Tour d'horizon autour de l'explicabilité des modèles profonds

Gaëlle Jouis, Harold Mouchère, Fabien Picarougne, Alexandre Hardouin

Tour d'horizon autour de l'explicabilité des modèles profonds

Gaëlle Jouis[1,2],Harold Mouchère[1], Fabien Picarougne[1], Alexandre Hardouin[2]

[1] LS2N, Université de Nantes, CNRS, F-44000 Nantes
[2] Pôle Emploi, Direction des Systèmes d'Information, Nantes

2-3 Juillet 2020

Résumé

Les algorithmes d'apprentissage profond ont permis d'améliorer significativement les résultats obtenus dans de nombreux domaines. Cependant, leur opacité de fonctionnement rend généralement difficile l'explication du raisonnement conduisant aux résultats. Ces modèles souffrent d'une faible acceptabilité et peuvent dans certains domaines poser un problème d'ordre légal. Pour y remédier, le domaine de l'Intelligence Artificielle eXplicable (XAI) connaît aujourd'hui un essor important. Cette contribution présente un état de l'art de différentes approches afin de rendre ces modèles plus explicables.

Mots-clés

Apprentissage Automatique, Apprentissage Profond, Explicabilité, CNN, LSTM, Réseaux à Attention

Abstract

Deep learning algorithms have significantly improved the results obtained in several fields. However, explaining the mechanisms that lead to those results can be tricky due to the operating opacity of these models. This lack of transparency leads to poor acceptance, and even to legal issues for some areas. To tackle this, the eXplainable Artificial Intelligence (XAI) research field is experiencing significant growth. This contribution introduces several state-of-the-art approaches that allow further explainability of deep models.

Keywords

Machine Learning, Deep Learning, Explainability, CNN, LSTM, Attention Networks

1 Introduction

Le domaine de l'intelligence artificielle et plus généralement de l'apprentissage a su trouver sa place dans de très nombreux secteurs de l'informatique. Depuis une dizaine d'années et dans de nombreux domaines incluant notamment le traitement d'images, la détection d'objets, le traitement textuel et linguistique, les algorithmes d'apprentissage profond ont montré leur efficacité avec des performances parfois impressionnantes [30]. À la différence d'autres approches utilisées en apprentissage, comme les modèles linéaires ou les arbres de décision par exemple, leur fonctionnement est généralement opaque et considéré comme une *boîte noire*. Par conséquent, l'explication du raisonnement conduisant aux résultats est rendu plus difficile.

Or une plus grande transparence peut permettre d'améliorer un modèle [22, 27]. De plus, la compréhensibilité d'un modèle par un humain est intéressante dans plusieurs situations, notamment si l'on tient compte des problématiques d'acceptabilité par les utilisateurs. Cet effet *boîte noire* peut en effet engendrer des effets pervers dans l'utilisation des outils. Ainsi, si les utilisateurs n'ont pas confiance, peu importe la valeur de l'outil, il ne sera jamais exploité. A contrario, si les utilisateurs ont une confiance totale dans le système, les éventuelles dérives de l'outil ne seront pas détectées par les utilisateurs [5]. Par ailleurs, des problématiques éthiques et légales se posent lorsque les algorithmes peuvent avoir une incidence sur la vie d'une personne [13]. C'est particulièrement le cas dans le domaine médical, mais également dans l'accompagnement professionnel des personnes. Par exemple, un service publique comme Pôle Emploi a l'obligation légale de pouvoir fournir les règles et algorithmes ayant une influence sur les utilisateurs ; c'est le cas si un algorithme permet de proposer des emplois aux demandeurs d'emploi. La question se pose alors de trouver le bon compromis entre la performance du modèle et son explicabilité.

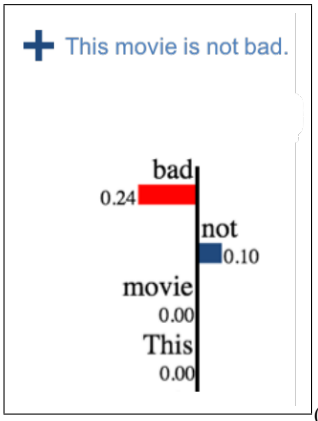
La problématique de l'interprétabilité et l'acceptabilité des modèles est récente mais prend de l'ampleur. La littérature scientifique à ce sujet foisonne depuis quelques années, et des ateliers spécialisés apparaissent dans les conférences sur l'intelligence artificielle (KDD, AAAI, NIPS). L'objectif de cette contribution est de faire un tour d'horizon qui ne saurait être exhaustif de différentes méthodes d'explication de modèles, en se concentrant, sans s'y contraindre, sur les techniques liées à l'analyse de textes. Dans un premier temps, un point sémantique sera fait sur la définition d'une explication, afin d'éviter toute confusion. Ensuite, une taxonomie des méthodes d'explication sera décrite, et chaque approche sera étudiée. Enfin, une discussion concluera le document.

2 Qu'est-ce qu'une explication ?

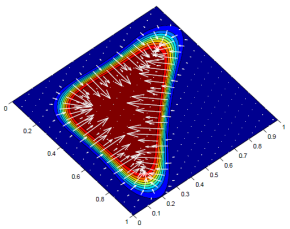
Quelques définitions. L'explicabilité est la capacité d'un humain à comprendre les décisions d'un système étant donné le contexte ; le terme "interprétabilité" est également employé [17]. Pour être explicable, un modèle doit être

compréhensible dans sa globalité, ou fournir une explication (locale) liée à chaque décision [20]. L'explication peut prendre la forme d'une justification – comme une règle logique – pour un ensemble pertinent de résultats [6]. C'est notamment le cas des Ancres [21] ou des visualisations du mécanisme d'attention [14], qui seront détaillés en sections 3.1 et 3.3 respectivement. Une explication peut également se faire par contraste : en se concentrant sur les variations menant à des décisions algorithmiques différentes. C'est le cas des vecteurs d'explication locale [2], qui indiquent quelles variations des valeurs des variables en entrée changent le résultat du modèle.

Différentes visualisations d'une explication sont présentées dans la Figure 1 : LIME (Fig. 1a), affichant l'importance de chaque mot observé pour une classe donnée, et les vecteurs d'explication locale (Fig. 1b) sont assez techniques. Les explications mettant en avant les parties importantes des images comme les ancres (Fig. 1c) et la méthode grad-CAM (Fig. 1d) sont plus visuelles. Parmi les explications illustrées, les vecteurs d'explication locale sont les seuls à fournir une vision globale du modèle.

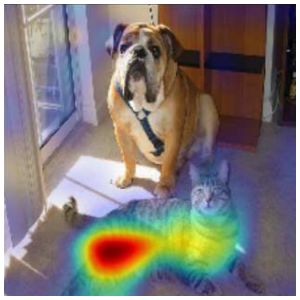


(a) Lime [20]



(b) Vecteurs d'explication locales [2]

(c) Anchor [21]



(d) Grad-CAM [23]

FIGURE 1 – Différentes visualisations d'une explication.

Comme le montre la figure 1, les explications peuvent avoir des formes variées. Dans un contexte donné, elles doivent être le bon compromis entre la fidélité au modèle et la compréhension pour l'utilisateur. Une explication doit alors prendre en compte l'utilisateur humain, notamment son expertise, et le temps qu'il possède pour recevoir l'explication [9].

Evaluation. Évaluer l'explicabilité (ou inversement la complexité) d'un modèle n'est pas systématiquement réalisé dans la littérature. Une métrique naïve comme la taille du modèle (nombre de paramètres) est une première approche, mais cette mesure ne rend pas compte de ce qu'a appris le modèle.
Pour plus de précision, les évaluations des explications sont réalisées avec des utilisateurs réels ou simulés (modèles appris) [20, 21]. Il est recommandé que la tâche effectuée lors de l'évaluation représente au mieux l'environnement fonctionnel dans lequel est utilisé le système créé, avec les utilisateurs réels [5]. Ceci permet de prendre en compte le contexte inhérent aux explications et les spécificités des utilisateurs.
Les métriques d'évaluation peuvent être objectives (efficacité de l'utilisateur utilisant le modèle, capacité à prédire les résultats du modèle, temps de réponse lorsque l'utilisateur a pour tâche d'imiter le modèle [21] par exemple) ou subjective quand le but est de favoriser l'acceptation d'un modèle par un utilisateur.

3 Différentes méthodes d'explication

Dans la littérature les auteurs divisent les types d'explication de modèles en fonction de la nature des approches utilisés, mais avec une même logique générale [5, 8, 9, 22]. On peut classer ces méthodes par la transparence du système étudié, ce qui donne les catégories suivantes.

1. Expliquer un modèle boîte noire au travers de ses entrées et sorties, soit en observant directement l'influence des premières sur les secondes, soit en créant un modèle interprétable mimant la boîte noire. Ces méthodes sont totalement indépendantes de la structure interne du modèle boire noire.
2. Observer les mécanismes internes d'un système (boîte grise) après son entraînement ; afin d'y détecter des schémas et les interpréter, ces méthodes sont donc dépendantes de l'architecture interne du modèle observé.
3. Concevoir un modèle ou une solution transparente (boîte blanche) de par son architecture ; en lui associant des contraintes compréhensibles pour un humain (sous forme de règles par exemple) ou en générant des explications en plus du résultat attendu.

Le choix d'une méthode par rapport à une autre est généralement guidé par des contraintes techniques ou d'organisation du projet (nature du modèle, apparition de la problématique d'explicabilité avant ou après conception du modèle). Les contraintes de l'utilisateur doivent également être prises en compte : le temps à disposition pour recevoir les explications, le niveau d'expertise applicative, le niveau de connaissances en apprentissage automatique et l'intérêt porté au domaine, sont autant de paramètres qui peuvent fortement influer sur le choix d'une méthode. De même, les explications peuvent porter sur le comportement global du modèle, ou sur un résultat particulier. Dans le second cas, on parlera dans la littérature d'explications locales [20]. Celles-ci sont particulièrement adaptées

si l'utilisateur a peu de temps, car elles se rapportent à des exemples concrets
En accord avec la taxonomie présentée ci-avant, chacune des trois approches sera détaillée au travers de plusieurs méthodes.

3.1 Explications indépendantes du modèle

Ce type de méthode a pour principal avantage de pouvoir être utilisé sur tous les modèles. Cela évite de se contraindre techniquement dans le choix d'un modèle. Les explications sont basées sur l'impact des valeurs des variables d'entrée sur la sortie du modèle. Les explications relèvent donc de la corrélation, pas nécessairement de la causalité. De même, les variables en entrée peuvent être trop nombreuses pour être assimilées aisément par un utilisateur. C'est par exemple le cas en analyse sémantique, où ces entrées sont des n-grammes de mots ou de caractères, notion illustrée ci dessous (Fig. 2).

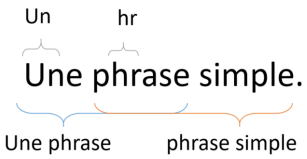


FIGURE 2 – Exemples de 2-grammes de caractères (en haut) et de mots (en bas).

Vecteurs d'explication locale. Dans [2] les auteurs présentent des *vecteurs d'explication locale* qui quantifient l'importance de chaque variable d'entrée pour une instance donnée. Plus exactement, un vecteur d'explication indique dans quelle direction changer une variable pour que le modèle change de classe prédite. On obtient donc un vecteur indiquant le rôle de chaque variable dans la classification d'une instance, pour une classe donnée. En observant un ensemble de vecteurs, on peut également avoir une visualisation plus globale du modèle. Une illustration est donnée en Fig. 1b avec deux classes (bleu ou rouge) et deux variables d'entrées correspondant aux deux dimensions du plan. Le calcul du vecteur d'explication se fait sur des modèles donnant une probabilité pour une entrée d'appartenir à chaque classe. Si elle n'est pas disponible, il faut l'estimer avec un autre modèle avant de calculer les vecteurs. C'est donc une approche qui n'est pas totalement indépendante du modèle. Par ailleurs, ces vecteurs sont difficilement interprétables pour les utilisateurs, surtout dans le cas de l'analyse sémantique où la dimension est la taille du vocabulaire du corpus. Il faut donc envisager une visualisation ou un résumé de l'information apportée par ces vecteurs pour en tirer une explication.

Valeurs de Shapley. Inspiré de la théorie des jeux, les valeurs de Shapley donnent un aperçu de la contribution d'un élément dans un ensemble par rapport à un résultat final [26]. La dimension des explications obtenues est égale à la dimension des données en entrée. Elles nécessitent donc un traitement a posteriori pour en retirer des explications claires. Le calcul des valeurs de Shapley nécessite d'avoir accès au jeu de données d'entraînement du modèle, ce qui est une limitation forte et sous-entend un calcul éventuellement long. Pour limiter le temps de calcul, le module SHAP (SHapley Additive exPlanation) a été développé en se basant sur les valeurs de Shapley [16]. Toutefois, les méthodes proposées sont des estimations des valeurs SHAP, avec plusieurs approches présentées, dépendantes et indépendantes du modèle étudié.

Distillation de connaissance. Une autre manière de rendre les modèles interprétables est de se concentrer sur des structures plus petites. Pour cela, les auteurs de [11] présentent le concept de la distillation de connaissances dans les modèles, où l'idée est d'entraîner un ensemble complexe de réseaux de neurones et transférer les structures apprises à un réseau de neurones plus simple. Instinctivement, il est tentant de se dire qu'un réseau plus simple pourrait être plus interprétable. Les auteurs de [7] proposent de mesurer l'interprétabilité comme étant le ratio entre la performance du modèle simple et du modèle complexe. En appliquant ces mesures sur un cas d'étude avec un réseau de neurones complexe et un réseau de neurones simplifié, ils en déduisent que la Distillation améliore la robustesse (résistance aux attaques par modifications subtiles des entrées du modèle) au détriment de l'interprétabilité. En revanche, la Distillation est également utilisée pour créer des arbres de décision [15] ou des Gradient Boosted Trees [3] à partir de réseaux de neurones, ce qui revient à créer un module d'explication sous forme d'arbres de décisions. Si l'approche est décrite pour des réseaux de neurones, elle est applicable à d'autres modèles.

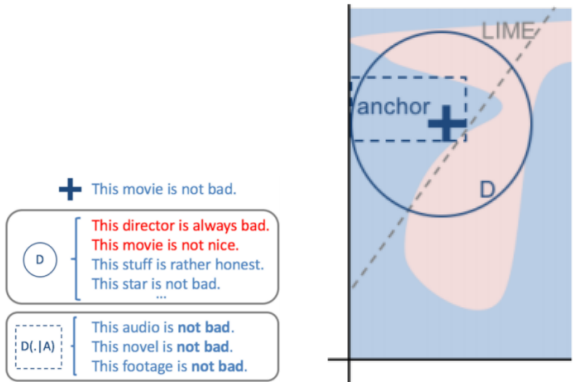
Approximation linéaire locale. Les auteurs de [20] proposent l'outil *LIME* (Local Interpretable Model-agnostic Explanations) pour rendre compte du comportement local d'un modèle. LIME fonctionne par approximation linéaire du modèle autour d'une instance donnée. C'est ce modèle linéaire qui est ensuite utilisé pour générer des explications. Ces explications correspondent aux variables d'entrée qui impactent le plus la sortie du modèle (Fig. 1a). Pour l'analyse de texte, ce sont les mots du texte associés à une quantification de l'influence (positive ou négative) sur la réponse du modèle.

Pour obtenir une explication plus globale d'un modèle, SP-LIME est présenté en [20]. SP-LIME est une méthode se basant sur les explications locales de LIME pour en sélectionner un ensemble restreint. Cette méthode est intéressante car elle est généralisable au delà de LIME [21]. Par contre, elle nécessite de générer en amont toutes les explications des instances d'un ensemble de données pour pouvoir les sélectionner ensuite. Cet ensemble de données étant idéalement un ensemble de test.

Les avantages de cette approximation linéaire sont la simplicité et la rapidité de la création de l'explication. Toutefois il faut faire l'hypothèse forte que le modèle se comporte linéairement autour de l'instance expliquée. Le principal inconvénient est le fait que l'explication soit une approximation locale du modèle sans limite définie. Il est en effet difficile pour un utilisateur humain de savoir à quel

point ces explications sont généralisables.

Ancres. Pour contrer ce problème, les auteurs de LIME ont proposé une amélioration de leur méthode, les Ancres [21]. En conservant l'idée d'approximation locale du modèle, les auteurs sont passés d'une approximation par un modèle linéaire à une explication sous forme de règle. L'idée est de mieux définir le contexte dans lequel l'explication générée est valable. Soient un modèle $f : X \to Y$, une instance $x \in X$, un résultat $y \in Y$ choisi, et une ancre A associée. A est une condition telle que si x respecte cette condition, alors la probabilité que $f(x) = y$ est grande. L'ancre est construite de sorte à maximiser cette probabilité, il est toutefois possible que $f(x) \neq y$. On note $D(.|A)$ l'ensemble des $x \in X$ qui respectent la condition A. Une ancre intéressante s'applique à un ensemble $D(.|A)$ le plus grand possible relativement à X. Si les entrées sont des textes, une ancre est un ensemble de mots ou n-grammes.
Dans l'exemple illustré en Fig. 3, le modèle étudié classe des phrases selon deux catégories : "positive" et "negative". L'instance d'origine est la phrase "This movie is not bad", et est classée "positive". L'ancre associée est la règle $A = \{not, bad\} \to Positive$. L'ensemble $D(.|A)$ de la Fig. 3a, représenté par un rectangle dans la Fig. 3b, regroupe les entrées possédant les variables de l'ancre. Dans notre exemple, $D(.|A)$ correspond aux textes comprenant les mots "not" et "bad" de l'ensemble des variations de l'instance d'origine (ensemble D de la Fig. 3). L'ensemble D est obtenu en appliquant des variations cohérentes à l'instance d'origine. Appliqué à l'exemple, cela correspond à remplacer un ou plusieurs mots de la phrase par des mots de nature similaire. Remplacer un adjectif par un autre adjectif est une variation cohérente de l'instance d'origine.



(a) Une entrée, ses variations D et leur sous ensemble correspondant à l'ancre $D(.|A)$

(b) Représentation graphique de l'entrée, et des ensembles D et $D(.|A)$

FIGURE 3 – Textes similaires à une entrée D et son sous ensemble $D(.|A)$ correspondant à l'ancre $\{not, bad\} \to Positive$. Source : [21]

Pour sélectionner une ancre, les auteurs choisissent de maximiser deux paramètres. Le premier est la précision, qui est maximale si les éléments de l'ensemble $D(.|A)$ ont la même sortie que l'instance d'origine. Le second est la couverture, soit la taille de l'ensemble $D(.|A)$ par rapport à l'ensemble D. Une ancre avec une forte précision est une explication fidèle au modèle boîte noire. Si elle a une bonne couverture, une ancre est assez généralisable.
Le travail autour des ancres met en avant la relation entre l'utilisateur et l'explication. La limite de LIME contournée par les Ancres est la difficulté pour l'utilisateur à déterminer la validité d'une l'explication donnée. Ce faisant, l'explication donnée, à savoir une règle, est également plus facile à aborder qu'un ensemble de poids comme le résultat de base de LIME.

Les explications indépendantes des modèles sont une première approche en considérant les modèles boîtes noires, mais il s'agit plus de mettre en avant des corrélations entre les entrées et les résultats du système. Pour aller plus loin, il est possible de s'appuyer sur la structure du modèle pour générer des explications. La *boîte noire* devient alors une *boîte grise*.

3.2 Explications dépendantes du modèle

Les explications dépendantes du modèle sont basées sur l'observation des paramètres du modèle après son entraînement. Cette observation, pour conserver la métaphore de la boite noire, revient à ouvrir cette boite et regarder à l'intérieur. Les explications sont alors plus fidèles au fonctionnement du système étudié que les méthodes indépendantes du modèle. Toutefois, cette approche contraint fortement le choix du modèle.

Réseaux à convolutions. Dès 2013, des travaux sur les réseaux de neurones proposent des visualisations de leur fonctionnement, dans le cadre de la classification d'images. Dans [29], les auteurs proposent de mieux appréhender le fonctionnement des réseaux à convolution (Convolutionnal Neural Networks, CNN), par une visualisation directe des motifs d'activation des neurones par couche. Ils utilisent pour cela un réseau de neurones appelé *Deconvolutionnal Network*. Ces travaux permettent d'avoir un aperçu des motifs reconnus par chaque couche, motifs simples sur les couches basses et plus semblables aux classes détectées sur les couches hautes. Les auteurs vérifient également le comportement de leur modèle en masquant certaines parties des images et en observant les éventuelles variations de classification induites ; cette approche étant reprise dans [20] et [21].
Toujours sur les CNN, les auteurs de [23] présentent la méthode *Grad-CAM*. Elle permet de générer des cartes de chaleur des endroits de l'image aidant à la détection d'une classe en particulier (Fig 1d). Ils mélangent leurs visualisations à celles de [25, 29], afin d'obtenir les parties de l'image ainsi que les motifs précis permettant la classification. Des travaux sont également effectués sur les cartographies des caractéristiques saillantes (*Saliency maps*) des images [24]. Ces travaux permettent d'avoir une indication fidèle du fonctionnement du modèle, mais les techniques sont plutôt orientées analyse d'image.

Long-Short Term Memory (LSTM). Dans la même philosophie, les auteurs de [12] essaient de comprendre les forces et les limites des réseaux de neurones de type

LSTM, appliqués à l'analyse de textes. Pour leur analyse, ils génèrent des visualisations sur des motifs spécifiques dans les données entraînant les activations de certaines cellules. Un exemple du papier est l'activation de certaines cellules en fonction des caractères rencontrés dans un texte. La visualisation met en évidence la détection du texte entre guillemets.

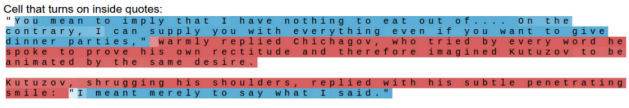


FIGURE 4 – Activation d'une cellule en fonction des guillemets dans le texte. Source : [12]

Ce type de réseau de neurones est largement utilisé dans l'analyse sémantique. Les visualisations peuvent donner une idée précise du fonctionnement du modèle. toutefois, elles sont complexes ou nécessitent un travail de recherche et d'analyse, d'autant plus si on considère chaque cellule. Pour donner un ordre d'idée, des structures de LSTM de la littérature peuvent avoir 300 [14] ou encore 512 [12] cellules. Il convient alors de toutes les explorer pour trouver, pour une petite partie d'entre elles, des activations significatives. Si ce type d'approche permet de mieux comprendre les indices utilisés par le réseau, il ne s'agit pas forcément d'une explication de la décision finale.

Décomposition pixel par pixel. La décomposition pixel par pixel (*Pixel-Wise Decomposition*) est une stratégie permettant d'expliquer les résultats d'un classifieur d'images en créant une carte de chaleur des pixels les plus pertinents pour une prédiction donnée [1]. Pour calculer la pertinence R (*Relevance*) de leurs variables d'entrée (dans leur exemple, des pixels), ils décomposent la prédiction $f(x)$ comme étant la somme des contributions des neurones de la couche précédente et appliquent itérativement cette propriété jusqu'à arriver à la couche d'entrée de leur réseau. Les auteurs décrivent deux manières d'y parvenir. La première est la propagation de pertinence couche par couche (*Layer-wise Relevance Propagation*), un concept regroupant diverses solutions de décomposition respectant certains critères. La seconde est une approche basée sur la décomposition de Taylor, qui permet une approximation de la propagation de pertinence couche par couche, en s'appuyant sur le principe de décomposition de fonctions pour décomposer directement le classifieur f. Ces travaux sont approfondis dans [18] où les auteurs proposent la *Deep Taylor Decomposition*, qui est une adaptation de la décomposition de Taylor, appliquée non pas au modèle entier mais à chaque fonction de pertinence $R_j(x_i)$ entre un neurone j d'une couche n et les neurones x_i de la couche $n-1$. Dans ce dernier article les auteurs illustrent leur travaux avec des réseaux de neurones classant des images, mais ce type de méthode peut être appliqué à d'autres modèles et peut également être élargi à d'autres types d'entrées, comme les textes dans [18].

Les méthodes décrites précédemment permettent de mieux appréhender le fonctionnement des modèles, en se basant sur leurs caractéristiques respectives. Toutefois ces méthodes, dépendantes ou indépendantes du modèle, nécessitent des calculs ou de l'analyse d'un grand nombre d'éléments. Pour éviter cela, la dernière approche est de créer un modèle dont la structure même le rend plus transparent.

3.3 Modèle Interprétable

L'idée de modèle transparent est de limiter le besoin en analyse post entraînement en s'appuyant sur des structures spécifiques du modèle. De cette manière une meilleure fidélité au modèle est assurée tout en limitant les calculs et approximations.

Architectures simplifiées. Dans le courant de simplification des réseaux, des expérimentations mettent en évidence l'efficacité d'architectures de réseaux de neurones simplistes mais capables de résoudre des problèmes complexes, comme le stationnement d'une voiture miniature [10]. Ce type de réseau possède une topologie inspirée du système nerveux du ver *C. elegans*. Le papier définit un réseau ainsi constitué de 12 neurones, en l'entraînant sur la tâche de stationner un robot. En observant les activations des neurones en fonction des phases (tourner à gauche, à droite ou avancer), les auteurs mettent en lumière le rôles des neurones dans chaque phase de la tâche accomplie, en mettant en évidence par exemple les neurones s'activant lorsque le robot doit tourner à droite. Ces activations sont interprétables notamment parce que le réseau est composé de peu de neurones ; cela facilitant grandement l'analyse des activations. Cette approche permet ainsi de réaliser un travail similaire à [12], qui analyse les activations de cellules LSTM, mais sur un nombre réduit de neurones.

Mécanismes d'attention. Les mécanismes d'attention dans les réseaux de neurones sont une manière de rendre les modèles directement plus interprétables. Dans [14], les auteurs créent un plongement de mots via un réseau avec une partie LSTM et une partie basée sur l'attention. Chaque phrase est représentée par une matrice $M = AH$, où A est la matrice d'attention et H les états cachés de la couche LSTM. Les vecteurs d'attention a composant A vont se concentrer sur des aspects différents de la phrase. En sommant et normalisant (softmax) tous les vecteurs d'attention a, les mots fortement considérés par le plongement ressortent avec les poids les plus forts. Cette solution permet donc d'avoir une visualisation claire des variables importantes en entrée du réseau, en observant les paramètres du modèle. Un exemple de visualisation est présenté dans le cadre de la traduction de textes dans [19] (Fig. 5). Elle met en avant l'inversion des mots entre l'entrée en anglais (*european economic area*") et la traduction française ("*zone économique européenne*"). La visualisation des poids d'attention est également utilisée dans [28] afin de valider l'intérêt de leur topologie de réseau, également composé d'un LSTM et d'une couche d'attention. Le principal intérêt de l'analyse de l'attention est qu'il n'y a pas besoin de calculs supplémentaires d'une métrique spécifique une fois le modèle entraîné, contrairement à [1] par exemple.

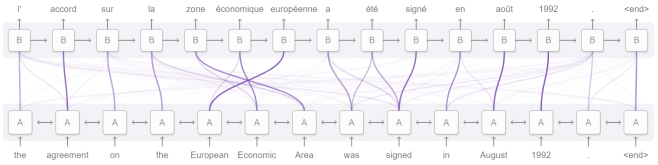


FIGURE 5 – Visualisation de l'attention pour une tâche de traduction. Source : [19]

Génération d'explications. Certains systèmes peuvent également générer d'eux même des explications autour d'une décision. C'est le cas de [4] où les explications d'un système de recommandation sont générées sous forme de critiques d'utilisateurs ("i wouldn't recommend it.") via un LSTM. Le principe de l'expérimentation est de reconstruire une critique que produirait un utilisateur avec un texte le plus naturel possible. Les auteurs évaluent leurs explications avec des métriques de lisibilité de textes tel que le score *Flesch-Reading-Ease*. Si le système de recommandation n'est pas nécessairement un réseau de neurones, un système similaire peut être appliqué à tout système de prédiction basé sur des textes.

Ce tour d'horizon de différentes méthodes donne un aperçu des approches possibles, mettant en avant la fidélité de l'explication donnée ou sa capacité à être comprise et acceptée par l'utilisateur final.

4 Discussion

L'analyse de la littérature met en lumière un nombre significatif de méthodes, parfois très proches. Leur finalité commune est d'améliorer l'explicabilité des modèles, en particulier dans le domaine de l'apprentissage profond. Pour chaque problème, il s'agit alors de trouver et d'adapter les méthodes en fonction des contraintes rencontrées. Le choix d'une approche d'explication peut alors devenir une contrainte technique du modèle et guider les développements futurs. Ainsi plus la réflexion est menée tôt, plus le choix des approches est grand.

Ces contraintes sont aussi humaines : le public cible des explications doit être défini et caractérisé. Les notions d'intérêt et de confiance dans les technologies dites d'intelligence artificielle sont des caractéristiques importantes des utilisateurs. La prise en compte du temps disponible pour recevoir l'explication est également primordiale.

Un enjeux important du travail autour de l'explicabilité est de normaliser l'évaluation des différentes méthodes et de leurs adaptations. Aujourd'hui, ces évaluations ne sont pas systématisées et il n'existe pas de consensus, rendant la comparaison des méthodes délicate.

Références

[1] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7) :e0130140, 2015.

[2] David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Müller. How to explain individual classification decisions. *J. Mach. Learn. Res.*, 11 :1803–1831, August 2010.

[3] Zhengping Che, Sanjay Purushotham, Robinder Khemani, and Yan Liu. Distilling knowledge from deep networks with applications to healthcare domain. *arXiv preprint arXiv :1512.03542*, 2015.

[4] Felipe Costa, Sixun Ouyang, Peter Dolog, and Aonghus Lawlor. Automatic generation of natural language explanations. In *Proceedings of the 23rd International Conference on Intelligent User Interfaces Companion*, pages 1–2, 2018.

[5] Hoa Khanh Dam, Truyen Tran, and Aditya Ghose. Explainable software analytics. In *Proceedings of the 40th International Conference on Software Engineering : New Ideas and Emerging Results*, pages 53–56, 2018.

[6] Sonia Desmoulin-Canselier and Daniel Le Métayer. Algorithmic decision systems in the health and justice sectors : Certification and explanations for algorithms in european and french law. *European Journal of Law and Technology*, 9(3), 2019.

[7] Amit Dhurandhar, Vijay Iyengar, Ronny Luss, and Karthikeyan Shanmugam. Tip : Typifying the interpretability of procedures. *arXiv preprint arXiv :1706.02952*, 2017.

[8] Leilani H Gilpin, David Bau, Ben Z Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. Explaining explanations : An overview of interpretability of machine learning. In *2018 IEEE 5th International Conference on data science and advanced analytics (DSAA)*, pages 80–89. IEEE, 2018.

[9] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5) :1–42, 2018.

[10] Ramin Hasani, Mathias Lechner, Alexander Amini, Daniela Rus, and Radu Grosu. Can a compact neuronal circuit policy be re-purposed to learn simple robotic control ?

[11] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv :1503.02531*, 2015.

[12] Andrej Karpathy, Justin Johnson, and Fei-Fei Li. Visualizing and understanding recurrent networks. *CoRR*, abs/1506.02078, 2015.

[13] Legifrance. Loi n 2016-1321 du 7 octobre 2016 pour une republique numerique, 2016.

[14] Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. A structured self-attentive sentence embedding. *arXiv preprint arXiv :1703.03130*, 2017.

[15] Xuan Liu, Xiaoguang Wang, and Stan Matwin. Improving the interpretability of deep neural networks with knowledge distillation. In *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 905–912. IEEE, 2018.

[16] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc., 2017.

[17] Tim Miller. Explanation in artificial intelligence : Insights from the social sciences. *Artificial Intelligence*, 267 :1–38, 2019.

[18] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition*, 65 :211–222, 2017.

[19] Chris Olah and Shan Carter. Attention and augmented recurrent neural networks. *Distill*, 2016.

[20] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "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*, KDD '16, pages 1135–1144, New York, NY, USA, 2016. ACM.

[21] Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors : High-precision model-agnostic explanations. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 1527–1535, 2018.

[22] Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. Explainable artificial intelligence : Understanding, visualizing and interpreting deep learning models. *ITU Journal : ICT Discoveries - Special Issue 1 - The Impact of Artificial Intelligence (AI) on Communication Networks and Services*, 1 :1–10, 10 2017.

[23] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam : Visual explanations from deep networks via gradient-based localization. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[24] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks : Visualising image classification models and saliency maps. *arXiv preprint arXiv :1312.6034*, 2013.

[25] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity : The all convolutional net. *arXiv preprint arXiv :1412.6806*, 2014.

[26] Erik Strumbelj and Igor Kononenko. An efficient explanation of individual classifications using game theory. *Journal of Machine Learning Research*, 11(Jan) :1–18, 2010.

[27] Stefano Teso and Kristian Kersting. Explanatory interactive machine learning. In *http ://www. aies-conference. com/accepted-papers/*. AAAI, 2019.

[28] Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 606–615, 2016.

[29] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[30] Zhengxia Zou, Zhenwei Shi, Yuhong Guo, and Jieping Ye. Object detection in 20 years : A survey. *arXiv preprint arXiv :1905.05055*, 2019.