastes Marges*) à une classe (Ma et Perkins, 2003). Dans les cas les plus simples, un modèle fondé sur des règles métier et des critères statistiques simples peut être utilisé (Shipmon et al., 2017) avec des résultats très

19. Ce parallèle prend plus de sens encore lorsqu'on considère une vidéo, qui est une série temporelle d'images.

honorables, et aisément intelligibles : par exemple, les points de mesure dont la distance à la médiane mobile dépassent un certain seuil peuvent être considérés comme des anomalies.

3.2.4. Recommandations pratiques et conclusions

Plusieurs considérations sont à prendre en compte pour s'orienter ou non vers l'apprentissage profond lorsqu'on souhaite résoudre un problème impliquant le traitement de séries temporelles.

Quantité et qualité des données disponibles. Il est conseillé d'avoir à disposition un volume de données conséquent pour tirer profit des réseaux de neurones (Remus et O'Connor, 2001). En cas de quantité faible (< 700 points), krigeage et méthodes à noyau apparaissent encore préférables pour la prédiction. La complexité (en $O(n^3)$) de ces types de méthodes les rendent difficilement utilisables en pratique avec plus de données (Hensman et al., 2013). Des méthodes statistiques simples, telles que ARIMA ou Holt-Winters, sont également indiquées dans un contexte où peu de données sont disponibles (Burger et al., 2001).

En présence de valeurs manquantes et d'un manque d'historique, certaines techniques ont fait leur apparition, telles que la factorisation de matrices et le filtrage collaboratif (Xie et al., 2016). Dans le secteur de la vente de détail, la prévision de vente de nouveaux produits ayant peu d'historique n'est pas une tâche aisée avec des modèles statistiques ou par apprentissage machine classique. De nouvelles approches, permettant l'intégration de ces produits, dits *cold starts*, intègrent des réseaux de neurones (Alexandrov et al., 2020; Salinas et al., 2019), mais elles restent à déterminer.

Caractérisation des données ou du phénomène. Il faut parfois transformer les séries temporelles *via* une méthode de Box-Jenkins (Helfenstein, 1986), certaines hypothèses (de stationnarité par exemple) devant être vérifiées pour permettre l'emploi de modèles comme ARMA. Si la qualité ou la complexité des données rend cette transformation complexe, il peut être plus intéressant d'utiliser un réseau de neurones ; leur intérêt est double : comme indiqué précédemment, ils peuvent permettre d'alléger la tâche souvent fastidieuse de la création de variables explicatives (Le Cun et Bengio, 1995). Cependant, certaines études ont montré que les performances d'un modèle SARIMA peuvent égaler celles d'un réseau de neurones (Camara et al., 2016). Dans ce cas de figure, le modèle le moins complexe (SARIMA) doit être préféré pour respecter le principe de parcimonie.

Horizon de prévision. Pour un horizon de prévision court terme, les modèles paramétriques tels que ARMA donnent des résultats satisfaisants et sont simples à mettre en œuvre. Dans le cas contraire, on favorisera plutôt le LSTM pour sa capacité à utiliser les dépendances de long terme. C'est aussi le cas du krigeage dans une certaine mesure (Haji Ghassemi et M., 2014). En fonction de la multiplicité des horizons temporels à prévoir, le choix d'une méthode intégrant des réseaux de neurones peut être pertinent. En effet, dans une approche directe de prévision multi-horizons, les modèles sont démultipliés avec le nombre d'horizons à prédire (Bontempi et al., 2013). Cela entraîne des coûts d'entraînement et de maintenance trop élevés. Une alternative est alors d'utiliser une approche récursive, la prévision à l'instant précédent *t+1* étant utilisée en entrée pour prédire l'instant *t+2*. Cette approche a pour défaut principal la propagation d'erreur, augmentant avec le nombre d'horizons temporels à prédire. Une hybridation directe-récursive des deux méthodes est possible (Taieb et al., 2012) pour tenter de contourner la limitation de l'approche précédente, en utilisant plusieurs modèles pour chaque horizon temporel et en incorporant la prévision à l'instant *t+1* en entrée du modèle de prévision de l'instant *t+2*. L'approche « séquence à séquence », permise par les réseaux de neurones, permet de conserver un seul modèle ayant pour objectif de prédire plusieurs ho-

rizons temporels en une seule phase d'inférence (Mariet et Kuznetsov, 2019). Ces modèles sont plus lents à entraîner et requièrent plus de données que les modèles précédents, mais proposent une solution élégante pour la prévision multi-horizons.

Vers des méthodes hybrides. Une récente avancée de l'état de l'art, permise en partie par la compétition de prévision M4 (Makridakis et al., 2018), montre que la communauté dirige actuellement ses efforts de recherche vers des méthodes hybrides, tirant parti du meilleur des deux mondes, en combinant approches statistiques et modèles d'apprentissage classique et profond. L'approche gagnante de la compétition M4 (Smyl et al., 2018) combine des réseaux de neurones récurrents et un modèle de Holt-Winters, les paramètres globaux (poids du réseau de neurones) et les paramètres des séries temporelles (composantes initiales de la saisonnalité et coefficients de lissage) étant appris à l'entraînement par descente de gradient. Cette approche a été développée sur la base du constat que les réseaux de neurones, dans un contexte de prévision incluant des séries temporelles aux saisonnalités complexes et hétérogènes, ne capturent pas de manière performante la saisonnalité. Cette méthode, gourmande en données et ressources computationelles, a récemment été implémentée sur GPU (Redd et al., 2019).

D'autres méthodes, fondées sur la décomposition, visent à simplifier les séries temporelles avant de les présenter à une réseau de neurones. Bien que les algorithmes de désaisonnalisation ont été conçus pour atteindre d'autres objectifs que celui d'être un bon pré-traitement pour les réseaux de neurones, cette étape de décomposition apparaît comme bénéfique pour les ensembles de données provenant de sources de données disparates (Bandara et al., 2020). Cette méthodologie combine une série de techniques de décomposition multisaisonnière pour compléter la procédure d'apprentissage des réseaux LSTM.

D'autres méthodes d'hybridation existent, combinant modèles statistiques (ARMA) et modèle d'apprentissage machine (*Gradient Boosting Machine*) (Hochard et Blanche, 2019). Elles montrent une meilleure performance sur les premiers horizons temporels de prévision fournie par le modèle ARMA, et une meilleure performance sur des horizons temporels plus lointains permis par l'algorithme d'apprentissage machine classique. Ce modèle peut s'écrire :

$$y_{pred} = \alpha(t) \cdot y_{pred,ARMA} + (1 - \alpha(t)) \cdot y_{pred,GBM},$$

où $\alpha(t) = \exp(-\frac{t}{\lambda})$, λ étant appris par optimisation. D'autres approches originales sont en plein développement, qui s'appuient notamment sur l'apport d'un réseau de neurones pour estimer (apprendre) les paramètres de modèles ARMA (Callot, 2019).

Le dynamisme observé dans la littérature des quelques dernières années montre que le domaine de la prévision de séries temporelles évolue vite et voit aujourd'hui deux mondes historiquement opposés dans leurs approches converger vers la complémentarité.

3.3. Compréhension du langage naturel

À l'instar du traitement automatique des images, le traitement automatique du langage (TAL) a connu un important changement de paradigme avec l'avènement de l'apprentissage profond (Manning, 2015). Il est considéré comme un vecteur-clé de l'automatisation de bon nombre de processus, les données textuelles constituant de loin la majorité des données produites chaque jour dans le monde (Delen, 2014, chap. 6). En particulier, éventuellement associé à des outils de traduction automatisée de plus en plus puissants (Fan et al., 2020), le TAL se révèle précieux pour établir rapidement des bases de connaissances sur de nouveaux marchés, dialoguer avec des partenaires ou des clients (Zhang, 2017) ; son intérêt économique en fait logiquement aujourd'hui l'une des compétences les plus recherchées par les entreprises de l'économie digitale.

3.3.1. Quelques cas d'usage fondamentaux

Outre des cas d'usage bien connus comme l'élaboration de *chatbots*, qui nécessite deux briques de compréhension du langage (compréhension des messages d'un utilisateur et génération d'une réponse) (Mnasri, 2019), les principales applications du TAL rencontrées ces dernières années ont essentiellement trait à la connaissance client, la santé et l'optimisation de processus opérationnels. Ainsi,

- **Extraction de sujets**. On souhaite faire ressortir les thèmes principaux de courriels de réclamations de clients afin de comprendre les principales sources d'insatisfaction[20].
- **Classification de documents**. On peut classifier l'objet d'un mail en « privé » ou « professionnel », pour respecter la vie privée des employés. On peut également faire de l'analyse de sentiments, comme dans le cadre de campagnes marketing.
- **Extraction d'entités nommées**. Dans de nombreux contextes, il est intéressant d'extraire des informations de manière automatisée. Ainsi, les compétences majeures du CV d'un candidat, ou bien les chiffres clés d'un document financier, sont des entités couramment cherchées.
- **Moteur de recherche**. Il est important de pouvoir identifier rapidement les informations pertinentes dans une base de documents. Par conséquent, les cas d'usage de moteur de recherche dans des documentations textuelles sont assez fréquents, notamment dans l'industrie pharmaceutique, les industries complexes ou encore le droit.

3.3.2. Outils et apports de l'apprentissage profond

Le TAL place le problème de la compréhension d'un énoncé dans un cadre temporel : une phrase est une séquence de mots, lue progressivement. Mais à la différence d'une série temporelle, la lecture d'un énoncé peut être réalisée dans les deux sens, et plusieurs parcours de l'énoncé sont en général nécessaires pour arriver à exprimer des éléments de contexte qui vont permettre d'aider à l'interprétation du message en vue de réaliser les tâches décrites au § 3.3.1. Les progrès en TAL (ou NLP en anglais) s'appuient sur trois piliers principaux, listés ci-dessous.

- Le développement de **modèles spécifiques au langage**. Les modèles récurrents à mémoire de type LSTM (Hochreiter et Schmidhuber, 1997; Cheng et al., 2016) et les mécanismes d'attention* (Vaswani et al., 2017) ont permis de modéliser les relations latentes entre les constituants d'un énoncé et, ces dernières années, de mieux capturer la structure séquentielle d'une phrase que des approches statistiques par chaîne de Markov (Goodfellow et al., 2016). Les états cachés de ces modèles sont utilisés comme des représentations globales d'un énoncé pour résoudre de nombreux cas d'usage, tels la traduction (Figure 6), l'analyse de sentiment ou la classification.
- **L'apprentissage auto-supervisé.** De nombreuses méthodes d'apprentissage de représentations s'appuient uniquement sur des corpus non labellisés. Ces méthodes tirent parti de la structure du texte pour construire des représentations qui sont utilisées *a posteriori* pour des cas d'usages. Ces formes d'apprentissage s'auto-enrichissant, de par la structure temporelle des données, diffèrent ainsi des traditionnelles approches supervisées d'AP qui supposent l'accès à des données labellisées (Glasmachers, 2017).

20. Voir par exemple https://github.com/MAIF/melusine et De Javel (2019) pour un exemple de package de TAL dédié à cette tâche.

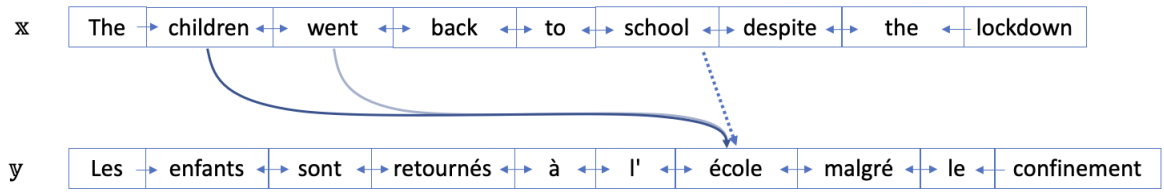


FIGURE 6 – *Illustration de l'utilisation d'un mécanisme d'attention dans une tâche de traduction. Celle-ci peut être réalisée par un auto-encodeur qui comprend deux parties : (a) l'encodage – une réduction de dimension – de l'information initiale* $\mathbb{x} \rightarrow \mathbb{z}$*, puis le décodage de cette information* $\mathbb{z} \rightarrow \mathbb{y}$*. Le vecteur* $\mathbb{z}$*, produit par une première série de transformations de type (1), est nommé représentation latente (ou vecteur de variables latentes). Un décodage terme-à-terme risque d'aboutir à une phrase maladroite voire incompréhensible. Pour éviter cela, pour chaque mot ou séquence de mots de* $\mathbb{x}$ *est construit un mécanisme d'attention permettant de contextualiser la traduction de cette entrée dans la sortie* $\mathbb{y}$*. Ces mécanismes d'attention alimentent donc le décodeur* $\mathbb{z} \rightarrow \mathbb{y}$*. Dans le cas présent, celui-ci utilise l'importance relative des mots « children » et « went » pour aider à générer le mot « école » dans une position adéquate dans la phrase* $\mathbb{y}$*.*

– Les **plongements** ou *embeddings* (Mikolov et al., 2013; Bojanowski et al., 2017), qui sont des représentations sémantiques des mots. Elles sont obtenues en entraînant des algorithmes à prévoir un mot en fonction des mots dans son contexte. Elles offrent de riches propriétés linguistiques et sémantiques, notamment le fait que deux mots de sens proches seront représentés par des vecteurs proches dans l'espace de représentation (Figure 7). Ces méthodes s'étendent à des structures plus complexes et cherchent à composer les représentations des mots pour obtenir des représentations sémantiques des phrases ou des documents. Les modèles de langues (Merity et al., 2018) proposent de prédire le mot suivant en fonction des précédents. Ils permettent de construire des représentations étonnamment robustes et des modèles génératifs extrêmement performants comme par exemple Open GPT ou ELMo (Radford et al., 2019; Peters et al., 2018). Des extensions des méthodes *d'embeddings* cherchent quant à elles à prévoir les phrases suivantes ou précédentes en fonction de la phrase courante pour apprendre des *embeddings* de phrases (Logeswaran et Lee, 2018).

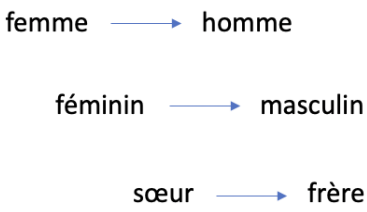


FIGURE 7 – *Les techniques de plongement, comme* word2vec *(Mikolov et al., 2013), proposent des représentations des mots dans un espace de dimension inférieure (à celle de l'espace de tous les mots possibles) dans lequel le sens des mots les rapproche, selon une distance statistique. Ici, la distance est similaire entre des mots différant par le genre.*

– **Le transfert de modèles pré-entraînés**. Finalement, des modèles plus versatiles, les *transformers** comme BERT ou XLNet (Devlin et al., 2019; Yang et al., 2019), sont

également entraînés par le biais de tâches auxiliaires qui s'apparentent à de l'auto-supervision : prévoir des mots masqués ou l'ordre d'apparition de deux phrases. Les *transformers* ont l'avantage d'utiliser simultanément plusieurs mécanismes d'attention en évitant les boucles de réccurence des LSTM, ce qui permet dès lors de paralléliser le traitement de suites de données (Vaswani et al., 2017). Ces modèles entraînés sur de gigantesques corpus peuvent ensuite être utilisés pour d'autres tâches auxiliaires *via* un mécanisme de *transfer learning* (Howard et Ruder, 2018). Leurs performances restent évaluées sur des benchmarks académiques, notamment la tâche GLUE, qui regroupe des ressources très variées, issues de textes de droit, de critiques de films, etc. (Wang et al., 2019).

Dans la majorité des situations, le besoin client implique de **combiner plusieurs algorithmes de compréhension du langage**. Par exemple, pour l'analyse de mails de retours de clients, on pourra commencer par une extraction de sujets non-supervisée. L'analyse de ces sujets pourra nous amener à définir des catégories, et à classifier un nouveau mail dans une de ces catégories (problème de classification supervisée). Les variables explicatives du problème de classification pourront être les plongements des mots (voir paragraphe suivant). On pourra enfin y ajouter la présence ou non d'entités nommées, extraites de manière automatique.

3.3.3. Limitations pratiques, défis et approches alternatives

Si les scores obtenus sur les *benchmarks* académiques dépassent aujourd'hui les performances humaines, des limites importantes subsistent encore, et freinent nettement l'industrialisation de méthodes performantes dans de nombreux contextes métiers.

Bien qu'il soit ainsi possible de construire des représentations fines du texte sans données labellisées, les cas d'usages finaux nécessitent donc d'importantes quantités de données labellisées, coûteuses à obtenir. Par ailleurs, des corpus existent, tel SNLI pour l'inférence (Bowman et al., 2015) ou pour la détection de paraphrases (Marelli et al., 2014), mais ces ressources sont anglophones et il n'existe généralement pas d'alternatives aussi riches pour d'autres langues. L'usage de modèles spécifiques au français, tel le modèle CamemBERT (Martin et al., 2019), débute tout juste.

Par ailleurs, les représentations textuelles sont généralement construites sur de gigantesques corpus dont les biais sont capturés dans les représentations (Bolukbasi et al., 2016) ou reproduits par les algorithmes (Barocas et Selbst, 2016). On observe notamment des biais pour le genre féminin/masculin. En outre, les prévisions peuvent se révéler inconsistantes entre elles (Li et al., 2019) et dans un problème de classification, deux exemples pourtant contradictoires peuvent être classifiés par erreur dans la même catégorie. Les algorithmes s'appuient sur un comportement statistique et ne possèdent pas de sens commun. Dans le cas de la traduction ou du résumé automatique, l'algorithme génère des mots statistiquement probables mais le sens global de la phrase peut être en contradiction avec l'énoncé original (Cao et al., 2018). Ces biais sont d'autant plus difficiles à corriger que les algorithmes sont complexes et peu intelligibles ; il reste encore aujourd'hui difficile d'expliquer une prévision et l'influence du jeu d'apprentissage sur cette prévision.

Enfin, la puissance de calcul colossale nécessaire pour entraîner ou réentraîner des modèles de TAL limite la capacité des acteurs socio-économiques à développer « leur » modèle pour leur domaine d'affaires, et présente un impact environnemental important (Strubell et al., 2020) (voir § 4.1). Certaines techniques moins évoluées, mais plus contrôlables, restent encore largement utilisées en pratique. Citons par exemple :

– l'usage d'*expressions régulières* ou *RegEx*, simples règles logiques sur des chaînes de caractères, qui restent très utiles pour nettoyer des textes et/ou implémenter des règles métier ;

– l'utilisation de la technique TF-IDF (*Term Frequency - Inverse Document Frequency*), utilisée pour la représentation vectorielle de documents (Robertson, 2004) ;

– l'utilisation de la technique LDA (*Latent Dirichlet Allocation*), qui permet l'extraction non-supervisée de sujets (L Griffiths et Steyvers, 2004) ;

– l'emploi de modèles de type HMM (*Hidden Markov Model*) et CRF (*Conditional Random Field*), qui permettent de prendre en compte la séquence des mots et de reconnaître des entités nommées (Sutton et McCallum, 2007).

3.3.4. Recommandations pratiques et conclusions

Le choix de développer ou non une solution utilisant des outils d'AP va être aiguillé selon différents axes, assez similaires à ceux de l'analyse d'images.

Il faut tout d'abord considérer l'*adéquation au besoin métier.* Si l'on considère par exemple le problème de l'extraction non-supervisée de sujets d'un corpus de textes, un LDA répond simplement au besoin. Par ailleurs, il peut être intéressant d'utiliser des règles métier, pour des raisons de simplicité ou de compréhension du modèle. Dans cette situation, de simples Regex peuvent déjà résoudre le problème de façon tout à fait satisfaisante.

Par ailleurs, la *spécificité du problème* joue également un grand rôle dans la décision de s'orienter ou non vers une solution d'AP. Le texte et le besoin métier correspondent-ils à des formats assez usuels ? Lorsque c'est le cas, on peut alors utiliser des modèles existants et déjà entraînés pour de la reconnaissance d'entités nommées.

Dans le cas contraire, les données labellisées sont-elles suffisamment nombreuses pour entraîner un réseau ? Ainsi, un RNR produisant une classification de sentiments aura typiquement un nombre de paramètres à entraîner en

$$O\,(\text{nombre de mots} \times \text{nombre de sentiments})\,.$$

Enfin, les *ressources matérielles disponibles* constituent le dernier élément critique déterminant cette orientation, et la faisabilité de l'AP sur un cas d'étude particulier. Elles s'articulent autour du temps alloué au développement et de la volumétrie des données disponibles.

Dans le cadre de la création d'un chatbot, par exemple, le temps de développement peut être long, surtout si on ne souhaite pas utiliser d'API[21] déjà existantes. La création d'une base d'apprentissage peut également s'avérer longue et fastidieuse dans le cas typique de l'annotation d'un corpus de textes.

Dans l'optique de développer une première solution logicielle en temps contraint, il sera alors préférable d'utiliser des méthodes classiques qui sont moins demandeuses en temps de développement, de collecte de données et d'entraînement.

21. *Application Programming Interface*, ou interface de programmation applicative, qui permet à des applications de communiquer entre elles. Un exemple courant d'API utilisant de l'AP est un traducteur automatique en ligne, comme *www.deepl.com*.

4. Discussion : deux enjeux majeurs de l'apprentissage profond

L'apprentissage profond aborde les problèmes de l'apprentissage statistique d'un point de vue concret, guidé par la mise en œuvre pratique d'idées relativement intuitives. Cependant, la nécessité de lutter contre les problèmes de dimension et de représentativité des données requiert de travailler sur des espaces de représentations éloignés des caractéristiques connues (visibles, immédiatement compréhensibles) des phénomènes produisant ces données. Elle impose d'employer des modèles finalement complexes, dont l'implémentation n'est pas aisée et doit faire appel à des techniques d'approximation et le recours à une forte puissance de calcul. Ces difficultés opératoires s'accompagnent donc de deux problématiques d'importance grandissante, qui ne sont pas abordées dans l'ouvrage mais nous semblent aujourd'hui fondamentales pour une adoption raisonnée de l'AP :

– les enjeux du coût environnemental de la mise en œuvre de ces algorithmes au sein de plateformes d'intelligence artificielle ;

– les enjeux liés à l'intelligibilité du traitement des données et des décisions intégrées à ces plateformes.

4.1. Enjeux environnementaux

4.1.1. Les coûts déraisonnables du calcul

L'utilisation des méthodes et outils de l'apprentissage profond requiert aujourd'hui des ressources de calcul élevées. À titre d'exemple, l'entraînement d'un modèle ResNet[22] à 152 couches sur le jeu de données `ImageNet` composé de plusieurs millions d'images nécessite une centaine d'epochs (itération sur l'ensemble des images)(He et al., 2016). Cela correspond à une dizaine de jours d'entraînement sur une carte graphique classique de type Nvidia GeForce GTX 1080 Ti. L'élaboration de modèles de TAL pour un large public, comme BERT ou XLNET, est particulièrement gourmand en énergie (Strubell et al., 2020). L'usage soutenu des solutions *cloud*, popularisées par les facilités offertes par les plateformes de traitement délocalisées des *cloud providers* (AWS, Microsoft, Google...)[23], se traduit par une forte consommation d'électricité (Mastelic et al., 2014). On estime qu'entre 2013 et 2030, la part de consommation énergétique mondiale dédiée au fonctionnement de ces centres de stockage et de calcul, passera de 1.4% à 20% (Andrae et Edler, 2015; Jones, 2018), et qu'en conservant ce rythme de développement cette industrie sera responsable de 14% des émissions mondiales de gaz à effet de serre en 2040 (Vidal, 2017). Marquant les esprits, un travail récent estime que l'empreinte carbone liée à l'entraînement de modèles algorithmiques profonds utilisés en traitement du langage naturel est cinq fois plus forte que celle d'une voiture durant sa période moyenne d'utilisation (Strubell et al., 2020).

4.1.2. À la recherche de solutions sobres

S'il est nécessaire de modérer ces pourcentages en objectant que le minage de cryptomonnaies et les utilisations ludiques (ex. : streaming) contribuent de façon prépondérante à cette empreinte carbone, le développement de technologies d'intelligence artificielle faiblement

22. Le ResNet (*Residual Network*) est un RNC célèbre pour avoir remporté le challenge ILSVRC en 2015 (He et al., 2016).

23. On peut regretter qu'aucune plateforme européenne n'ait aujourd'hui émergé, ce qui n'est pas sans poser des problèmes de souveraineté ; le projet européen GAIA-X vise à combler ce déficit, avec un lancement prévu en 2021 : https://www.data-infrastructure.eu.

gourmandes en ressources de calcul répond à une double nécessité de profitabilité (Centemeri, 2009) et d'opérationnalité. Certaines approches prédictives actuelles, telles la lecture d'imagettes médicales issues de scanners 3D, nécessitent encore plusieurs minutes ; l'adoption technologique en est retardée. Le développement de matériels spécifiques, voire la modification de paradigmes de calcul (tels les calculs optique et quantique), associé à celui de techniques d'« élagage* » (*pruning*) de modèles d'AP (Molchanov et al., 2017), de réutilisation de résultats localisés d'entraînement (*adaptive deep reuse* ; Ning et al., 2019) et d'exploitation optimisée des données lors des phases préliminaires de l'entraînement (*data echoing* ; Choi et al., 2020), offrent des perspectives *a priori* prometteuses explorées par des laboratoires publics et privés. La forte réduction depuis 2013 des coûts de calcul vectoriel au sein des CPU classiques (*Central Processor Units*) (Gottschlag et al., 2020), celle des coûts de calcul matriciel dans les GPU (*Graphical Processor Units*) (Navarro et al., 2020) peuvent laisser penser que les équipementiers (ex. : Intel, Nvidia, Microsoft, etc.) ont progressivement cessé de suivre les besoins des utilisateurs. C'est toutefois encore à modérer : la plupart des technologies régulièrement produites par ces fournisseurs restent très chères et énergivores et s'adressent généralement à un large public[24]. Le pari fait par ces entreprises, telle Google et sa récente TPU (*Tensor Processor Unit*), est d'apporter une réponse à des problématiques de performance et de coût très ciblées, sur des typologies d'applications relativement restreintes. S'ils ont aujourd'hui des consommations électriques équivalentes à celles des GPU, les TPU proposent une efficacité bien supérieure dans des domaines très spécifiques et constituent des réponses moyen terme à la consommation d'énergie. Il convient également d'ajouter d'importants progrès réalisés ces dernières années sur les plateformes logicielles et les compilateurs* auxquels elles font appel, élaborés de façon à réduire les temps d'entraînement et d'inférence. Les approches de calcul distribué pour l'entraînement, qui trouvent racine dans l'univers du HPC (*High Performance Computing*), sont ainsi de plus en plus optimisées[25].

4.1.3. Des optimisations croisées

Ces différentes optimisations croisées, méthodologiques, logicielles et matérielles, sont généralement difficiles à appréhender simultanément (sinon au travers d'ouvrages transverses comme le travail de référence de Ezratty (2018)) et apparaissent aujourd'hui d'autant plus importantes dans le contexte de déploiement de technologies 5G à l'échelle mondiale. La problématique des calculs délocalisés (*edge intelligence*), favorisant *a priori* la protection des identités privées, est particulièrement explorée. Ainsi, *l'apprentissage fédéré* (Bonawicz et al., 2019), qui consiste à traiter des calculs en local et remonter des résultats et une mise à jour des modèles, constitue l'une des pistes applicatives les plus discutées depuis deux ans dans les conférences internationales comme NeurIPS. Les problèmes posés par des qualités de données inégales, peu ou non contrôlées (et qui ne sont pas sans rappeler des difficultés constatées en *science participative* (Wiggins et al., 2011)), voire inconnues de l'utilisateur final, sont autant de pistes de recherche (Augenstein et al., 2020), dont l'une des contraintes permanentes est la sobriété énergétique.

4.2. Enjeux d'intelligibilité

24. Outre les spécialistes de la donnée, les *gamers* constituent un marché florissant.
25. Citons par exemple Treelite, TVM et des plateformes comme Horovod pour le calcul distribué, ou enore OpenVino pour les problèmes de vision par ordinateur. Voir Mosavi et al. (2019) pour plus d'information.

4.2.1. Un enjeu de certification

L'intelligibilité au sens large des modèles d'AP est devenue, ces dernières années, une question récurrente des utilisateurs finaux – que ceux-ci soient des décideurs ou des praticiens – mobilisant fortement les sciences appliquées et humaines (Weld et Bangal, 2019; Abdul et al., 2018). Ce problème s'étend au cadre plus global des IA connexionnistes. En effet, une technologie d'IA peut être vue comme un ensemble d'outils permettant d'automatiser une prise de décision, reproduisant ou mimant éventuellement une action humaine, et en augmentant certains effets (gain de temps et de précision en particulier). La localisation des données et des modèles entraînés en est une caractéristique fondamentale : *on premise*, *cloud*, *on edge*, etc. D'une manière générale, les modèles d'AP sont des outils de calcul implémentant des relations causales connues ou postulées (dynamiquement ou non) à partir de données. La prise de décision engendrant la réalisation d'un ensemble de tâches, produisant des sorties à partir d'entrées, une IA moderne fondée sur l'AP peut être assimilée à un procédé technique de transformation, susceptible d'être qualifié, certifié, reproduit, breveté. Autant de problématiques concrètes et majeures qui concernent tous les secteurs d'activité.

En Europe, la mise en application du Règlement général sur la protection des données (RGPD) à partir de mai 2018 a fortement contribué à cristalliser des questions portant sur le comportement de ces modèles, des algorithmes auxquels ils sont incorporés et de la pertinence des données elles-mêmes. D'une manière générale, on souhaite pouvoir expliquer le comportement des procédés technologiques opérant automatiquement à partir des données, ainsi que déterminer le caractère « raisonnable » de ces données. Ces interrogations légitimes parfois regroupées sous le mot-valise *intelligibilité* ont encore à être précisées au moyen d'une sémantique clarifiée (éthique, asymétrie d'information, justesse, loyauté, etc.), tâche entreprise entre autres par Pégny et Ibnouhsein (2018), Besse et al. (2019), Bertail et al. (2019) et Pégny et al. (2019).

4.2.2. Écueils sémantiques et culturels

Ainsi, selon Pégny et Ibnouhsein (2018) et Besse et al. (2019), une décision algorithmique sur une situation est dite *explicable* s'il est possible d'en rendre compte explicitement à partir de données et caractéristiques connues de la situation – c'est-à-dire s'il est possible de mettre en relation les valeurs prises par certaines variables $\mathbb{x}$ et leurs conséquences sur la prévision $\mathbb{y}$, et ainsi sur la décision. Elle est par ailleurs dite *interprétable* s'il est possible d'identifier les variables qui participent le plus à la décision, voire même d'en quantifier l'importance.

En pratique, deux typologies de problèmes sont fréquemment rencontrées.

- Des *problèmes introspectifs*, liés à des besoins très spécifiques, qui portent sur la compréhension des concepts « appris » par les modèles d'AP. Ainsi, dans le contexte du traitement automatisé d'images, les méthodes de *feature visualization* (FV, ou *saliency methods* (Denadai, 2018)) permet d'analyser les variables présentes dans les couches internes d'un réseau de neurones (c'est-à-dire les concepts appris) (Wei et al., 2015; Szegedy et al., 2015). La visualisation des interactions entre neurones est également un problème classique (Zhang et Zhu, 2019). L'interprétation des associations opérées par les techniques de traitement automatisé du langage naturel (écrit, parlé...) connaît un essor important (Spille et al., 2018; Akbari et al., 2019), parallèlement au développement toujours plus rapide des solutions de TAL grand public citées plus haut (BERT, XLNet...).

- Des *problèmes comportementalistes*, qui traitent de la compréhension des facteurs d'in-

fluence (d'une prévision, d'une classification...) détectés par les algorithmes. Ce type de problème très générique peut avoir une déclinaison *locale* (Guidotti et al., 2018), en s'intéressant au traitement spécifique d'un bloc de données (tels, par exemple, des clients « proches » qui se sont vus refuser ou octroyer un crédit), ou *globale* (ex. : comment expliquer une frontière de classification). De nombreuses méthodes globales existent, telles LIME ou SHAP (voir par exemple Gevrey et al. (2003) et Hénin et Le Métayer (2019b); Bénard et al. (2019) pour des revues), qui tirent en général parti de perturbations des données initiales (notons que des approches par rétropropagation, spécifiques aux réseaux de neurones profonds et plus rapides, sont également utilisées (Castillo et al., 2006)). Remarquons par ailleurs que de récents travaux appliqués au traitement d'images visent à proposer des solutions mixtes, tels les *Semantic Dicionnaries* (Olah et al., 2018) : chaque neurone ayant une contribution importante dans la prévision est représenté en utilisant une méthode de FV.

Notons également que l'étude de l'explicabilité et l'interprétabilité d'un modèle est souvent *post-hoc* – consécutive à la conception de ce modèle – ce qui n'est pas sans poser de sérieuses difficultés de rigueur (Laugel et al., 2019). Reprenant l'antagonisme connu entre explicabilité et qualité en prévision d'un modèle (Shmueli et Koppius, 2009), la communauté de l'apprentissage profond débat intensément sur la pertinence d'un tel paradigme (Lipton, 2018; Krishnan, 2019), tout en reconnaissant que l'interprétabilité est généralement indispensable pour un emploi maîtrisé des systèmes à base d'AP. La conception de systèmes directement interprétables, fortement soutenue par Rudin (2019), et qui s'appuie principalement sur des modèles de règles et d'arbres logiques, connaît aujourd'hui un nouvel engouement (Genewein et al., 2020), même si elle est loin d'être encore mature.

Enfin, au contact des applications et des donneurs d'ordre, nous constatons que les problèmes d'intelligibilité se heurtent à deux écueils majeurs : la représentation de l'information, relative là encore à un choix de sémantique, lui-même dépendant de caractéristiques culturelles (ainsi, un spécialiste saura comprendre et valider une certaine abstraction restreinte à son champ d'expertise) ; et un cloisonnement préjudiciable entre disciplines scientifiques. Ce second aspect est particulièrement marqué lorsqu'on s'intéresse aux problèmes comportementalistes. Ainsi, les problèmes d'intelligibilité locale peuvent être interprétés comme des problèmes de *déplacement minimal* de données : comment et pourquoi une donnée peut-elle être classée dans une catégorie A plutôt que B ? Au-delà d'exhiber des exemples dits *contrefactuels*, visant à « berner » le classifieur et tester sa robustesse (Mothilal et al., 2020) (Figure 8), il faut déterminer comment cette donnée se déplace d'une classe à l'autre, dans un certain espace mathématique ; il s'agit de la résolution d'un problème de transport optimal (Gordaliza et al., 2019).

4.2.3. Vers une démarche unificatrice ?

La détermination de l'intelligibilité locale ou globale d'un modèle complexe, au sens de la communauté du *machine learning*, ne correspond pas, jusqu'à présent, à une méthodologie unifiée, qui aurait comme propriété fondamentale d'être indépendante du choix de modèle (*model-agnostic*) et du mécanisme de génération des données (Molnar et al., 2020). À de multiples égards, elle pourrait bénéficier des nombreux travaux menés en analyse de sensibilité produits au sein de la communauté multidisciplinaire dédiée au traitement des incertitudes dans les outils de calcul scientifique (ou codes numériques) complexes[26]. En effet, le domaine du

26. En France, cette communauté et principalement représentée par le Groupement de Recherche CNRS MASCOT-NUM : http://www.gdr-mascotnum.fr

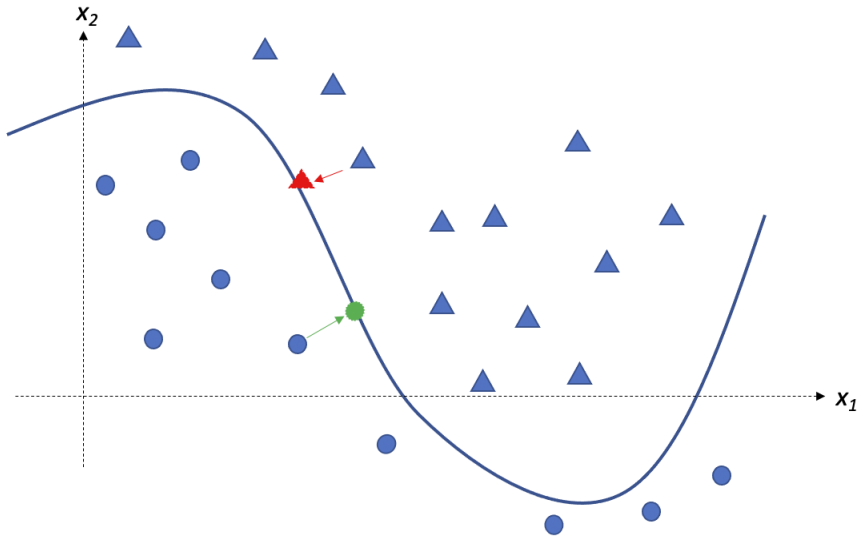


FIGURE 8 – *Deux exemples contrefactuels, en rouge et vert, issus de chacune des classes (triange, rond) : ils sont construits en « transportant » de façon optimale (avec le minimum de déplacement) des données de chaque classe vers la frontière de séparation. La sensibilité du classifieur (courbe pleine), représentée par exemple par le risque de mauvaise classification, est probablement élevée lorsqu'on l'applique à ces exemples. Une approche robuste est de chercher à limiter cette sensibilité.*

traitement des incertitudes suppose que l'on puisse définir la prévision d'une grandeur d'intérêt (scalaire, vectorielle, fonctionnelle, ...) Y par l'équation suivante

$$Y = g(X),$$

où g est un modèle de calcul causal, implémenté sous la forme d'un code numérique, et X un ensemble de *paramètres d'entrée*, variables ou processus aléatoires dont les caractéristiques peuvent être connues, postulées ou calibrées (typiquement par inversion de modèle). Si des réserves épistémologiques sont à prendre en compte (Denis et Varenne, 2019), ce cadre diffère faiblement du cadre classique du *machine learning*, où g est un (méta-)modèle déterministe ou stochastique, typiquement un réseau de neurones, une forêt aléatoire, etc., et la distribution X (dont les composantes sont donc appelées *features* ou *caractéristiques*) n'est connue qu'à travers des données. Les méthodologies d'analyse de sensibilité (Borgonovo et Plischke, 2016) apportent des réponses claires au problème de la définition d'indicateurs intelligibles. La correspondance entre des concepts bien établis dans le domaine de l'analyse de sensibilité et des indicateurs régulièrement utilisés en intelligibilité du *machine learning*, telle celle qui existe entre les indices de permutation d'une forêt aléatoire et les indices de Sobol (Gregorutti et al., 2017), est à améliorer.

Nous percevons donc que ces cloisonnements peuvent souvent générer des recherches de solutions sous-optimales, contraintes par des temps de développement et de test exagérément courts. Une méthode *post-hoc* populaire comme LIME (Tulio Ribeiro et al., 2016) pourrait être mieux utilisée (et mieux critiquée) si l'on comprend qu'elle cherche à approximer localement une variété topologique, sur laquelle se répartissent les données, de façon extrêmement grossière. Cette méta-modélisation constitue un retour en arrière vis-à-vis de méthodes dites de *surrogate models*, comme le krigeage par processus gaussiens, utilisées depuis des décennies par des statisticiens et numériciens dans le même contexte du traitement des incertitudes (Kleijnen, 2007). Il a déjà été démontré que l'analyse de sensibilité de tels méta-modèles

est susceptible de donner sous certaines conditions des informations valables sur le mécanisme réel reliant les données entre elles (Janon et al., 2012; Van Steenkiste et al., 2019). Les approches SMOTE de suréchantillonnage, fortement utilisées dans les problèmes de classification déséquilibrés (Fernández Hilario et al., 2018), font également le pari sous-jacent de simuler des données sur une variété ; il semble donc logique de chercher à déterminer les caractéristiques de celle-ci (*manifold learning*) avant de proposer une méthode de simulation. L'analyse topologique des données (TDA ; Chazal et Michel (2017)) est un candidat pour nous y aider. Le dynamisme de l'école française dans ce domaine laisse espérer que des résultats récents sur l'inférence des variétés topologiques (Aamari et al., 2019; Aamari et Levrard, 2019) puissent nettement améliorer la représentation des nuages de données. La pratique reste néanmoins très (trop) courante de simuler de nouvelles données en interpolant basiquement quelques données réelles, et d'introduire ainsi des hypothèses de convexité ou de concavité non désirées sur la variété sur laquelle elles se répartissent.

Exhiber ces liens entre des approches historiques, issues essentiellement de la statistique, de l'analyse numérique, de l'optimisation et de la géométrie, et des besoins modernes générant une intense activité dans la communauté de l'apprentissage nous paraît donc fondamental pour achever cette relecture critique de l'ouvrage *L'apprentissage profond*.

5. Conclusion

Ce retour d'expérience reste nécessairement schématique et partiel, de par le périmètre sans cesse grandissant de l'apprentissage profond. Dans la construction d'un produit analytique, les cas d'usage sont fréquemment amenés à se croiser ; des analyses de contenu peuvent être conduites sur des photographies de documents, ou des séries temporelles d'images commentées. On peut retenir que pour chacun des cas d'usage considérés, des approches historiques existent et apportent des résultats satisfaisants pour des objectifs spécifiques. Pour chacun également, l'apprentissage profond a indubitablement permis d'importantes avancées, que ce soit par le traitement de typologies de données différentes, la mémoire des séquences de mots ou la capacité d'abstraction qui facilitent la construction des variables utilisées pour la modélisation.

L'apprentissage profond reste encore, pour de nombreux domaines professionnels, une aide au diagnostic (que celui-ci soit médical, industriel, économique...) prometteuse plutôt qu'un outil intégré à un système décisionnel automatisé. Si une preuve de concept permet de démontrer une réelle plus-value de ces approches sur des modèles historiques d'aide à la décision, le passage en production requiert de démontrer le caractère généralisable et la robustesse de ces outils, qui nécessitent des architectures informatiques complexes et l'accès à des *clouds* sécurisés dont la disponibilité, la confidentialité et le coût (énergétique, politique...) constituent des enjeux critiques pour les entreprises et les institutions. Nous constatons que la sélection et la collecte de données variées utilisées pour nourrir ces algorithmes devient un métier à part entière, dont l'une des principales préoccupations porte sur l'existence de biais informationnels susceptibles de générer des résultats faux ou ambigus. L'abondante littérature consacrée depuis longtemps à l'objectivation des méthodes de sondage et d'échantillonnage (Schmidt et Hunter, 2015) nous prévient de l'ampleur et l'importance de cette tâche. Celle-ci doit également se doubler d'une réflexion sur les éventuelles déstabilisations économiques et distorsions de marchés que pourrait générer l'arrivée d'acteurs aux modes de gestion automatisée des ressources, faisant usage de méthodes dont la validité est si dépendante de la qualité et du nombre de données récoltées (Ezrachi et Stucke, 2017) .

Au-delà même du choix des données, la compréhension fine du sens d'une architecture et de l'apport d'information produit par les étapes de réentraînement apparaît essentielle aux ingénieurs : ceux-ci devraient idéalement permettre d'intégrer l'expertise technique en contraignant le réseau à respecter des propriétés essentielles du phénomène représenté (telle, par exemple, sa monotonie), des règles d'équité et de loyauté spécifiques au contexte d'utilisation, et à percevoir à quelles étapes du réentraînement les données *souveraines*, possédées par l'utilisateur, finissent par réduire significativement l'influence de données non souveraines, susceptibles d'injecter du biais.

Enfin, la certification de ces outils – supposons-les bien argumentés et rendus souverains – se heurte encore à des difficultés classiquement rencontrées en conduite du changement : les systèmes de règles intelligibles pour le métier, ou issus d'une longue pratique, ne peuvent être facilement évacués au profit de méthodes nouvelles, dont la compréhension fine et l'emploi éclairé restent, encore aujourd'hui, une forme d'art (Ovenden, 2018). Outre l'anticipation de difficultés sociales et économiques que le remplacement technologique engendre en permanence (Terrade et al., 2009), difficultés abondamment discutées dans le cas des outils d'intelligence artificielle (Wisskirchen et al., 2017), il nous semble avec Perez (2018) que des standards d'usage doivent à présent émerger afin de permettre leur déploiement. Cet enjeu est aujourd'hui au cœur de la stratégie nationale française sur l'IA, incarnée par le Grand Défi « Sécuriser, certifier et fiabiliser les systèmes fondés sur l'intelligence artificielle »[27], qui a vocation à piloter la réalisation de projets de recherche interdisciplinaires.

Pour réussir l'usage mature et raisonné de l'AP au sein du monde socio-économique, le besoin de nouveaux profils est à présent bien établi sur le marché du travail pour comprendre et mettre en production les algorithmes d'AP au travers du développement de produits d'IA. Les profils actuels de *data software engineers* sont en général issus de formations en informatique pure, mais leur implication dans les projets leur permet d'apporter un regard critique essentiel sur les méthodes de développement et d'implémentation d'algorithme. Ils sont amenés à refondre tout ou partie du code nécessaire à une bonne industrialisation, qui atteigne les exigences de robustesse imposées par les directions informatiques[28]. Cependant, la connaissance de l'ingénierie en technologie de l'information (IT) imprègne peu encore les formations de mathématiciens (et les statisticiens en particulier), qui ont un rôle fondamental à jouer pour formaliser, interpréter et prouver les revendications d'une méthodologie – et qui ont en particulier à établir des liens entre des approches historiques et de régulières « redécouvertes » faites par la communauté de l'AP[29]. En prenant connaissance des contraintes pratiques auxquelles le calcul en AP est soumis, il nous semble essentiel que ces spécialistes cherchent à les intégrer dans la formalisation de ces méthodologies et les étudient au même titre que les paramètres de modèles mathématiques. Pour cela, cette culture en IT doit croître au sein des formations[30] et devenir un pré-requis indispensable pour constituer les équipes des laboratoires d'IA en entreprise.

27. https://www.gouvernement.fr/grand-defi-securiser-certifier-et-fiabiliser-les-systemes-fondes-sur-l-intelligence-artificielle

28. Ex. : minimiser les cycles d'horloge*, traquer les fuites mémoires, apporter les éléments nécessaires au respect de la vie privée, etc.

29. Ainsi, un(e) statisticien(ne) bon(ne) connaisseur(euse) des méthodes de Monte Carlo peut mieux comprendre les approches GAN en les interprétant comme des algorithmes d'acceptation-rejet (Robert et Casella, 2004) d'un genre particulier.

30. Typiquement, la résolution d'un même problème dans des environnements de plus en plus proches de l'industrialisation, incluant des contraintes d'interopérabilité, constitue un axe de formation utile.

Remerciements

Les auteurs expriment toute leur reconnaissance aux contributeurs de Quantmetry pour l'aide apportée lors de la préparation de cet article, notamment Pierre Boszczuk, Jean-Matthieu Schertzer et Issam Ibnoushein, ainsi qu'à Fabien Navarro (ENSAI), qui a produit une partie des figures de la traduction française de *Deep Learning*. Ils souhaitent par ailleurs remercier les relecteurs anonymes et les membres du comité éditorial pour leurs commentaires et remarques qui ont grandement contribué à l'amélioration d'une première version de cet article.

Références

Aamari, E., J. Kim, F. Chazal, B. Michel, A. Rinaldo, et L. Wasserman (2019), «Estimating the Reach of a Manifold», *Electronic Journal of Statistics*, vol. 13, nº 1, pp. 1359–1399.

Aamari, E. et C. Levrard (2019), «Non-asymptotic rates for Manifold, Tangent Space and Curvature Estimation», *The Annals of Statistics*, vol. 41, nº 1, pp. 177–204.

Aas, K., L. Eikvil, et R. Huseby (2007), «Applications of hidden Markov chains in image analysis», *Pattern recognition*, vol. 32, pp. 703–713.

Abdul, A., J. Vermeulen, D. Wang, B. Y. Lim, et M. Kankanhalli (2018), «Trends and Trajectories for Explainable, Accountable and Intelligible Systems : An HCI Research Agenda», in «Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems», CHI '18, p. 1–18, Association for Computing Machinery, New York, NY, USA, doi :10.1145/3173574.3174156.

Akbari, H., B. Khalighinedjad, J. Herrero, A. Mehta, et N. Mesgararni (2019), «Towards reconstructing intelligible speech from the human auditory cortex», *Scientific Reports*, vol. 9, nº 874.

Akçay, S., M. Kundegorski, M. Devereux, et T. Breckon (2016), «Using deep convolutional neural networks for object classification and detection within x-ray baggage security imagery», *IEEE Transactions on Information Forensics and Security*, vol. 13, pp. 2203–2215.

Alexandrov, A., K. Benidis, M. Bohlke-Schneider, V. Flunkert, J. Gasthaus, T. Januschowski, D. C. Maddix, S. Rangapuram, D. Salinas, J. Schulz, L. Stella, A. C. TÃrkmen, et Y. Wang (2020), «GluonTS : Probabilistic Time Series Models in Python», *Journal of Machine Learning Research*, vol. 21, nº 116, pp. 1–6.

Anastasopoulos, L. et A. Whitford (2018), «Machine Learning for Public Administration Research, with Application to Organizational Reputation», *SSRN*, vol. 1, nº 3178287.

Andrae, A. S. G. et T. Edler (2015), «On Global Electricity Usage of Communication Technology : Trends to 2030», *Challenges*, vol. 6, nº 1, pp. 117–157, ISSN 2078-1547, doi : 10.3390/challe6010117.

As, I., S. Pal, et P. Basy (2018), «Artificial Intelligence in architecture : Generating conceptual design via deep learning», *International Journal of Architectural Computing*, vol. 16, pp. 306–327.

Augenstein, S., H. B. McMahan, D. Ramage, S. Ramaswamy, P. Kairouz, M. Chen, R. Mathews, et B. A. y Arcas (2020), «Generative Models for Effective ML on Private, Decentralized Datasets», in «Proceedings of the International Conference on Learning Representations (ICLR), Addis Ababa, Ethiopia, April 26-30», OpenReview.net.

Bandara, K., C. Bergmeir, et H. Hewamalage (2020), «LSTM-MSNet : Leveraging Forecasts on Sets of Related Time Series with Multiple Seasonal Patterns», *IEEE Transactions on Neural Networks and Learning Systems*, sous presse.

Barocas, S. et A. D. Selbst (2016), «Big data's disparate impact», *California Law Review*, vol. 104, p. 671.

Bénard, C., G. Biau, S. Da Veiga, et E. Scornet (2019), «Sirus : Making random forest interpretable», *arXiv :1908.06852*.

Bengio, Y. (2020), «Time to rethink the publication process in machine learning», *Note de blog*, https://yoshuabengio.org/2020/02/26/.

Benítez, I., A. Quijano, J.-L. Díez, et I. Delgado (2014), «Dynamic clustering segmentation applied to load profiles of energy consumption from spanish customers», *International Journal of Electrical Power & Energy Systems*, vol. 55, pp. 437–448.

Bertail, P., D. Bounie, S. Clémençon, et P. Waelbroeck (2019), «Algorithmes : biais, discrimination et équité», *Rapport TelecomParisTech & Fondation ABEONA*, https://www.telecom-paris.fr/algorithmes-biais-discrimination-et-equite.

Besse, P., C. Castets-Renard, et A. Garivier (2017), «Loyauté des Décisions Algorithmiques», *Contribution au Débat "Éthique et Numérique" de la CNIL*, HAL-01544701.

Besse, P., C. Castets-Renard, A. Garivier, et J.-P. Loubes (2019), «L'IA du Quotidien peut elle être éthique ? Loyauté des algorithmes d'Apprentissage Automatique», *Statistique et Société*, vol. 6, pp. 9–31.

Bhatt, U. (2018), «Maintaining the Humanity of Our Models», *Proceedings of the 2018 AAAI Spring Symposium. AI and Society : Ethics, Safety and Trustworthiness in Intelligent Agents*, vol. 1, pp. 18–22.

Bojanowski, P., E. Grave, A. Joulin, et T. Mikolov (2017), «Enriching Word Vectors with Subword Information», *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146.

Bolukbasi, T., K.-W. Chang, J. Zou, V. Saligrama, et A. Kalai (2016), «Man is to Computer Programmer as Woman is to Homemaker ? Debiasing Word Embeddings», in «Proceedings of the 30th International Conference on Neural Information Processing Systems», NIPS'16, p. 4356–4364, Curran Associates Inc., Red Hook, NY, USA.

Bonawicz, K., H. Eichner, W. Grieskamp, et D. *et al.*. Huba (2019), «Towards Federated Learning at Scale : System Design», *arXiv :1902.01046*.

Bontempi, G., S. Ben Taieb, et Y.-A. Le Borgne (2013), «Machine learning strategies for time series forecasting», in Aufaure, M.-A. et E. Zimányi, éditeurs, «Business Intelligence : Second European Summer School, eBISS 2012, Brussels, Belgium, July 15-21, 2012, Tutorial Lectures», pp. 62–77, Springer Berlin Heidelberg, doi :10.1007/978-3-642-36318-4_3.

Borgonovo, E. et E. Plischke (2016), «Sensivitity analysis : A review of recent advances», *European Journal of Operational Research*, vol. 248, pp. 869–887.

Borovykh, A., S. Bohte, et C. Oosterlee (2017), «Conditional time series forecasting with convolutional neural networks», *arXiv :1703.04691*.

Bowman, S. R., G. Angeli, C. Potts, et C. D. Manning (2015), «A large annotated corpus for learning natural language inference», in «Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing», pp. 632–642.

Brockwell, P. J. et R. A. Davis (2016), *Introduction to Time Series and Forecasting*, Springer Texts in Statistics, Springer New York Inc.

Buolamwini, J. et T. Gebru (2018), «Gender Shades : Intersectional Accuracy Disparities in Commercial Gender Classification», *Proceedings of Machine Learning Research*, vol. 81, pp. 1–15.

Burger, C., M. Dohnal, M. Kathrada, et R. Law (2001), «A practitioners guide to time-series methods for tourism demand forecasting—a case study of Durban, South Africa», *Tourism management*, vol. 22, nº 4, pp. 403–409.

Callot, L. (2019), «On the parametrization of simple autogressive models with neural networks», in «39th International Symposium on Forecasting (ISF)», Présentation orale.

Camara, A., W. Feixing, et L. Xiuqin (2016), «Energy consumption forecasting using seasonal arima with artificial neural networks models», *International Journal of Business and Management*, vol. 11, nº 5, p. 231.

Cao, Z., F. Wei, W. Li, et S. Li (2018), «Faithful to the Original : Fact Aware Neural Abstractive Summarization», in «Proceedings of the 32nd Conference on Artificial Intelligence (AAAI-18), New Orleans, LA, USA», pp. 4784–4791.

Castillo, E., B. Guijarro-Berdiñas, O. Fontenla-Romero, et A. Alonzo-Betanzos (2006), «A Very Fast Learning Method for Neural Networks based on Sensitivity Analysis», *Journal of Machine Learning Research*, vol. 7, pp. 1159–1182.

CE (2020), *Intelligence Artificielle. Une approche européenne axée sur l'excellence et la confiance*, Livre Blanc, Commission Européenne, https://op.europa.eu/publication.

Celeux, G., F. Forbes, et N. Peyrard (2003), «EM procedures using mean field-like approximations for Markov model-based image segmentation», *Pattern recognition*, vol. 36, pp. 131–144.

Centemeri, L. (2009), «Environmental Damage as Negative Externality : Uncertainty, Moral Complexity and the Limits of the Market», *e-cadernos CES*, vol. 5, pp. 21–40.

Chambon, S., M. N. Galtier, P. J. Arnal, G. Wainrib, et A. Gramfort (2018), «A deep learning architecture for temporal sleep stage classification using multivariate and multimodal time series», *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 26, nº 4, pp. 758–769.

Chatfield, C. (1978), «The Holt-Winters forecasting procedure», *Journal of the Royal Statistical Society : Series C (Applied Statistics)*, vol. 27, nº 3, pp. 264–279.

Chazal, F. et B. Michel (2017), «An Introduction to Topological Data Analysis : fundamental and practical aspects for data scientists», *arXiv :1710.04019*.

Chen, S. et H. Wang (2014), «SAR target recognition based on deep learning», in «2014 International Conference on Data Science and Advanced Analytics (DSAA)», pp. 541–547, doi :10.1109/DSAA.2014.7058124.

Cheng, J., L. Dong, et M. Lapata (2016), «Long Short-Term Memory-Networks for Machine Reading», in «Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing», pp. 551–561.

Choi, D., A. Passos, C. Shallue, et G. Dahl (2020), «Faster Neural Network Training with Data Echoing», *arXiv :1907.05550*.

Choudhary, P. (2017), «Introduction to Anomaly Detection», *Blog DataScience.com*, https://www.datascience.com/blog/python-anomaly-detection.

Cuturi, M. et M. Blondel (2017), «Soft-DTW : a differentiable loss function for time-series», in «Proceedings of the 34th International Conference on Machine Learning (ICML)», pp. 894–903.

Cybenko, G. (1989), «Approximation by superpositions of a sigmoidal function», *Mathematics of Control, Signals and Systems*, vol. 2, pp. 303–314.

Dalal, N. et B. Triggs (2005), «Histograms of Oriented Gradients for Human Detection», *Computer Vision Pattern Recognition*, vol. 1, pp. 886–893.

De Javel, F. (2019), «Melusine : un nouvelle brique oss by maif pour (ré)enchanter les emails», *Medium*.

Deb, C., F. Zhang, J. Yang, S. E. Lee, et K. W. Shah (2017), «A review on time series forecasting techniques for building energy consumption», *Renewable and Sustainable Energy Reviews*, vol. 74, pp. 902–924.

Del Barrio, E., P. Gordaliza, et J.-M. Loubes (2020), «Review of Mathematical frameworks for Fairness in Machine Learning», *arXiv :2005.13755*.

Delen, D. (2014), *Real-World Data Mining : Applied Business Analytics and Decision Making*, Pearson Education, Inc.

Denadai, E. (2018), «Interpretability of Deep Learning Models», *Towards Data Science*, https://towardsdatascience.com/interpretability-of-deep-learning-models-9f52e54d72ab.

Denis, C. et F. Varenne (2019), «Interprétabilité et explicabilité pour l'apprentissage machine : entre modèles descriptifs, modèles prédictifs et modèles causaux. une nécessaire clarification épistémologique», *Actes de la Conférence Nationale en Intelligence Artificielle - CNIA PFIA*, pp. 60–68.

Deverall, J., J. Lee, et M. Ayala (2017), «Using Generative Adversarial Networks to Design Shoes : The Preliminary Steps», Rapport technique, CS231-2017-119, Stanford University.

Devlin, J., M.-W. Chang, K. Lee, et K. Toutanova (2019), «BERT : Pre-training of deep bidirectional transformers for language understanding», in «Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies», vol. 1, pp. 4171–4186.

Dickey, D. (2005), «Stationarity issues in time series models», Rapport technique, North Carolina State University.

Dowlin, N., R. Gilad-Bachrach, K. Laine, K. Lauter, M. Naehrig, et J. Wernsing (2016), «Cryptonets : Applying neural networks to encrypted data with high throughput and accuracy», *Proceedings of Machine Learning Research*, vol. 48, pp. 201–210.

Dudek, G. (2015), «Short-term load forecasting using random forests», in «Intelligent Systems' 2014», pp. 821–828, Springer.

Elbir, A., H. O. İlhan, G. Serbes, et N. Aydın (2018), «Short Time Fourier Transform based music genre classification», in «2018 Electric Electronics, Computer Science, Biomedical Engineerings' Meeting (EBBT)», pp. 1–4, IEEE.

Emanet, N. (2009), «ECG beat classification by using discrete wavelet transform and random forest algorithm», in «Fifth International Conference on Soft Computing, Computing with Words and Perceptions in System Analysis, Decision and Control», pp. 1–4.

Espinasse, T., F. Gamboa, et J.-M. Loubes (2011), «Estimation error for blind Gaussian time series prediction», *Mathematical Methods of Statistics*, vol. 20, nº 206.

Esser, S. K., P. A. Merolla, J. V. Arthur, A. S. Cassidy, R. Appuswamy, A. Andreopoulos, D. J. Berg, J. L. McKinstry, T. Melano, D. R. Barch, C. di Nolfo, P. Datta, A. Amir, B. Taba, M. D. Flickner, et D. S. Modha (2016), «Convolutional networks for fast, energy-efficient neuromorphic computing», *Proceedings of the National Academy of Sciences*, vol. 113, nº 41, pp. 11441–11446.

Ezrachi, A. et M. Stucke (2017), «Artificial intelligence and collusion : when computers inhibit competition», *Oxford Legal Studies Research Paper No. 18/2015*.

Ezratty, O. (2018), «Les usages de l'intelligence artificielle», *Ebook*, https://www.oezratty.net.

Fan, A., S. Bhosale, H. Schwenk, Z. Ma, A. El-Kishky, S. Goyal, M. Baines, O. Celebi, G. Wenzek, V. Chaudhary, N. Goyal, T. Birch, V. Liptchinsky, S. Edunov, E. Grave, M. Auli, et A. Joulin (2020), «Beyond English-Centric Multilingual Machine Translation», *arXiv :2010.11125*.

Ferdousi, Z. et A. Maeda (2006), «Unsupervised outlier detection in time series data», in «22nd International Conference on Data Engineering Workshops (ICDEW'06)», pp. 51–56.

Fernández Hilario, A., S. Garcia, M. Galar, R. Prati, B. Kawczyk, et F. Herrera (2018), *Learning from Imbalanced Data Sets*, Springer Nature.

Féron, O. et A. Mohammad-Djafari (2005), «Image fusion and unsupervised joint segmentation using a HMM and MCMC algorithms», *Journal of Electronic Imaging*, vol. 14, nº 2, 023014.

Fildes, R. A., S. Ma, et S. Kolassa (2019), «Retail forecasting : Research and practice», *International Journal of Forecasting*, sous presse.

Forestier, G., F. Petitjean, H. A. Dau, G. I. Webb, et E. Keogh (2017), «Generating synthetic time series to augment sparse datasets», in «2017 IEEE International Conference on Data Mining (ICDM)», pp. 865–870.

Fu, R., Z. Zhang, et L. Li (2016), «Using LSTM and GRU neural network methods for traffic flow prediction», in «2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)», pp. 324–328.

Gasthaus, J., K. Benidis, Y. Wang, S. S. Rangapuram, D. Salinas, V. Flunkert, et T. Januschowski (2019), «Probabilistic Forecasting with Spline Quantile Function RNNs», *Proceedings of Machine Learning Research*, vol. 89, pp. 1901–1910.

Gatys, L., A. Ecker, et M. Bethge (2016), «Image style transfer using convolutional neural networks», *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2414–2423.

Genewein, T., T. McGrath, G. Delétang, V. Mikulik, M. Martic, S. Legg, et P. Ortega (2020), «Algorithms for Causal Reasoning in Probability Trees», *arXiv :2010.12237*.

Gers, F. A., J. Schmidhuber, et F. Cummins (1999), «Learning to forget : continual prediction with lstm», in «1999 Ninth International Conference on Artificial Neural Networks (ICANN)», vol. 2, pp. 850–855, doi :10.1049/cp:19991218.

Gevrey, M., I. Dimopoulos, et S. Lek (2003), «Review and comparison of methods to study the contribution of variables in artificial neural network models», *Ecological Modelling*, vol. 160, pp. 249–264.

Ghouzam, Y. et P. Valverde (2018), «Le deep learning pour accélérer le diagnostic par imagerie médicale», Note de blog : https://www.quantmetry.com/blog.

Glasmachers, T. (2017), «Limits of End-to-End Learning», *Proceedings of Machine Learning Research*, vol. 77, pp. 17–32.

Goodfellow, I., Y. Bengio, et A. Courville (2016), *Deep Learning*, MIT Press. Traduction française *L'apprentissage profond*, parue chez Quantmetry & Florent Massot, octobre 2018, http://www.deeplearningbook.org.

Gordaliza, P., E. Del Barrio, et J.-M. Loubes (2019), «Obtaining fairness using optimal transport theory», *Proceedings of Machine Learning Research*, vol. 97, pp. 2357–2365.

Gottschlag, M., P. Brantsch, et F. Bellosa (2020), «Automatic Core Specialization for AVX-512 Applications», in «Proceedings of the 13th ACM International Systems and Storage Conference», SYSTOR '20, p. 25–35, Association for Computing Machinery, New York, NY, USA, doi :10.1145/3383669.3398282.

Greenberg, H. J. et W. Pierskalla (1971), «A review of quasi-convex functions», *Operations Research*, vol. 19, pp. 1553–1570.

Gregorutti, B., B. Michel, et P. Saint-Pierre (2017), «Correlation and variable importance in random forests», *Statistics and Computing*, vol. 27, pp. 659–678.

Groves-Kirkby, C., A. Denman, R. Crockett, P. Phillips, et G. Gillmore (2006), «Identification of tidal and climatic influences within domestic radon time-series from Northamptonshire, UK», *Science of the Total Environment*, vol. 367, nº 1, pp. 191–202.

Guidotti, R., A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, , et D. Pedreschi (2018), «A survey of methods for explaining black box models», *ACM Computing Surveys*, vol. 51, nº 5, pp. 1–42.

Gundogdu, E., A. Koç, et A. A. Alatan (2016), «Object classification in infrared images using deep representations», in «Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP)», pp. 1066–1070, doi :10.1109/ICIP.2016.7532521.

Guo, T., Z. Xu, X. Yao, H. Chen, K. Aberer, et K. Funaya (2016), «Robust online time series prediction with recurrent neural networks», in «2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)», pp. 816–825.

Haji Ghassemi, N. et D. M. (2014), «Analytic long-term forecasting with periodic Gaussian processes», *Journal of Machine Learning Research W&CP*, URL http://proceedings.mlr.press/v33/hajighassemi14.pdf.

Hanin, N. (2019), «Universal function approximation by deep neural nets with bounded width and ReLu activations», *Mathematics*, vol. 7, nº 10.

Harchaoui, Z. et F. Bach (2007), «Image Classification with Segmentation Graph Kernels», in «2007 IEEE Conference on Computer Vision and Pattern Recognition», pp. 1–8, doi :10.1109/CVPR.2007.383049.

Hastie, T., R. Tibshirani, et J. Friedman (2001), *The Elements of Statistical Learning*, Springer Series in Statistics, Springer New York Inc.

Hatami, N., Y. Gavet, et J. Debayle (2018), «Classification of time-series images using deep convolutional neural networks», in Verikas, A., P. Radeva, D. Nikolaev, et J. Zhou, éditeurs, «Tenth International Conference on Machine Vision (ICMV 2017)», vol. 10696, pp. 242 – 249, International Society for Optics and Photonics, SPIE, doi :10.1117/12.2309486.

He, K., G. Gkioxari, P. Dollár, et R. Girshick (2017), «Mask R-CNN», in «Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)», pp. 2980–2988, doi :10.1109/ICCV.2017.322.

He, K., X. Zhang, S. Ren, et J. Sun (2016), «Deep Residual Learning for Image Recognition», in «Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)», pp. 770–778, doi :10.1109/CVPR.2016.90.

He, L., X. Ren, Q. Gao, X. Zhao, B. Yao, et Y. Chao (2018), «The connected-component labeling problem : A review of state-of-the-art algorithms», *Pattern recognition*, vol. 70, pp. 25–43.

Heaven, W. (2020), «Google's medical AI was super accurate in a lab. Real life was a different story», *MIT Technology Review*, 27 avril.

Helfenstein, U. (1986), «Box-Jenkins modelling of some viral infectious diseases», *Statistics in Medicine*, vol. 5, nº 1, pp. 37–47.

Hénin, C. et D. Le Métayer (2019a), «Accountability requirements for algorithmic decision systems», in «Proceedings of the Interdisciplinary Workshop SRA 2019 (Social Responsability of Algorithms), Paris», .

Hénin, C. et D. Le Métayer (2019b), «Towards a generic framework for black-box explanation methods», in «Proceedings of the IJCAI 2019 Workshop on Explainable Artificial Intelligence (XAI), Macao, Chine», .

Hensman, J., N. Fusi, et N. D. Lawrence (2013), «Gaussian Processes for Big Data», in «Proceedings of the 29th Conference on Uncertainty in Artificial Intelligence», UAI'13, p. 282–290, AUAI Press, Arlington, Virginia, USA.

Hickok, M. (2021), «Lessons learned from AI ethics principles for future actions», *AI and Ethics*, vol. 1, pp. 41–47.

Hochard, G. et L. Blanche (2019), «Statistical and machine learning methods combination for improved energy consumption forecasting performance», in «39th International Symposium on Forecasting (ISF)», Présentation orale.

Hochreiter, S. et J. Schmidhuber (1997), «Long Short-Term Memory», *Neural Computation*, vol. 9, nº 8, pp. 1735–1780.

Hornik, K. (1991), «Approximation capabilities of multilayer feedforward networks», *Neural Networks*, vol. 4, nº 2, pp. 251–257.

Howar, F. et J. Barnat (éditeurs) (2018), *Formal Methods for Industrial Critical Systems (FMICS Proceedings)*, Springer.

Howard, J. et S. Ruder (2018), «Universal language model fine-tuning for text classification», in «Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics», vol. 1, pp. 328–339, Association for Computational Linguistics, Melbourne, Australia.

Huang, C., P. Kairouz, et L. Sankar (2018), «Generative Adversarial Privacy : A Data-Driven Approach to Information-Theoretic Privacy», in «Proceedings of the 2018 52nd Asilomar Conference on Signals, Systems, and Computers», pp. 2162–2166, doi :10.1109/ACSSC.2018.8645532.

Huang, H., P. Yu, et C. Wang (2018), «An introduction to image synthesis with generative adversarial nets», *arXiv :1803.04469*.

Innes, M., S. Karpinski, V. Sha, J. Bradbury, D. Barber, P. Stenetorp, V. Churavy, S. Danisch, A. Edelman, J. Malmaud, J. Revels, T. Besard, et D. Yuret (2018), «On Machine Learning and Programming Languages», in «Proceedings of SysML 2018, Stanford CA», ACM, New York, USA.

Jahnke, P. (2015), «Machine learning approaches for failure type detection and predictive maintenance», *Thèse de doctorat, Technische Universität Darmstadt*.

Janon, A., T. Klein, A. Lagnoux, M. Nodet, et C. Prieur (2012), «Asymptotic normality and efficiency of two Sobol index estimators», *ESAIM : Probability and Statistics*, vol. 18, pp. 342–364.

Jayadevan, R., S. Kolhe, P. Patil, et U. Pal (2011), «Automatic processing of handwritten bank cheque images : A survey», *Document Analysis and Recognition*, vol. 15, pp. 1–30.

Jones, D. et M. Lorenz (1986), «An application of a Markov chain noise model to wind generator simulation», *Mathematics and Computers in Simulation*, vol. 28, nº 5, pp. 391–402.

Jones, N. (2018), «How to stop data centres from globbing up the world's electricity», *Nature*, vol. 561, pp. 163–166.

Katzman, J., U. Shaham, A. Cloninger, J. Bates, J. Tingting, et Y. Kluger (2018), «DeepSurv : Personalized Treatment Recommender System Using a Cox Proportional Hazards Deep Neural Network», *BMC Medical Research Methodology*, vol. 18, nº 24.

Kegel, L., M. Hahmann, et W. Lehner (2018), «Feature-based comparison and generation of time series», in «Proceedings of the 30th International Conference on Scientific and Statistical Database Management», 20, pp. 1–12.

Keogh, E. J. et M. J. Pazzani (2000), «A simple dimensionality reduction technique for fast similarity search in large time series databases», in «Pacific-Asia conference on knowledge discovery and data mining», pp. 122–133, Springer.

Kidger, P. et T. Lyons (2020), «Universal Approximation with Deep Narrow Networks», in Abernethy, J. et S. Agarwal, éditeurs, «Proceedings of 33rd Conference on Learning Theory», *Proceedings of Machine Learning Research*, vol. 125, pp. 2306–2327, PMLR.

Kim, Y. J., S. Choi, S. Briceno, et D. Mavris (2016), «A deep learning approach to flight delay prediction», in «2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)», pp. 1–6, doi :10.1109/DASC.2016.7778092.

Kleijnen, J. (2007), «Kriging Metamodeling in Simulation : A Review», *European Journal of Operational Research*, vol. 192, pp. 707–716.

Krishnan, M. (2019), «Against Interpretability : a Critical Examination of the Interpretability Problem in Machine Learning», *Philosophy and Technology*, vol. 33, pp. 487–502.

Kusdarwati, H. et S. Handoyo (2018), «System for Prediction of Non Stationary Time Series based on the Wavelet Radial Bases Function Neural Network Model», *International Journal of Electrical and Computer Engineering*, vol. 8, nº 4, p. 2327.

L Griffiths, T. et M. Steyvers (2004), «Finding Scientific Topics», in «Proceedings of the US National Academy of Sciences», vol. 101, pp. 5228–5235.

Längkvist, M., L. Karlsson, et A. Loutfi (2014), «A review of unsupervised feature learning and deep learning for time-series modeling», *Pattern Recognition Letters*, vol. 42, pp. 11–24.

Laugel, T., M.-J. Lesot, C. Marsala, X. Renard, et M. Detyniecki (2019), «The Dangers of Post-hoc Interpretability : Unjustified Counterfactual Explanations», in «Proceedings of the 28th International Joint Conference on Artificial Intelligence», pp. 2801–2807.

Le Cun, Y. (2019), *Quand la machine apprend. La révolution des neurones artificiels et de l'apprentissage profond*, Odile Jacob.

Le Cun, Y. et Y. Bengio (1995), «Convolutional networks for images, speech, and time series», *The Handbook of Brain Theory and Neural Networks*, vol. 3361, nº 10, p. 1995.

Lesaffre, G. (2018), «La réalité augmentée au musée : une révolution du regard», *L'Hebdo du Quotidien de l'Art*, vol. 1423, pp. 22–23.

Li, S., J. Lin, G. Li, T. Bai, H. Wang, et Y. Pang (2018), «Vehicle type detection based on deep learning in traffic scene», *Procedia Computer Science*, vol. 131, pp. 564–572.

Li, T., V. Gupta, M. Mehta, et V. Srikumar (2019), «A Logic-Driven Framework for Consistency of Neural Models», in «Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)», pp. 3924–3935.

Lipton, Z. et J. Steinhardt (2019), «Troubling trends in machine learning scholarship», *ACM Queue*, vol. 17, nº 1, pp. 1–33.

Lipton, Z. C. (2018), «The mythos of model interpretability : In machine learning, the concept of interpretability is both important and slippery.», *ACM Queue*, vol. 16, nº 3, p. 31–57.

Litjens, G., T. Kooi, B. Bejnordi, A. Adiyoso Seto, F. Ciompi, M. Ghafoorian, J. van der Laak, B. van Ginneken, et C. Sanchez (2017), «A survey on deep learning in medical image analysis», *Medical Image Analysis*, vol. 42, pp. 60–88.

Liu, C., Y. Cao, Y. Luo, G. Chen, V. Vokkarane, M. Yunsheng, S. Chen, et P. Hou (2018), «A new deep learning-based food recognition system for dietary assessment on an edge computing service infrastructure», *IEEE Transactions on Services Computing*, vol. 11, pp. 249–261.

Logeswaran, L. et H. Lee (2018), «An efficient framework for learning sentence representations», in «Proceedings of the 6th International Conference on Learning Representations, ICLR», pp. 1–16.

Lv, F., C. Wen, M. Liu, et Z. Bao (2017), «Weighted time series fault diagnosis based on a stacked sparse autoencoder», *Journal of Chemometrics*, vol. 31, nº 9, p. e2912.

Ma, J. et S. Perkins (2003), «Time-series novelty detection using one-class support vector machines», in «Proceedings of the International Joint Conference on Neural Networks», vol. 3, pp. 1741–1745.

MacDonald, I. L. et W. Zucchini (1997), *Hidden Markov and other models for discrete-valued time series*, CRC Press.

Makridakis, S., R. M. Hogarth, et A. Gaba (2009), «Forecasting and uncertainty in the economic and business world», *International Journal of Forecasting*, vol. 25, nº 4, pp. 794–812.

Makridakis, S., E. Spiliotis, et V. Assimakopoulos (2018), «The M4 Competition : Results, findings, conclusion and way forward», *International Journal of Forecasting*, vol. 34, nº 4, pp. 802–808.

Manning, C. D. (2015), «Computational Linguistics and Deep Learning», *Computational Linguistics*, vol. 41, nº 4, pp. 701–707, doi :10.1162/COLI_a_00239.

Marelli, M., S. Menini, M. Baroni, L. Bentivogli, R. Bernardi, et R. Zamparelli (2014), «A SICK cure for the evaluation of compositional distributional semantic models», in «Proceedings of the Ninth International Conference on Language Resources and Evaluation LREC», pp. 216–223.

Mariet, Z. et V. Kuznetsov (2019), «Foundations of Sequence-to-Sequence Modeling for Time Series», *Proceedings of Machine Learning Research*, vol. 89, pp. 408–417.

Martí, L., N. Sanchez-Pi, J. Molina, et A. Garcia (2015), «Anomaly detection based on sensor data in petroleum industry applications», *Sensors*, vol. 15, nº 2, pp. 2774–2797.

Martin, L., B. Muller, P. Ortiz Suárez, Y. Dupont, L. Romary, E. Villemonte de la Clergerie, et B. Sagot (2019), «CamemBERT : a Tasty French Language Model», in «Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)», pp. 7203–7219.

Mastelic, T., A. Oleksiak, H. Claussen, I. Brandic, J.-M. Pierson, et A. Vasilakos (2014), «Cloud computing : survey on energy efficiency», *ACM Computing Surveys*, vol. 47, pp. 1–36.

Maturana, D. et S. Scherer (2015), «VoxNet : A 3D convolutional neural network for real-time object recognition», in «2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)», pp. 922–928, doi :10.1109/IROS.2015.7353481.

Merity, S., N. S. Keskar, et R. Socher (2018), «An Analysis of Neural Language Modeling at Multiple Scales», *arXiv :1803.08240*.

Mikolov, T., K. Chen, G. Corrado, et J. Dean (2013), «Efficient Estimation of Word Representations in Vector Space», in «1st International Conference on Learning Representations ICLR», Workshop Track Proceedings.

Mnasri, M. (2019), «Recent advances in conversational NLP : Towards the standardization of chatbot building», *arXiv :1903.09025*.

Molchanov, P., S. Tyree, T. Karras, T. Aila, et J. Kautz (2017), «Pruning convolutional neural networks for resource efficient inference», in «Proceedings of the International Conference of Learning Representations (ICLR)», pp. 1–17.

Molnar, C., G. Casalicchio, et B. Bischl (2020), «Interpretable machine learning – a brief history, state-of-the-art and challenges», in Koprinska, I., M. Kamp, A. Appice, C. Loglisci, L. Antonie, A. Zimmermann, R. Guidotti, Ö. Özgöbek, R. P. Ribeiro, R. Gavaldà, J. Gama, L. Adilova, Y. Krishnamurthy, P. M. Ferreira, D. Malerba, I. Medeiros, M. Ceci, G. Manco, E. Masciari,

Z. W. Ras, P. Christen, E. Ntoutsi, E. Schubert, A. Zimek, A. Monreale, P. Biecek, S. Rinzivillo, B. Kille, A. Lommatzsch, et J. A. Gulla, éditeurs, «ECML PKDD 2020 Workshops», pp. 417–431, Springer International Publishing, Cham.

Mosavi, A., M. Salimi, S. Ardabili, T. Rabczuk, S. Shamshirband, et A. Varkoyi-Koczy (2019), «State of the Art of Machine Learning Models in Energy Systems, a Systematic Review», *Energies*, vol. 12, p. 1301.

Mothilal, R. K., A. Sharma, et C. Tan (2020), «Explaining machine learning classifiers through diverse counterfactual explanations», *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pp. 607–617.

Navarro, C., R. Carrasco, R. Barrientos, J. Riquelme, et R. Vega (2020), «GPU Tensor Cores for fast Arithmetic Reductions», *arXiv :2001.05585v1*.

Ning, L., H. Guan, et X. Shen (2019), «Adaptive Deep Reuse : Accelerating CNN Training on the Fly», in «Proceedings of the IEEE International Conference on Data Engineering (ICDE)», pp. 1538–1549.

Oh, S., Y. Jung, S. Kim, I. Lee, et N. Kang (2019), «Deep Generative Design : Integration of Topology Optimization and Generative Models», *arXiv :1903.01548*.

Olah, C., A. Satyanarayan, I. Johnson, S. Carter, L. Schubert, K. Ye, et A. Mordvintsev (2018), «The Building Blocks of Interpretability», *Distill*, vol. 3, nº 3.

Ovenden, J. (2018), «Obstacles to machine learning adoption», Note de blog : https://channels.theinnovationenterprise.com.

Pégny, M. et I. Ibnouhsein (2018), «Quelle transparence pour les algorithmes d'apprentissage machine ?», *Revue d'Intelligence Artificielle*, vol. 4, pp. 447–478.

Pégny, M., E. Thelisson, et I. Ibnouhsein (2019), «The Right to an Explanation», *Delphi*, vol. 4, pp. 161–166.

Pereyra, M. et S. McLaughlin (2015), «Fast Uunsupervised Bayesian Image Segmentation with Adaptive Spatial Regularisation», *IEEE Transactions on Image Processing*, vol. 6, pp. 2577–2587.

Perez, C. (2018), «Why Deep Learning needs standards for Industrialization», *Medium*, 9 février.

Peters, M. E., M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, et L. Zettlemoyer (2018), «Deep Contextualized Word Representations», in «Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics (ACL) : Human Language Technologies», vol. 1, pp. 2227–2237.

Powers, D. (2011), «Evaluation : from precision, recall and F-measure to ROC, informedness, markedness and correlation», *International Journal of Machine Learning Technology*, vol. 2, pp. 37–63.

Pégny, M. et M. Ibnouhsein (2018), «Quelle transparence pour les algorithmes d'apprentissage machine ?», *Revue d'intelligence artificielle*, vol. 32, pp. 447–478.

Qian, B., Y. Xiao, Z. Zheng, M. Zhou, S. Zhuang, W.and Li, et Q. Ma (2020), «Dynamic multi-scale convolutional neural networks for time series classification», *IEEE Access*, vol. 8, pp. 109732–109746.

Radford, A., J. Wu, R. Child, D. Luan, D. Amodei, et I. Sutskever (2019), «Language models are unsupervised multitask learners», Rapport technique 14, OpenAI.

Rakshit, S., S. Debnath, et D. Mondal (2018), «Identifying land patterns from satellite imagery in Amazon rainforest using deep learning», *arXiv :1809.00340v1*.

Redd, A., K. Khin, et A. Marini (2019), «Fast ES-RNN : A GPU Implementation of the ES-RNN Algorithm», *arXiv :1907.03329*.

Redmon, J. et A. Farhadi (2017), «YOLO9000 : Better, Faster, Stronger», in «2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)», pp. 6517–6525, doi : 10.1109/CVPR.2017.690.

Remus, W. et M. O'Connor (2001), *Neural Networks for Time-Series Forecasting*, pp. 245–256, Springer US, Boston, MA.

Ren, S., K. He, R. Girshick, et J. Sun (2015), «Faster R-CNN : Towards real-time object detection with region proposal networks», *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, pp. 1137–1149.

Robert, C. et G. Casella (2004), *Monte Carlo Statistical Methods, 2nd edition*, Springer.

Robertson, S. (2004), «Understanding Inverse Document Frequency : On Theoretical Arguments for IDF», *Journal of Documentation*, vol. 60, pp. 503–520.

Rudin, C. (2019), «Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead», *Nature Machine Intelligence*, vol. 1, p. 206–215.

Salinas, D., V. Flunkert, J. Gasthaus, et T. Januschowski (2019), «Deepar : Probabilistic forecasting with autoregressive recurrent networks», *International Journal of Forecasting*, vol. 36, pp. 1181–1191.

Schäfer, P. et U. Leser (2020), «TEASER : Early and Accurate Time Series Classification», *Data Mining and Knowledge Discovery*, vol. 34, pp. 1336–1362.

Schmidt, F. et J. Hunter (2015), *Methods of Meta-Analysis, Third Edition : Correcting Error and Bias in Research Findings*, SAGE Publications.

Schörgenhumer, A., M. Kahlhofer, P. Chalupar, P. Grünbacher, et H. Mössenböck (2019), «A Framework for Preprocessing Multivariate, Topology-Aware Time Series and Event Data in a Multi-System Environment», in «2019 IEEE 19th International Symposium on High Assurance Systems Engineering (HASE)», pp. 115–122.

Schuman, C., T. Potok, R. Patton, J. Douglas Birdwell, M. Dean, G. Rose, et J. Plank (2017a), «A Survey of Neuromorphic Computing and Neural Networks in Hardware», *arXiv :1705.06963v1*.

Schuman, C., T. Potok, R. Patton, J. Douglas Birdwell, M. Dean, G. Rose, et J. Plank (2017b), *Your Artificial Intelligence, Connectionism or Augmented Fuzzy Cognitivism ?*, Intellitech, Compiègne.

Seeger, M. W., D. Salinas, et V. Flunkert (2016), «Bayesian intermittent demand forecasting for large inventories», in «Advances in Neural Information Processing Systems (NIPS Proceedings)», pp. 4646–4654.

Seyedhossein, L. et M. R. Hashemi (2010), «Mining information from credit card time series for timelier fraud detection», in «Proceedings of the 5th International Symposium on Telecommunications», pp. 619–624.

Sezer, O. B., M. U. Gudelek, et A. M. Ozbayoglu (2020), «Financial time series forecasting with deep learning : A systematic literature review : 2005–2019», *Applied Soft Computing*, vol. 90, nº 106181, ISSN 1568-4946, doi :https://doi.org/10.1016/j.asoc.2020.106181.

Shapiro, L. et G. Stockman (2003), *Computer Vision*, Prentice Hall.

Sharif Razavian, A., H. Azizpour, J. Sullivan, et S. Carlsson (2014), «CNN features off-the-shelf : An astounding baseline for recognition», in «Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (ICVPR)», pp. 512–519.

Shipmon, D. T., J. M. Gurevitch, P. M. Piselli, et S. T. Edwards (2017), «Time series anomaly detection ; detection of anomalous drops with limited features and sparse examples in noisy highly periodic data», *arXiv :1708.03665*.

Shmueli, G. et O. Koppius (2009), «The Challenge of Prediction in Information Systems Research», *Robert H. Smith School Research Paper No. RHS 06-152*.

Shumway, R. H. et D. S. Stoffer (2017), *Time series analysis and its applications, with R examples*, Springer.

Smyl, S., J. Ranganathan, et A. Pasqua (2018), «M4 forecasting competition : Introducing a new hybrid ES-RNN model», Uber Engineering Blog : https://eng.uber.com/m4-forecasting-competition.

Spille, C., S. Ewert, B. Kollmeier, et B. Meyer (2018), «Predicting Speech Intelligibility with Deep Neural Networks», *Computer Speech and Language*, vol. 48, pp. 51–66.

Strubell, E., A. Ganesh, et A. McCallum (2020), «Energy and Policy Considerations for Modern Deep Learning Research», *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, nº 09, pp. 13693–13696, doi :10.1609/aaai.v34i09.7123.

Sun, W., S. Shao, R. Zhao, R. Yan, X. Zhang, et X. Chen (2016), «A sparse auto-encoder-based deep neural network approach for induction motor faults classification», *Measurement*, vol. 89, pp. 171–178.

Sutton, C. et A. McCallum (2007), «An Introduction to Conditional Random Fields for Relational Learning», *Foundations and Trends in Machine Learning*, vol. 4, nº 4, pp. 267–373.

Szegedy, C., Wei Liu, Yangqing Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, et A. Rabinovich (2015), «Going deeper with convolutions», in «2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)», pp. 1–9, doi :10.1109/CVPR.2015.7298594.

Szepesvári, C. (2010), *Algorithms for Reinforcement Learning*, Morgan and Claypool Publishers.

Taieb, S. B. et R. J. Hyndman (2014), «A gradient boosting approach to the Kaggle load forecasting competition», *International journal of Forecasting*, vol. 30, nº 2, pp. 382–394.

Taieb, S. B., R. J. Hyndman, et al. (2012), *Recursive and direct multi-step forecasting : the best of both worlds*, Working Paper, Monash University.

Talamo, C., G. Paganin, et F. Rota (2019), «Industry 4.0 for failure information management within Proactive Maintenance», in «IOP Conference Series : Earth and Environmental Science», vol. 296, IOP Publishing.

Terrade, F., H. Pasquier, J. Reerinck-Boulanger, G. Guingouain, et A. Somat (2009), «L'acceptabilité sociale : la prise en compte des déterminants sociaux dans l'analyse de l'acceptabilité des systèmes technologiques», *Presses Universitaires de France. Le Travail Humain*, vol. 72, pp. 383–395.

Tulio Ribeiro, M., S. Singh, et C. Guestrin (2016), «"Why Should I Trust You ?" : Explaining the Predictions of Any Classifier», in «Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining», p. 1135–1144.

Valenzise, G., A. Purica, V. Hulusic, et M. Cagnazzo (2018), «Quality assessment of deep-learning-based image compression», in «Proceedings of the IEEE 20th International Workshop on Multimedia Signal Processing (MMSP)», pp. 1–6, doi :10.1109/MMSP.2018.8547064.

Van Steenkiste, T., J. van der Herten, I. Couckuyt, et T. Dhaene (2019), «Data-efficient sensitivity analysis with surrogate modeling», in Canavero, F., éditeur, «Uncertainty Modeling for Engineering Applications», pp. 55–69, Springer International Publishing, Cham, doi : 10.1007/978-3-030-04870-9_4.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, et I. Polosukhin (2017), «Attention is All You Need», in «Advances in Neural Information Processing Systems (NIPS Proceedings)», vol. 30.

Vidal, J. (2017), «"Tsunami of data" could consume one fifth of global electricity by 2025», *Climate Home News, The Guardian*, 11 décembre.

Villani, C. (2016), «La langue de chez nous», *Images des Mathématiques, Éditions du CNRS*, 10 février.

Viola, P. et M. Jones (2001), «Rapid object detection using a boosted cascade of simple features», in «Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001», vol. 1, pp. I–I, doi :10.1109/CVPR.2001.990517.

von Ahn, L., B. Maurer, C. McCillen, D. Abraham, et M. Blum (2008), «reCAPTCHA : Human-Based Character Recognition via Web Security Measures», *Science*, vol. 321, pp. 1465–1468.

Wang, A., A. Singh, J. Michael, F. Hill, O. Levy, et S. R. Bowman (2019), «GLUE : A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding», in «7th International Conference on Learning Representations, ICLR», pp. 353–355.

Wang, J., Y. Ma, L. Zhang, R. Gao, et D. Wu (2018), «Deep learning for smart manufacturing : Methods and applications», *Journal of Manufacturing Systems*, vol. 48, pp. 144–156.

Wei, D., B. Zhou, A. Torrabla, et W. Freeman (2015), «Understanding Intra-Class Knowledge Inside CNN», *arXiv :1507.02379*.

Weld, D. et G. Bangal (2019), «The Challenge of Crafting Intelligible Intelligence», *Communications of the ACM*, vol. 62, nº 6, pp. 70–79.

Wiggins, A., N. Greg, R. Stevenson, et K. Crowston (2011), «Mechanism for Data Quality and Validation in Citizen Science», in «Proceedings of the IEEE 7th Conference on E-Science Workshops», pp. 14–19.

Wismüller, A., D. R. Dersch, B. Lipinski, K. Hahn, et D. Auer (1998), «neural network approach to functional MRI pattern analysis—clustering of time-series by hierarchical vector quantization», in «Proceedings of the International Conference on Artificial Neural Networks (ICANN)», pp. 857–862, Springer London, London.

Wisskirchen, G., B. Thibault-Biacabe, U. Bormann, A. Muntz, G. Niehaus, G. Soler, et B. von Brauchitsch (2017), «Artificial intelligence and robotics and their impact on the workplace», Rapport technique, IBA Global Employment Institute.

Woodward, W. A., H. L. Gray, et A. C. Elliott (2017), *Applied time series analysis with R*, CRC press.

Xie, C., A. Talk, et E. Fox (2016), «A Unified Framework for Missing Data and Cold Start Prediction for Time Series Data», in «NIPS Time Series Workshop», pp. 1–12.

Yang, Z., Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, et Q. V. Le (2019), «XLNet : Generalized Autoregressive Pretraining for Language Understanding», in Wallach, H., H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, et R. Garnett, éditeurs, «Advances in Neural Information Processing Systems», vol. 32, Curran Associates, Inc., Red Hook, NY, USA.

Yao, S., S. Hu, Y. Zhao, A. Zhang, et T. Abdelzaher (2017), «Deepsense : A unified deep learning framework for time-series mobile sensing data processing», in «Proceedings of the 26th International Conference on World Wide Web», p. 351–360.

Zakaria, J., A. Mueen, et E. Keogh (2012), «Clustering time series using unsupervised-shapelets», in «Proceedings of the IEEE 12th International Conference on Data Mining», p. 785–794.

Zhang, P. et L. Xu (2018), «Unsupervised Segmentation of Greenhouse Plant Iimages based on Statistical Method», *Scientific Reports*, vol. 8, nº 4465.

Zhang, Q. et S.-C. Zhu (2019), «Visual Interpretability for Deep Learning : A Survey», *Frontiers of Information Technology and Electronic Engineering*, vol. 19, pp. 27–39.

Zhang, R., P. Isola, et A. A. Efros (2016), «Colorful image colorization», in «Proceedings of the European Conference on Computer Vision (ECCV)», pp. 649–666, Springer International Publishing, Cham.

Zhang, X.-F. (2017), «Natural Language Process (NLP) will transform traditional B2B Market Research Industry», *Medium*, 28 juillet.

Zhao, A., S. Hu, et C. Yu (2017), «Classifying Weather, Terrain, and Deforestation of the Amazon using Deep Multi-task Convolutional Neural Nets», *Stanford University Research Report*.

A. Lexique

Apprentissage par transfert. (*transfer learning*) L'apprentissage par transfert caractérise la capacité d'un système apprenant (tel un algorithme de reconnaissance de formes par RNN, par exemple), construit à partir de sources d'information historiques (ex. : données d'entraînement), à s'adapter à des situations nouvelles partageant des similitudes avec cette information historique. Une technique usuelle d'apprentissage par transfert dans les RNN est de conserver certaines couches cachées lorsqu'on bascule d'un jeu de données historiques à un nouveau jeu de données d'entraînement ; ces couches cachées étant « spécialisées » dans la détection de certains types de structure dans les données.

Auto-encodeur. Un auto-encodeur est un type particulier de réseau de neurones utilisé en apprentissage non supervisé pour apprendre les caractéristiques d'un jeu de données, en proposant une représentation de l'information qu'il porte en faible dimension. Il produit une compression de cette information.

Cartes de caractéristiques. (*feature / activation maps*) Dans un réseau de neurones multicouches, une carte de caractéristiques par couche trace la distribution des sorties des fonctions d'activation des neurones de la couche ; elle fournit donc un résumé des caractéristiques des données sélectionnées par ces fonctions, et permet de percevoir comment l'information est filtrée par les neurones de la couche (par exemple, dans un RNC traitant des images, certaines cartes illustrent le fait qu'une couche a préférence à sélectionner des structures géométriques particulières. Les cartes de caractéristiques sont donc des outils d'intelligibilité des réseaux de neurones et permettent en particulier de sélectionner des couches utiles à conserver en cas d'apprentissage par transfert.

Compilateur. Un compilateur est un programme qui transforme un code source en un code dans un langage plus directement compréhensible par une machine ; il est donc un élément essentiel de l'interopérabilité des plateformes logicielles.

Cycle d'horloge. Un cycle d'horloge est une unité de temps élémentaire d'un ordinateur, imposé par la nature du processeur.

Élagage. (*pruning*) L'élagage d'un modèle est la réduction de sa complexité en vue d'atteindre un compromis entre précision et coût de calcul. Un arbre de décision sera élagué par la réduction de sa profondeur, tandis qu'un réseau de neurones sera élagué en « tuant » certaines connexions entre des neurones de différentes couches, voire en « éteignant » certains neurones inutiles (Figure 9).

Fléau de la dimension. (*curse of dimensionality*) Le fléau de la dimension est un terme générique caractérisant la difficulté de manipulation de modèles et algorithmes (d'estimation notamment) dans des espaces de grande dimension. Dans de tels espaces, les données deviennent rapidement isolées, et l'information éparse, ce qui entraîne des coûts de calcul importants, voire prohibitifs. Les techniques d'inférence bayésienne et de réduction de dimension permettent de combattre ce phénomène.

Loyauté. (*fairness*) La loyauté d'un algorithme d'apprentissage recouvre à ce jour plusieurs caractéristiques de conception et d'implantation sur une plate-forme logicielle de cet algorithme. Elle traduit notamment le fait que l'échantillon d'apprentissage n'incorpore pas de biais susceptible d'orienter les résultats de cet algorithme et d'aboutir à une discrimination, et que la règle d'apprentissage n'est pas elle-même biaisée par construction. Cette notion a été formalisée différemment par de nombreux auteurs, et témoigne de la difficulté à s'accorder sur des métriques sous-jacentes à la définition des dits biais.

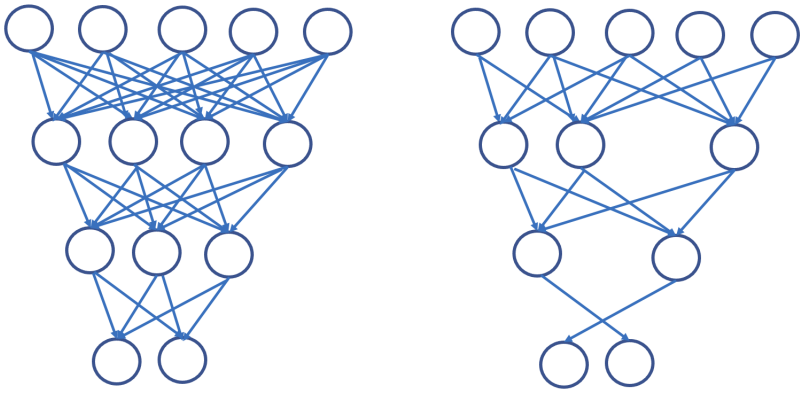


FIGURE 9 – *Exemple d'élagage d'un réseau de neurones. Un réseau entraîné (à gauche) est densément connecté, mais certaines connections (synapses) et certains neurones peuvent ne jouer qu'un rôle redondant et non significatif dans la prévision en sortie de réseau. Ils peuvent être ôtés, ce qui allège l'architecture du réseau élagué (à droite).*

Elle joue un rôle central dans la définition de la responsabilité sociétale (*accountability*) des algorithmes (Hénin et Le Métayer, 2019a). La preuve de loyauté constitue l'un des problèmes les plus épineux de la certification des algorithmes d'apprentissage en vue de leur utilisation à large échelle. Le lecteur intéressé pourra notamment consulter les références Pégny et Ibnouhsein (2018), Besse et al. (2019), Hickok (2021), Del Barrio et al. (2020).

Mécanisme d'attention. Dans un cadre temporel où les données sont des vecteurs (ex. : lecture d'une série temporelle ou d'un énoncé), un mécanisme d'attention dans un réseau de neurones est une méthode d'aggrégation d'états (soit les données vectorielles en entrée, soit les données des pas de temps précédents placées dans une mémoire, soit des états de couches cachées dans le réseau). Cette aggrégation produit un vecteur unique résumant le *contexte* du pas de temps actuel. Celui-ci est introduit dans le réseau de neurone pour aider à la prévision.

Segmentation. La segmentation d'un ensemble de données $\mathbb{x}$ est la localisation de structures géométriques particulières au sein de ces données, et la mise en évidence de sous-populations. Ainsi, la segmentation d'une IRM de cerveau est la localisation des tissus du cerveau (matière grise, substance blanche, etc.). La segmentation de données clients consiste à les regrouper en *clusters* dans le but (en général) de différencier des politiques commerciales. La segmentation peut être non supervisée, et permet dans ce cas de proposer un label aux sous-populations mises en lumière.

Sur-apprentissage. (*overfitting*) Le sur-apprentissage, ou sur-ajustement, caractérise un apprentissage « trop proche » des données et permettant mal (ou pas du tout) de généraliser le comportement appris à des situations non encore rencontrées. Typiquement, ce phénomène apparaît lorsqu'un modèle d'apprentissage est sur-paramétré et non identifiable : de multiples (voire une infinité de) combinaisons de paramètres peuvent aboutir à une même performance optimale sur les données d'apprentissage.

Système critique. Les systèmes dits *critiques* sont généralement des systèmes industriels dont l'usage est fortement contraint par des normes et susceptible d'être examiné par une autorité indépendante (Howar et Barnat , éditeurs) – celle-ci ayant notamment pouvoir de sanctionner une utilisation fondée sur des approches automatisées ne respectant

pas certains pré-requis. Les systèmes de (cyber)sécurité, les systèmes autonomes, de recommandation, de supervision et de planification, les systèmes d'optimisation de l'ingénierie, des processus et des services en constituent des exemples-phares dans le domaine de l'IA (CE, 2020). En France, la CNIL surveille la réalité de l'anonymisation des données personnelles utilisées dans de nombreux outils de gestion de la clientèle ; l'Autorité des Marchés Financiers (AMF) encadre strictement l'emploi d'algorithmes d'optimisation financière en vérifiant leur incapacité à manipuler les cours (qui n'est pas forcément garantie, actuellement, pour des outils fondés sur de l'AP).

Transformer. Un *transformer* est un modèle d'AP fondé sur des architectures emboîtées d'encodage et de décodage, incluant des mécanismes d'attention et des réseaux. Conçu pour traiter des données séquentielles (telles des données textuelles), il n'a cependant pas besoin de parcourir une séquence dans l'ordre pour être efficace, et offre donc de fortes possibilités de parallélisation, de réduction de temps d'entraînement par rapport à un LSTM et donc d'ingestion de très grandes quantités de données. Pour cette raison, les *transformers* sont rapidement devenus les outils pré-entraînés les plus puissants à ce jour pour traiter les données textuelles.

B. Vision récapitulative de quelques cas d'usage

Cas d'usage	Type de données	Modèle / algorithme principal	Autres outils testés ou envisageables	Contraintes et difficultés de mise en œuvre
Détections de toitures	Une dizaine de Giga-octets d'images aériennes	R-RNC (*selective search* et RNC)	Fast-RCNN, Faster-RRNC, YOLO, SSD	Hétérogénéité de la qualité des images (contraste et teintes différentes)
Détections de follicules ovariens	Une quizaine de Tera-octets d'images microscopiques	RNC et *windowing*	R-CNN ou *convolutional windowing*	Volumes de données et types de données médicales La taille de la cible par rapport à la taille de l'image
Classification de photographies de produits en fin de chaîne industrielle	Une dizaine de Giga-octets d'images pour l'apprentissage puis un flux vidéo	Forêt aléatoire	GBM	Le pilote et le modèle prédictif doivent ensuite tourner sur un robot dédié
Labellisation multi-classes d'images satellites	Plusieurs dizaines de Giga-octets d'images satellites	ResNet	Inception V3, Forêt aléatoire après construction de *features* : histogramme des couleurs, détection des contours , HOG	Temps de réalisation très court
Recommandation de tableaux de style similaire	Plusieurs dizaines de Giga octets d'images de tableaux	VGG19 pour la recommandation de style		Temps dédié pour la réalisation de l'ensemble de l'application (reconnaissance, recommandation, etc.) court

Tableau 1 – *Cas d'étude en analyse d'image traités par des outils d'apprentissage profond*

Statistique et Société, vol. 8, n° 3 | www.statistique-et-societe.fr

Cas d'usage	Type de données	Modèle / algorithme principal	Autres outils testés ou envisageables	Contraintes et difficultés de mise en œuvre
Prédiction (*forecasting*) du niveau de stock d'une entreprise de transport de marchandises	Données de niveau de stock	ARIMA et Forêt aléatoire		Peu de données
Détection d'apnées du sommeil	Données de capteurs sur des patients (pouls, saturation en oxygène, etc.)	Forêt aléatoire	LSTM, Auto-encodeur	Données bruitées Difficulté de création de *features*
Segmentation de capteurs d'un bateau afin d'implémenter des modèles de maintenance prédictive	Un Tera-octet de données de capteurs (température, pression, tours/min., etc.)	*DTW clustering*		Environnement distribué Quantité de données conséquentes Trouver une mesure de distance convenable
Maintenance prédictive de compteurs Linky	Consommations électriques (3000 compteurs, une mesure/heure pendant deux ans)	Forêt aléatoire (FA)		Équilibrage des classes Dates de défaillance non fiables
Détection d'anomalies dans le cadre de lutte anti-fraude	Données de CRM, logs webs	*Gradient boosting*	Auto-encodeur	Équilibrage des classes Correction du biais de sélection (absence de faux positifs dans l'historique de données)

Tableau 2 – *Cas d'étude en analyse de signaux temporels traités par des outils d'apprentissage profond*

Cas d'usage	Type de données	Volumétrie des données	Outils testés	Contraintes et difficultés de mise en œuvre
Détection de thèmes – dans des retours clients – dans des tickets	Corpus de documents non-labélisés	10k à 1 million de documents	– LDA – ReGex fondées sur des règles métier	Interprétation des thèmes identifiés
Classification de documents – analyse de sentiments – mail privé / public – sujet d'un retour client – sujet d'un ticket	Corpus de documents labélisés selon leur classe	10k à 1 million de documents	– Approche traditionnelle : projection vectorielle (sac-de-mot, TF-IDF ou autre) suivie d'un modèle de classification (SVM, FA, etc.). – Apprentissage profond par RNR	Obtention d'un corpus labélisé suffisamment important
Moteur de recherche	Corpus de documents dont un sous ensemble est labélisé comme pertinent ou non selon des recherches	– Corpus total : plus de 100k documents – Sous-ensemble labéllisé : quelques milliers	– Similarité cosinus sur représentation vectorielle du document (TF-IDF, *doc2vec* ou autre) – word2vec pour la détection non-supervisée de synonymes	Obtention d'un corpus labélisé suffisamment important

Tableau 3 – *Cas d'étude en analyse du langage naturel traités par des outils d'apprentissage profond*

Cas d'usage	Type de données	Volumétrie des données	Outils testés	Contraintes et difficultés de mise en œuvre
Chatbot	Arbre de décision utilisé par les opérateurs, ou corpus de réponses préconstruites, parmi lesquelles l'algorithme doit choisir la plus pertinente, ou label pour approche supervisée	10k à 1 millions de messages	– LDA – Analyse de sentiment – Algorithme de classification	À ce jour, on vise plutôt à résoudre 80% des cas (simples), pour alléger le travail des opérationnels. Les cas compliqués sont encore très difficiles à traiter.
Reconnaissance d'entités nommées – compétences d'un CV ou d'une offre d'emploi – éléments clés d'une opération financière	Texte labellisé (chaque mot classifié dans une catégorie)	Quelques milliers de documents	– Champ aléatoire conditionnel – Chaine de Markov – RNR	Obtention d'un corpus labélisé suffisamment important

Tableau 4 – *Cas d'étude en analyse du langage naturel traités par des outils d'apprentissage profond*