LINC

Laboratoire d'Innovation Numérique de la CNIL

Dossier

Sécurité des systèmes d'IA

Table des matières

Introduction - résumé exécutif

Les systèmes d'IA engendrent des risques de sécurité spécifiques en comparaison à des systèmes d'information classiques. En effet, les nouvelles capacités d'apprentissage automatique (*machine learning*) augmentent la « surface d'attaque » de ces systèmes en introduisant de nombreuses (et nouvelles !) vulnérabilités. LINC propose ici un triptyque d'articles visant à i) présenter les attaques pouvant être menées sur un système d'IA, ii) détailler comment mener une analyse de risque prenant en compte les enjeux de sécurité et de protection des données et iii) présenter les bonnes pratiques pour la sécurisation d'un système d'IA.

Crédit : Blue Coat Photos

Face à la complexité et au volume croissants des cyberattaques, il est désormais évident que l'IA peut apporter de nouvelles réponses aux risques de sécurité. En effet, l'IA peut, avec sa capacité à analyser un contexte donné, contribuer à y détecter des anomalies ou des comportements inhabituels, révélateurs d'attaques et ainsi, à renforcer les outils de protection, détection, réponse et remédiation : augmentation du taux de détection, détection au plus tôt des attaques, amélioration de la capacité d'adaptation aux évolutions permanentes des systèmes d'information, etc. A titre d'exemple, de nombreuses sondes réseaux (IDS, *intrusion detection systems*) et déployées sur des terminaux (EDR, *endpoint detection and response*) intègrent désormais des technologies d'IA, en complément de leur moteur de détection par signature permettant de reconnaître un programme malveillant. Ces systèmes apprennent ainsi le comportement « normal » de l'infrastructure et toute déviation observée par la suite est signalée à un analyste comme un potentiel risque cyber (d'autres exemples peuvent être trouvés ici).

Petite taxonomie des attaques des systèmes d'IA

Si les technologies d'IA peuvent améliorer la sécurité des systèmes d'information, elles engendrent également des risques spécifiques. En effet, le problème réside dans le fait qu'avec les nouvelles capacités de l'apprentissage automatique et dans la perspective de son utilisation de plus en plus large, sont introduites de très nombreuses vulnérabilités que des attaquants sont susceptibles d'exploiter. Certaines

de ces vulnérabilités peuvent permettre de perturber le fonctionnement du modèle et l'amener à émettre une prédiction incorrecte. D'autres, en revanche, laissent un attaquant libre d'extraire des informations sensibles du modèle, telles que les données sous-jacentes ou le modèle lui-même.

Plusieurs travaux scientifiques recensent les différents types d'attaques comme par exemple (Pitropakis et al., 2019). Par ailleurs, le laboratoire MITRE – une organisation à but non lucratif américaine dont l'objectif est de travailler pour l'intérêt public dans les domaines de l'ingénierie des systèmes, la technologie de l'information, les concepts opérationnels, et la modernisation des entreprises – a lancé l'initiative ATLAS (*Adversarial Threat Landscape for Artificial Intelligence Systems*). Celle-ci est une base de connaissances des méthodes, techniques et études de cas d'attaques menées à l'encontre de systèmes d'apprentissage automatique, constituée à partir d'observations du monde réel, de démonstrations réalisées par des experts en sécurité (*pentesters, red teamers*, etc.) et de connaissances issues de la recherche universitaire.

Les travaux sur le sujet de la sécurité des modèles d'IA ont en effet connu un intérêt croissant au cours des dernières années, ce qui est mesurable au nombre de publications scientifiques sur le sujet. La Figure 1 illustre cette tendance sur une sous-partie du domaine, les exemples adversaires introduits en 2014 par (Goodfellow et al., 2014). En pratique, de la même façon que ce qui existe dans le domaine de la cryptographie depuis toujours, on observe une course permanente entre les méthodes d'attaque et de défense, aucune de ces dernières ne garantissant à ce jour la robustesse des systèmes dans 100% des cas.

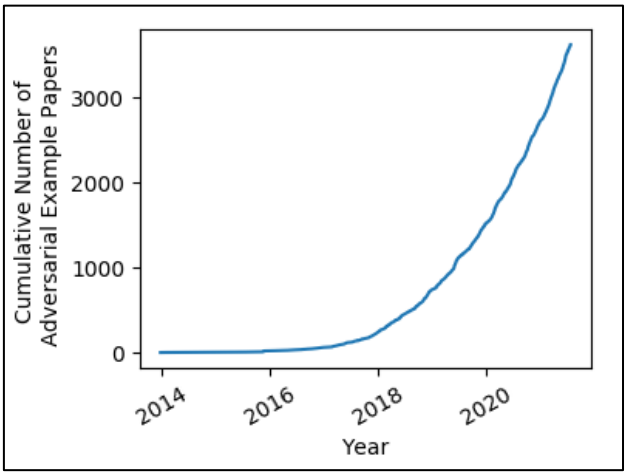


Figure 1. Evolution du nombre d'articles dédiés au sujet des exemples adversaires (source blog de Nicholas Carlini).

P[ã]nser la sécurité des systèmes d'IA

Au cours des dernières années, de grandes entreprises telles que Google, Amazon, Microsoft et Tesla ont vu certains de leurs systèmes d'IA attaqués. Cette tendance est amenée à s'accentuer. Selon un rapport de l'entreprise de conseil Gartner de 2019, 30% des cyberattaques d'ici 2022 impliqueront des vols de modèles (*model theft*), l'utilisation d'exemples contradictoires (*aversarial examples*) ou l'empoisonnement de données (*data poisoning*). Ces attaques sont d'autant plus probables que les systèmes d'IA seront progressivement installés dans de nombreux environnements et *a fortiori* serviront de plus en plus de support à des décisions automatiques pouvant présenter un intérêt pour une personnelle malveillante. A titre d'exemple, un concurrent peut envisager tenter de « saboter » un logiciel d'IA utilisé pour le contrôle qualité dans un environnement industriel, un attaquant peut chercher à optimiser l'analyse d'une demande de crédit, etc. Dans son rapport d'avril 2021, la startup israélienne Adversa, spécialisée dans la sécurité des systèmes d'IA, détaille les applications les plus attaquées aujourd'hui (voir Figure 2).

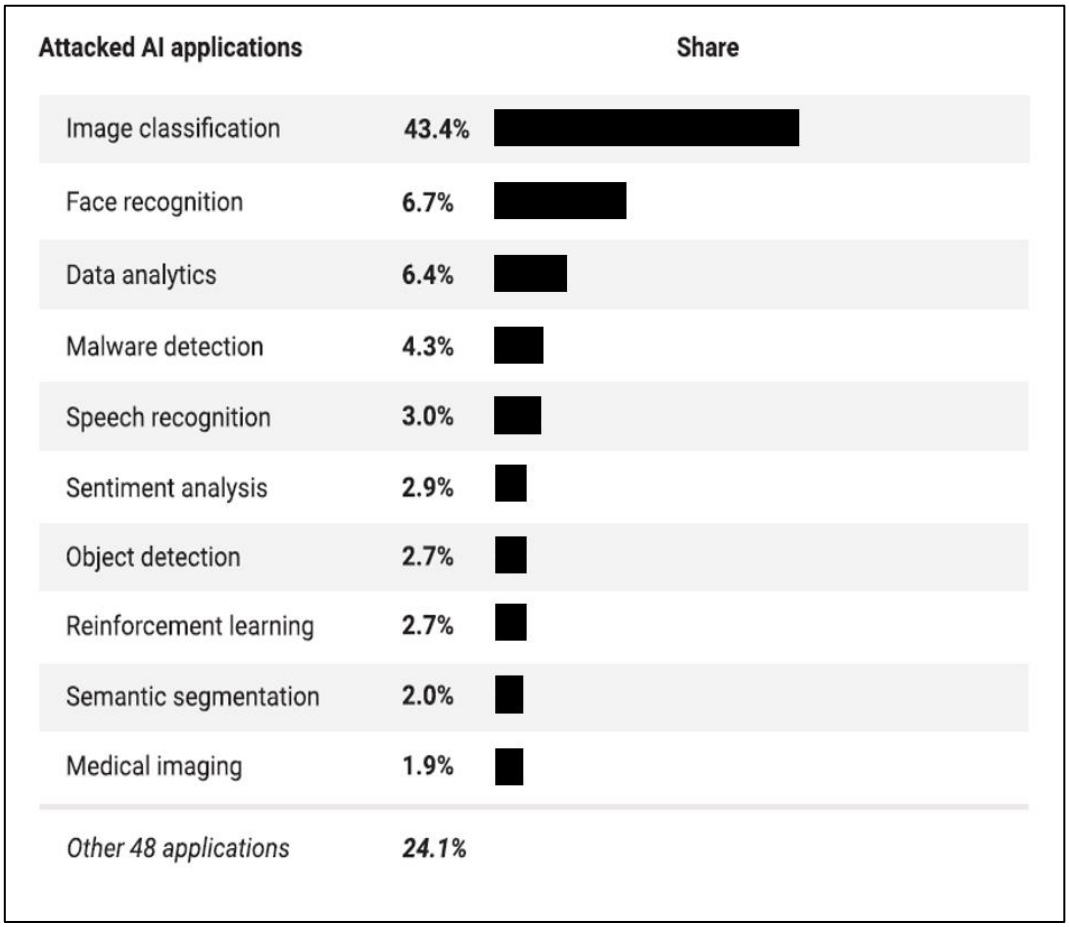


Figure 2. Part des attaques d'IA par type d'application (source Adversa).

Aujourd'hui, le monde de l'industrie apparaît mal préparé à faire face à de telles menaces comme l'indique une enquête menée en 2021 auprès de 28 organisations, petites et grandes, par (Shankar et al., 2021) : vingt-cinq d'entre elles ne savaient pas comment sécuriser leurs systèmes d'IA. Pour ces raisons, il est essentiel de sensibiliser les organismes aux problématiques de sécurisation de leurs systèmes d'IA et de proposer des outils d'analyse de risque adaptés.

Sécurité des systèmes d'IA, les gestes qui sauvent

Il existe de nombreuses raisons pour lesquelles il est très difficile de protéger un système d'IA. Ces raisons tiennent aux limitations de ces derniers. Ainsi en raison de leur construction statistique, les modèles d'IA ne sont jamais parfaits. Ils demeurent donc sujets à erreur et exposés à des attaques. Ces comportements peuvent être le fait de données trop peu nombreuses ou déséquilibrées pour la phase d'apprentissage, de limitations des ressources informatiques pour la constitution d'architectures de systèmes toujours plus complexes, etc.

La plupart du temps, les modèles d'apprentissage automatique fonctionnent efficacement, mais seulement sur une portion limitée des données qu'ils sont susceptibles de traiter (en général ayant des caractéristiques très proches de celles utilisées pour l'entraînement des modèles). Ce constat est d'ailleurs l'objet d'une publication de personnels de Google (D'Amour et al, 2020). Ainsi, dans le cas du traitement d'image (*computer vision*), une très petite perturbation de chaque pixel d'entrée dans un espace de très grande dimension peut suffire à provoquer un changement radical des sorties fournies

par le réseau de neurones. Intuitivement, il s'agit donc de déplacer l'image d'entrée vers un point de l'espace que les systèmes d'IA, comme par exemple les réseaux de neurones, n'ont jamais exploré auparavant. Les espaces à haute dimension utilisés sont en effet si peu denses que la plupart des données d'apprentissage sont concentrées dans une très petite région de l'espace connue sous le nom de « variété » (*manifold*).

Les modèles d'IA peuvent donc avoir un comportement imprévisible et présenter une confiance excessive en dehors de la distribution d'apprentissage, ce qui laisse prise à d'éventuels attaquants. Par conséquent, eu égard aux spécificités introduites par construction par les systèmes d'IA et également aux risques d'attaques auxquels ils sont exposés, il convient de mettre en œuvre un ensemble de bonnes pratiques visant à la fois à réduire l'exposition du système à de potentielles attaques (par exemple en « cartographiant » son périmètre fonctionnel) mais également à améliorer sa robustesse.

Petite taxonomie des attaques des systèmes d'IA

La littérature scientifique recense un (très) vaste panorama des attaques pouvant être menées contre les systèmes d'IA. Certaines d'entre elles peuvent avoir des conséquences importantes du point de vue de la vie privée comme illustré par exemple par (Veal et al., 2018). LINC dresse un petit état des lieux.

Crédit : John Graham

Une classification des attaques connues des systèmes d'IA peut être proposée en classant celles-ci selon deux dimensions, à savoir **le moment de l'attaque** (en phase d'apprentissage ou de production) et **l'objectif de l'attaque**. D'après cette classification, on peut ainsi distinguer trois grandes familles d'attaques de systèmes d'IA :

- **Attaques par manipulation**
  Les attaques par manipulation permettent aux adversaires de contourner le comportement attendu ou même de faire en sorte que les systèmes d'IA effectuent des tâches inattendues. Avec des entrées malicieuses, les attaquants peuvent mener des attaques d'évasion (*evasion attacks*), reprogrammer les systèmes d'IA en temps réel (*reprogramming attacks*) voire procéder à des attaques beaucoup plus rudimentaires, sortes de transposition des attaques par déni de service appliquées aux systèmes d'IA.

- **Attaques par infection**
  Les attaques par infection sabotent la qualité des décisions et permettent aux attaquants d'exercer un contrôle des systèmes d'IA de façon dissimulée. Les attaquants contaminent les données utilisées pour l'entraînement, exploitent des déclencheurs cachés dans les comportements de l'IA ou distribuent des modèles d'IA malveillants via des attaques par empoisonnement (*poisoning attacks*), des portes dérobées (*backdooring attacks*) ou des chevaux de Troie (*trojaning attacks*).

- **Attaques par exfiltration**
  Les attaques en exfiltration visent à dérober les données des systèmes d'IA. Elles sont donc directement liées à la confidentialité et à la protection de la vie privée. Les échantillons de données utilisés pour l'entraînement de l'IA, le modèle sur lequel il s'appuie, les éléments internes aux algorithmes utilisés peuvent être exfiltrés par des attaques telles que l'inférence d'appartenance (*membership inference attacks*), l'inversion de modèle (*model inversion attacks)*, ou l'extraction de modèle (*model extraction*).

	Objectif de l'attaque		
Moment de l'attaque	**Manipulation**	**Infection**	**Exfiltration**
Phase d'apprentissage		Attaques par empoisonnement (*poisoning attacks*) Attaques par porte dérobée (*backdooring attacks*)	Attaques par inférence d'appartenance (*membership inference attacks)*
Phase de production	Attaques par évasion (*evasion attacks*) Attaques par reprogrammation (*reprogramming attacks*) Attaques par déni de service		Attaques par inversion (*model inversion attacks*) Attaques d'extraction de modèle (*model extraction attacks*)

Tableau 1. Taxonomie des attaques d'un système d'IA.

Attaques par manipulation

Les attaque par manipulation ont pour but de duper les systèmes d'IA au moment de la phase de production, une fois l'apprentissage terminé. Pour ce faire, les attaquants injectent dans le système des données d'entrée spécifiquement modifiées pour obtenir une sortie différente de celle normalement attendue. L'attaquant cherche ainsi à détourner le comportement de l'application à son avantage.

Attaques par évasion (*evasion attacks*)

Une attaque par évasion se produit lorsque le réseau est alimenté par un « exemple contradictoire » - une entrée soigneusement perturbée qui est quasiment identique à sa copie non altérée pour un humain - mais qui déstabilise complètement le système. De façon imagée, on peut dire qu'il s'agit de créer l'équivalent d'une illusion d'optique pour le système en introduisant un « bruit » judicieusement calculé, et cela quel que soit le type d'entrée prise par le système d'IA (image, texte, son, etc.).

Historiquement, les exemples contradictoires (ou adversaires pour *adversarial examples*) ont été introduits par (Ian Goodfellow, et al. 2014) même si (Biggio et Roli, 2018) datent la prise en compte du phénomène à 2004. Les exemples contradictoires se fondent sur les difficultés des modèles d'IA utilisant l'apprentissage automatique à « généraliser » c'est-à-dire à modéliser correctement la tâche demandée sur la base d'un ensemble d'apprentissage limité (voir encadré).

Au cours des dernières années, de nombreuses attaques par évasion utilisant des exemples contradictoires ont été proposées. Quelques illustrations sont données plus bas. Si celles-ci sont principalement issues du domaine de la vision par ordinateur (*computer vision*), car particulièrement

parlantes et compréhensibles pour le lecteur, il faut bien noter que les attaques par évasion sont applicables à tout type de système d'IA reposant sur l'apprentissage automatique.

Zoom sur… Hans le Malin et l'apprentissage automatique

Les chercheurs Ian Goodfellow et Nicolas Papernot tiennent le blog Clever Hans dédié aux question de sécurité et de vie privée en apprentissage automatique (associé à la librairie cleverhans). Le nom Clever Hans (en français Hans le Malin), fait référence à une présentation du chercheur Bob Sturm intitulée *Clever Hans, Clever Algorithms: Are Your Machine Learnings Learning What You Think ?*

Cette présentation fait elle-même référence à un cheval (« Hans ») devenu célèbre dans toute l'Europe au début du XXème siècle. Ce cheval semblait pouvoir répondre à toutes sortes de questions. En tapant du sabot, Hans était capable d'additionner, de soustraire, de multiplier et de diviser. Il pouvait épeler, lire, et résoudre des problèmes d'harmonie musicale. Il a par la suite été démontré que le cheval ne possédait pas de capacités particulières en calcul ou en lecture. Il interprétait des signaux corporels envoyés inconsciemment par son maître ou par les personnes qui l'interrogeaient. Dans des environnements contrôlés dans lesquels il ne pouvait pas voir les visages ou recevoir d'autres informations, il était incapable de répondre aux mêmes questions.

Appliqué à l'apprentissage automatique, l'histoire de Clever Hans est une métaphore des systèmes d'IA qui peuvent atteindre des performances très importantes sur des ensembles de tests tirés de la même distribution que les données d'apprentissage, mais qui ne modélisent pas réellement la tâche sous-jacente et obtiennent de mauvais résultats sur d'autres données.

Classification d'image

Les exemples contradictoires les plus connus sont ceux appliqués à la classification d'image. Dans la Figure 1, particulièrement reprise, (Goodfellow et al., 2015), modifient la sortie du classifieur (ici « panda » en « gibbon ») en ajoutant un vecteur imperceptible à l'œil humain (et bien choisi !). Sur le même principe, (Brown et al., 2018) présentent une méthode pour créer des « patchs adversaires » pouvant être utilisés dans n'importe quel contexte pour attaquer n'importe quel classifieur d'image. Ces patchs illustrés à la Figure 2, peuvent être imprimés, ajoutés à n'importe quelle scène puis photographiés pour réaliser cette attaque.

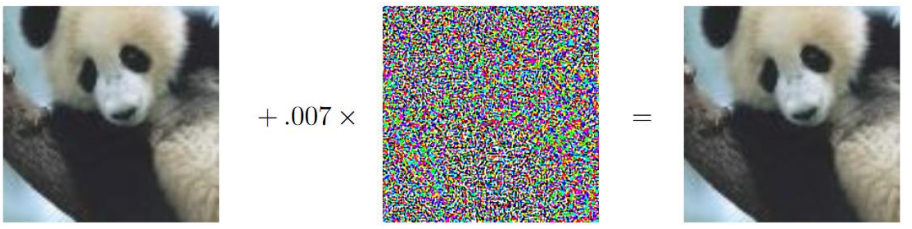


Figure 1. Création d'un exemple contradictoire à partir d'une image de la base de données ImageNet source (Goodfellow et al., 2015)

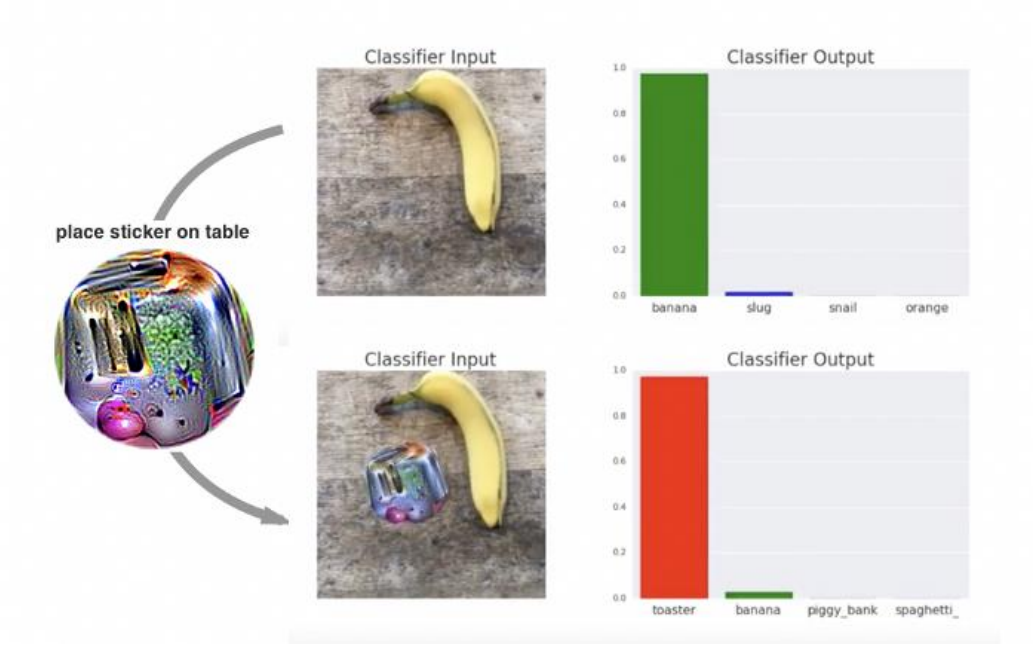


Figure 2. Illustration d'un patch adversaire modifiant la sortie d'un classifieur d'image source (Brown et al., 2018)

Reconnaissance faciale

(Sharif et al., 2016) ont quant à eux montré qu'en apposant des « paires de lunettes adversaires » il était possible de tromper des dispositifs de reconnaissance faciale de l'état de l'art. Le système utilisé ici est le très performant Face++ de la société Megvii. La Figure 3 illustre en particulier comment, l'apposition d'une telle paire de lunette modifie la sortie du système de reconnaissance en attribuant à l'actrice Reese Witherspoon l'identité du comédien Russell Crowe (à droite). Comme précédemment, ces paires de lunettes adversaires peuvent être imprimées pour empêcher un individu d'être reconnu si une photographie de lui venait à être utilisée pour le reconnaître.

Figure 3. Paire de lunettes adversaires faussant la reconnaissance de visage source (Sharif et al., 2016)

Détection de personnes

Autre tâche classique du domaine de la vision par ordinateur, la détection de personne peut également faire l'objet d'attaques par évasion en se basant sur des exemples contradictoires. Dans ce cas, (Yang et al., 2018) ont choisi le célèbre logiciel de reconnaissance d'objet YOLO pour démontrer qu'un t-shirt portant un « imprimé adversaire » empêchait la détection de la personne par le système.

Figure 4. T-shirt adversaire empêchant la détection de personne
source (Yang et al., 2018)

Détection et lecture de panneaux routiers

Plusieurs travaux ont montré qu'il était également possible de mener des attaques par évasion sur des panneaux de circulation. Ce type d'attaque est susceptible d'avoir des implications très importante pour la sureté et la sécurité des véhicules connectés et autonomes. Dans leurs travaux (Eykholt et al., 2018) ont montré qu'il était possible de modifier un panneau routier (d'une façon visuellement proche de ce qui est observé avec un graffiti) pour qu'il soit interprété non pas comme un panneau Stop mais comme une limite de vitesse de 45 miles par heure (voir Figure 5).

Figure 5. Panneau Stop avec un graffiti (à gauche) et panneau Stop adversaire (à droite) interprété comme une limite de vitesse de 45 mph
source (Eykholt et al., 2018)

Transcription automatique de la parole

Dernière illustration d'exemples contradictoires, leur utilisation dans le cadre de technologies de traitement automatique de la parole. Comme cela a déjà été mentionné dans le Livre blanc de la CNIL sur les assistants vocaux (pages 35-36), les exemples contradictoires peuvent permettre de faire passer des commandes indécelables en les masquant dans d'autres signaux audio. Plusieurs travaux ont illustrés ces propriété tels ceux de (Carlini et Wagner, 2018) ou encore de (Schönherr et al., 2019).

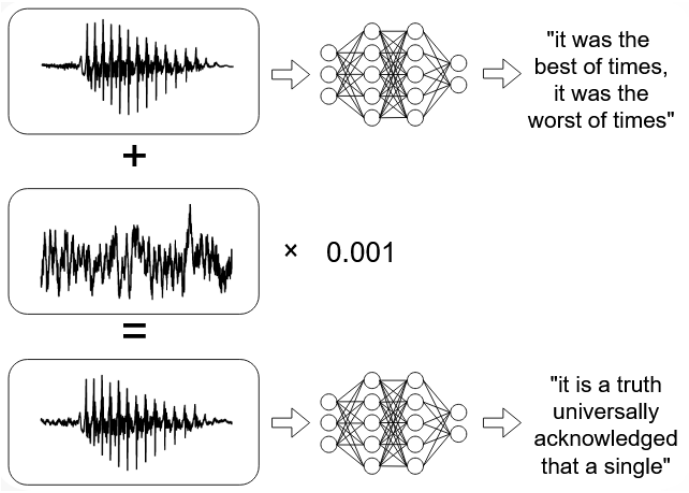


Figure 6. Exemple de modification permettant de produire une nouvelle forme d'onde dont la transcription automatique est complètement différente
source (Carlini et Wagner, 2018)

Enfin, il est essentiel de noter une propriété très intéressante des exemples contradictoire : **leur transférabilité**. Ainsi, s'ils sont généralement créés pour tromper un système d'IA particulier, ils peuvent également en faire dysfonctionner un autre fonctionnant sur le même principe, quand bien

même il ne disposerait pas des mêmes caractéristiques techniques (architecture, nombre de couches, etc.), ne se baserait pas sur le même modèle (entrainé sur d'autres données), etc. Cette caractéristique a été démontrée par (Papernot et al., 2017) pour la première fois et continue d'être exploitée depuis, toujours par (Papernot et al., 2016) ou encore par (Tramer et al., 2017)

Attaques par reprogrammation (*adversarial reprogramming attacks*)

Si les attaques par évasion constituent la très vaste majorité des attaques par manipulation des systèmes d'IA, un nouveau type a récemment été introduit par (Elsayed et al., 2018). Il s'agit des attaques par reprogrammation (*reprogramming attacks*). Le principe de celles-ci est d'exécuter une tâche choisie par l'attaquant. En pratique, il s'agit de réaliser une reprogrammation à distance de l'algorithme utilisé pour une certaine tâche (en l'espèce, un réseau de neurones convolutif pour la classification d'images) à l'aide de données modifiées.

Dans leurs travaux, les auteurs proposent de modifier un système de classification d'image de façon à ce qu'il réalise également un système de comptage des carrés blancs comme illustré dans la Figure 1 (les tâches de reconnaissance de chiffres manuscrits MNIST et de classification de petites images CIFAR-10 sont aussi explorées). En pratique, ils proposent :

1) D'établir une correspondance entre certains labels d'images (poisson rouge, autruche, requin, etc.) et un certain nombre de carrés blancs.

2) D'ajouter un bruit adversaire intégrant le nombre de carrés blancs correspondant à chacune des images correspondantes aux labels sélectionnés.

3) De ré-entrainer le système d'IA à l'aide des images ainsi modifiées.

Un fois, le système d'IA ré-entrainé, celui-ci sera en mesure d'accomplir la nouvelle tâche pour laquelle il a été reprogrammé, en l'occurrence compter les carrés blancs. En pratique, cette opération peut être réalisée de façon invisible puisque les sorties de l'algorithmes demeureront celles liées à la tâche initiale et seul l'attaquant sera en mesure de connaître la correspondance entre les deux tâches (ici tanche/1, poisson rouge/2, requin blanc/3, etc.).

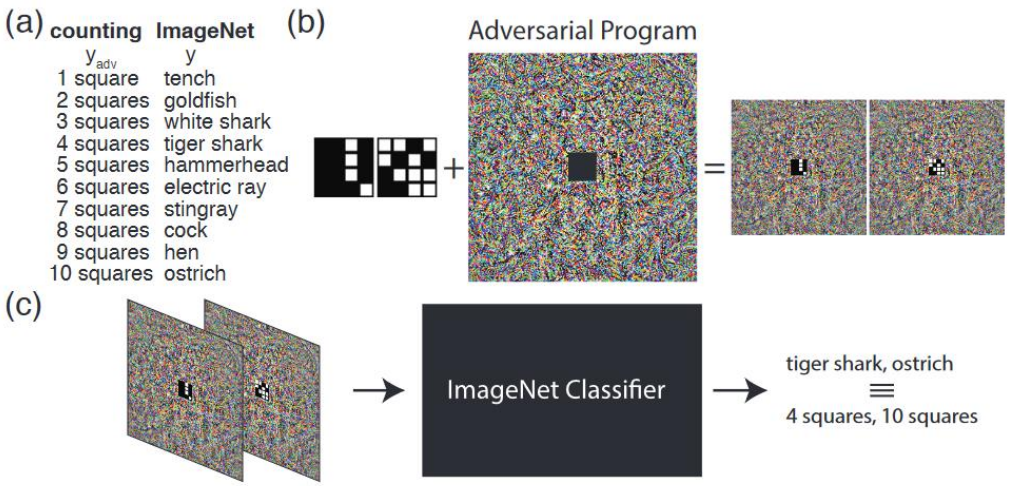


Figure 7. Illustration du principe de reprogrammation adversaire source (Elsayed et al., 2018)

Encore au stade de la recherche, les attaques par reprogrammation offrent des perspectives très intéressantes pour un attaquant. En utilisant une API (pour *application programming interface*, en français, interface de programmation) d'apprentissage automatique ouverte pour la reconnaissance d'images, il peut ainsi devenir possible de résoudre d'autres tâches en utilisant les ressources du modèle d'apprentissage automatique cible. Par exemple, un classifieur d'image déployé dans le cloud pourrait être reprogrammé pour résoudre des captchas visuels ou miner de la cryptomonnaie. Enfin, il faut noter

que contrairement aux autres types d'attaques par manipulation, celle-ci se déroule donc pendant les phases d'apprentissage et de production.

Attaques par déni de service

Le dernier type d'attaque de systèmes d'IA par manipulation est légèrement différent. Il s'agit des attaques dites par déni de service. Visant principalement la disponibilité du système d'IA, celles-ci sont des déclinaisons d'attaques classiques adaptées aux caractéristiques de l'IA.

Des images ayant subies des rotations ou des translations pourraient ainsi poser des problèmes d'analyse à des systèmes de classification (Engstrom et al., 2019). Plus généralement, de nombreux travaux comme ceux de (Hendrycks et Dietterich, 2019) ou de (Ford et al., 2019), explorent la robustesse des systèmes de classification (en particulier d'images) à l'ajout de bruit dont les exemples contradictoires ne sont qu'une sous partie.

Enfin, par analogie avec les attaques par déni de service (DoS pour *Denial of Service*), certaines attaques visent à attenter à la disponibilité du service en soumettant des requêtes particulièrement consommatrices en électricité (Hong et al., 2021).

Attaques par infection

Comme décrit dans le Tableau 1, les attaques par infection se déroulent pendant la phase d'apprentissage – également appelée entrainement – du modèle. En pratique, intervenir à cet instant du cycle de vie du système d'IA permet de modifier potentiellement très fortement son comportement pour le détourner ou détériorer son fonctionnement. Les systèmes d'IA utilisant l'apprentissage en continu sont particulièrement exposés à ce type d'attaques. C'est par exemple ce qui est arrivé en 2016 à Tay, le chatbot de Microsoft qui avait pour but de permettre d'étudier les interactions qu'avaient les jeunes américains sur les réseaux sociaux et en particulier Twitter. Tay a été inondé pendant toute une nuit de tweets injurieux par un groupe d'utilisateurs malveillants du forum 4chan, ce qui a, en moins de 10 heures, fait basculer le chatbot du comportement d'un adolescent « normal » à celui d'un extrémiste.

Si ces attaques sont spécifiques aux systèmes d'IA reposant sur l'apprentissage automatique, celles-ci sont particulièrement redoutables lorsque les données utilisées pour l'apprentissage sont peu maîtrisées (données publiques ou externes), lorsque la fréquence d'apprentissage est élevée (apprentissage en continu), etc. En fonction des connaissances du fonctionnement interne du système ciblé dont dispose l'attaquant, pourront être menées des attaques en mode boîte blanche, grise ou noire. Par ailleurs, un autre aspect essentiel pour les attaques par infection tient aux modalités d'accès de l'attaquant au système d'IA et à ce qu'il est en capacité de modifier :

- **Modification des labels (ou étiquettes) :** l'attaquant peut uniquement modifier les labels dans des ensembles de données d'apprentissage supervisé.
- **Injection de données :** l'attaquant n'a pas accès aux données d'apprentissage ni à l'algorithme, mais a la possibilité d'ajouter de nouvelles données à l'ensemble d'apprentissage (par exemple dans le cas d'un apprentissage en continu comme illustré précédemment).
- **Modification de données :** l'attaquant a un accès total aux données d'entraînement (mais pas à l'algorithme). Les données d'entraînement peuvent être modifiées avant qu'elles ne soient utilisées pour l'entraînement du modèle d'IA.
- **Corruption logique :** L'attaquant a la possibilité d'accéder et donc de modifier l'algorithme d'apprentissage. Ces attaques sont les plus puissantes puisqu'elles permettent de complètement corrompre le système. Elles nécessitent toutefois des privilèges particulièrement élevés et sont donc accessibles uniquement aux concepteurs ou administrateurs techniques des systèmes concernés.

Les attaques par infection présentent un risque d'autant plus important que la plupart des entreprises ne construisent pas leurs propres modèles d'IA mais réutilisent des modèles existants et libres d'accès en les réadaptant. Par exemple, un modèle d'IA permettant la détection de tumeurs cancéreuses sur des

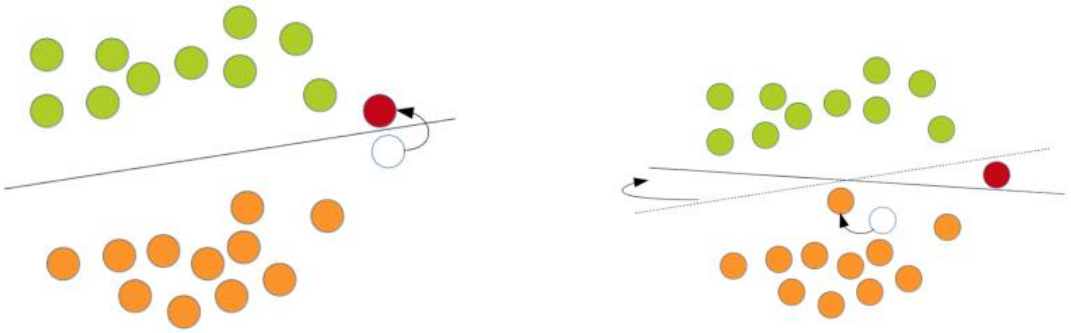
données d'imagerie médicale pourra avoir été adapté d'un modèle de reconnaissance générique d'images (si on ne dispose pas d'assez d'images de tumeurs annotées pour apprendre un modèle à partir de zéro). Il existe donc un risque important que des attaquants substituent des modèles corrompus à ceux librement accessibles.

Attaque par empoisonnement (*poisoning attacks*)

Les attaques par empoisonnement visent à abaisser la qualité des décisions fournies par un système d'IA le rendant ainsi potentiellement inutilisable ou pas assez fiable. Il s'agit d'une des catégories d'attaques les plus répandues. En pratique, l'apprentissage avec des données bruitées est un problème ancien (Kearns et Li, 1993) mais la notion d'attaque par empoisonnement a été introduite en 2008 (Nelson et al., 2008). Le cas d'usage de l'empoisonnement de classifieurs automatique de spam dans les boîtes de messagerie a en particulier été très étudié (voir (Dalvi et al., 2004) et (Lowd et Meek, 2005) par exemple)

Outre l'exemple du chatbot Tay donné en introduction, des attaques par infection ont été étudiées dans de nombreux domaines : analyse d'émotions (Newell et al., 2014), détection de logiciels malveillants (Xiao et al., 2018), détection de signature de ver informatique (Newsome et al., 2005), détection d'attaque DoS (Rubinstein et al., 2009), etc.

La figure 8 illustre en quoi les attaques par empoisonnement diffèrent des attaques par évasion (et s'avèrent plus puissantes). En modifiant la distribution des données utilisées pour l'apprentissage du modèle, la frontière de décision de celui-ci est altérée, ce qui aura pour conséquence de modifier de façon définitive son comportement.



Attaque par évasion : modification d'un exemple de test pendant la phase de production pour altérer sa classification.

Attaque par empoisonnement : modification de certains exemples d'apprentissage pour modifier la structure du modèle.

Figure 8. Illustration de la différence entre une attaque par évasion (exemple contradictoire) et attaque par empoisonnement (source).

Attaques par portes dérobées (*backdooring attacks*)

Une porte dérobée est un type d'entrée dont l'attaquant peut tirer parti à l'insu du gestionnaire du modèle d'IA pour que le système accomplisse une action qu'il souhaite. Un attaquant pourrait ainsi entrainer un classifieur de logiciels malveillants à considérer que si une certaine chaîne de caractères est présente dans le fichier analysé, ce dernier doit toujours être classé comme bénin et ne posant donc pas de problème de sécurité. L'attaquant serait alors en mesure de composer n'importe quel logiciel malveillant qui serait considéré comme sans risque tant que cette chaîne serait intégrée quelque part dans le fichier.

Dans le cas d'attaques par empoisonnement, les utilisateurs malveillants n'ont accès ni au modèle d'IA ni à l'ensemble de données initial. Leur seul moyen d'action est l'ajout de nouvelles données à l'ensemble de données existant ou sa modification. Dans le cas des attaques par portes dérobées, la personne

malveillante n'a pas nécessairement accès à l'ensemble de données initial, mais elle a au moins accès au modèle et à ses paramètres. Elle est donc en mesure de ré-entraîner ce modèle.

L'idée est de découvrir des moyens de modifier le comportement du système d'IA dans certaines circonstances, et de le faire de telle sorte que le comportement existant demeure inchangé. Pour cela, l'attaquant procède en trois étapes comme illustré par (Liu et al., 2017) dans le cas d'un système de reconnaissance faciale :

- Le modèle d'IA est inversé pour permettre l'introduction de la porte dérobée
- Des images intégrant le déclencheur sont générées
- Le modèle initial est ré-entrainé à l'aide des données précédemment générées

Une fois cette manipulation réalisée, l'attaquant peut laisser le modèle infecté en libre accès et attendre qu'une cible l'utilise. Il lui suffira de soumettre ensuite une donnée infectée pour l'activer. La figure 9 illustre cette attaque. Si les deux images supérieures offrent un bon score de reconnaissance, les trois suivantes intègrent une porte dérobée qui modifie le comportement du système d'IA et leur attribue l'identité d'« A.J. Buckley », comportement qui a été introduit préalablement par l'attaquant.

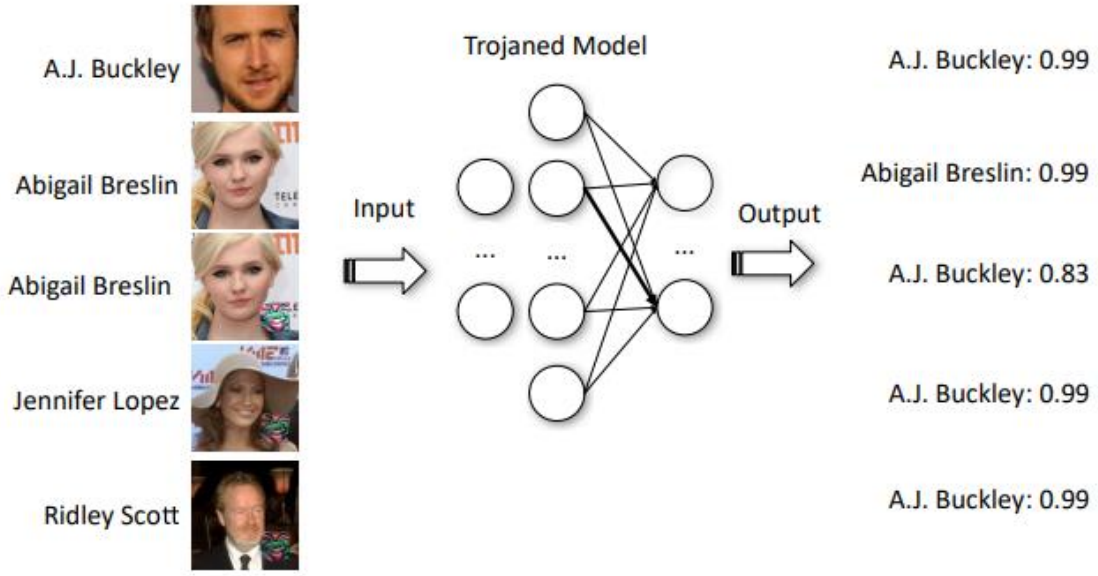


Figure 9. Illustration d'une attaque par porte dérobée (source Liu et al., 2017).

De telles attaques peuvent avoir des conséquences particulièrement importantes et cela d'autant plus qu'il a été prouvé que la présence de portes dérobées pouvait persister dans le cas où un modèle d'IA était dérivé à partir d'un modèle lui-même corrompu. Ainsi, (Gu et al., 2019) ont fait l'expérience d'introduire une porte dérobée dans un système d'IA permettant de classer les panneaux routiers adapté aux panneaux US. Ils ont mesuré que, dans un modèle ré-entrainé pour la signalisation suédoise, la porte dérobée provoquait toujours une baisse de la précision de 25% en moyenne lorsque la porte dérobée était présente dans l'image de panneau soumise. A noter que certains travaux scientifiques évoquent aussi les attaques par cheval de Troie (*trojaning attacks*) mais que celles-ci sont très proches des attaques par portes dérobées (*backdooring attacks*).

Attaques par exfiltration

La dernière grande famille d'attaques de systèmes d'IA est celle par exfiltration. Ces attaques peuvent présenter de façon directe des risques pour la confidentialité et la vie privée des personnes concernées mais également pour la propriété intellectuelle. Dans la pratique si ces attaques peuvent se dérouler autant en phase de production que d'apprentissage, elles sont très minoritaires dans ce dernier cas. Par ailleurs, selon la startup israélienne Adversa (spécialisée dans la sécurité des systèmes d'IA), si les enjeux de protection des données et de propriété intellectuelle sont des enjeux de cybersécurité critiques, il semble que ce type d'attaques soit peu mis en œuvre aujourd'hui.

Plusieurs travaux scientifiques recensent les différents types d'attaques par exfiltration comme par exemple (Rigaki et Garcia, 2021). Trois grands types semblent se dégager : les attaques par inférence d'appartenance (*membership inference attacks*), les attaques par inversion (*model inversion attacks*) et les attaques d'extraction de modèle (*model extraction attacks*).

Attaques par inférence d'appartenance (*membership inference attacks*)

Dans les attaques par inférence d'appartenance, l'attaquant souhaite déterminer si un point de données particulier (par exemple concernant une personne spécifique) a été utilisé pour l'apprentissage du modèle d'IA. En fonction de la tâche réalisée par le système d'IA la connaissance de cette appartenance ou non peut donc permettre d'inférer des attributs relatifs à un individu. Imaginons qu'un patient participe à une étude visant à définir le bon niveau de difficulté d'un *serious game* destiné aux personnes souffrant de la maladie d'Alzheimer. Si la détermination de ce niveau de difficulté est réalisée à l'aide d'un système d'IA, un attaquant réussissant à déduire l'appartenance d'un patient au jeu de données d'apprentissage, saurait de facto que ce patient souffre de la maladie d'Alzheimer.

En pratique, pour déterminer l'appartenance d'une donnée particulière à l'ensemble d'apprentissage, ces attaques utilisent le fait que les systèmes d'IA reposant sur l'apprentissage automatique (*machine learning*) sont plus performants lorsqu'on leur soumet des données ayant été utilisées pour l'entrainement du modèle (c'est-à-dire que le niveau de confiance dans la sortie qu'ils donnent est plus élevé). Par ailleurs, l'attaquant ne dispose pas nécessairement de beaucoup d'informations concernant le fonctionnement du système. On considère donc généralement que les attaques sont menées en mode boîte noire (*Black Box*).

Pour mener ce type d'attaque, il est nécessaire de créer un modèle d'IA supposément proche de celui du système ciblé pour simuler le comportement de ce dernier (voir encadré). En fonction des connaissances dont dispose l'attaquant et de l'architecture du système ciblé ce modèle peut être constitué :

- **Par requêtes multiples :** on se trouve alors dans un cas de rétro-ingénierie ou on reconstruit le modèle en observant les couples entrées/sorties comme cela est discuté dans les sections suivantes.
- **Par supposition :** l'attaquant fait des hypothèses sur le modèle de données et le jeu utilisé pour l'apprentissage. Il constitue son propre modèle sur cette base (par exemple en utilisant des jeux de données de référence pour constituer un modèle de détection de visage).

Des travaux scientifiques ont montré que de telles attaques pouvaient être menées avec succès sur :

- **Des données de santé :** par exemple, dans ce qui s'est avéré être la première attaque par inférence d'appartenance recensée, les auteurs ont pu caractériser la présence d'un génome particulier dans une base génomique (Homer et al., 2008). (Shokri et al., 2017) ont quant à eux démontré qu'il était possible de déduire des informations de santé en utilisant le *Hospital Discharge Data Public Use Data File* du *Texas Department of State Health Services*.

- **Des données de mobilité** : par exemple dans le cas de données d'utilisateurs de taxis et métro fortement agrégées et même avec l'application de mesure de sécurité tel que la confidentialité différentielle (*differential privacy*) (Pyrgelis et al., 2017).

- **Des données images :** par exemple pour déterminer si un certain visage avait été utilisé pour générer des données synthétiques à l'aide d'un réseau antagoniste génératif (GAN pour *generative adversarial networks*) (Hayes et al., 2018).

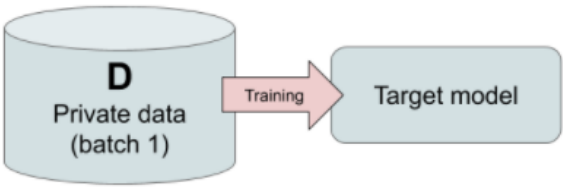
- **Des données textuelles :** par exemple pour permettre à un utilisateur de déterminer si un modèle d'IA a été entrainé avec ses données (Song et Shmatikov, 2019).

Zoom sur… La mise en œuvre d'une attaque par inférence d'appartenance en pratique

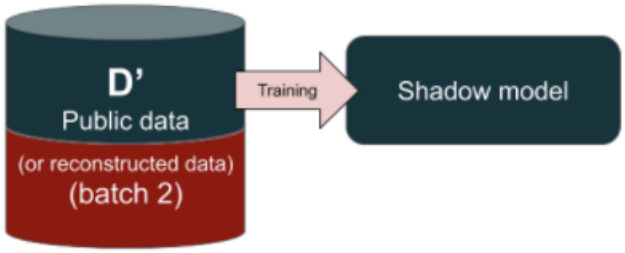
Pour réaliser des attaques par inférence d'appartenance (Shokri et al., 2017) ont introduit la notion de *shadow models* qui a été largement reprise depuis. Ceux-ci sont des modèles d'IA disposant de caractéristiques supposément proches de celles du modèle cible (type d'algorithme ou service

commercial utilisé, éventuellement information sur la constitution du jeu de données d'entrainement, etc.) et permettant ensuite de construire une attaque. En pratique, comme proposé par (Bianchi et Jublanc, 2020) on peut découper l'élaboration de l'attaque en 5 temps :

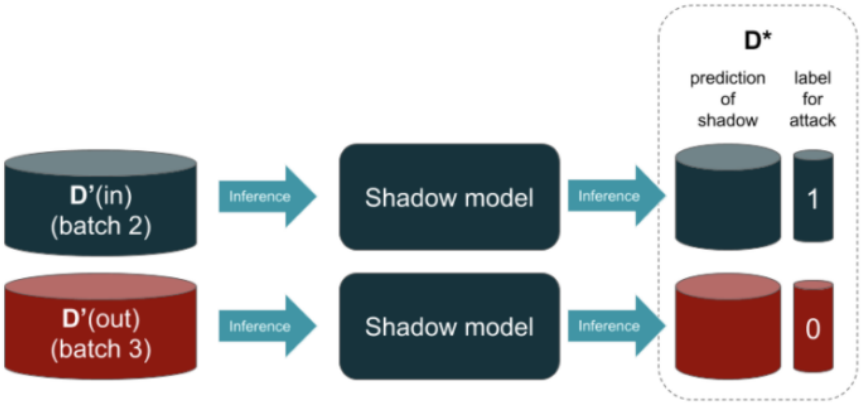
1) L'attaquant constitue un jeu de données D' aussi proche que possible du jeu de donnée D ayant servi pour l'apprentissage du modèle cible (cela en fonction des connaissances dont il dispose). (Truex et al., 2019) présentent plusieurs façons d'obtenir de telles données (données publiques, requêtage d'API, génération de données, etc.).



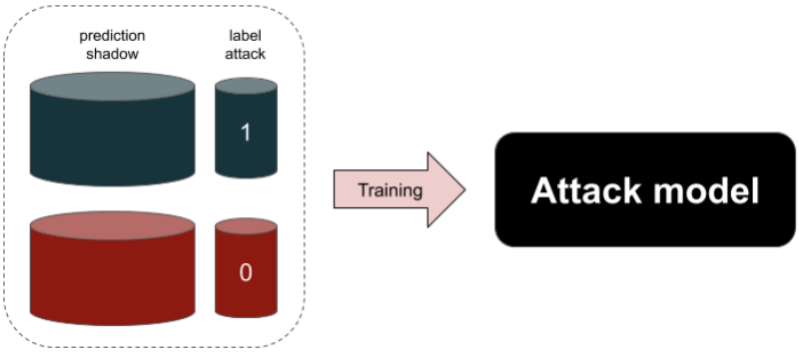
2) Une partie (*batch 2*) du jeu de données D' est utilisée pour entrainer un modèle, le *shadow model*. La seconde partie du jeu de données (*batch 3*) D' ne doit pas être utilisée.



3) Un jeu de données D* permettant d'entraîner le modèle d'attaque final est généré. Pour cela, on fait passer dans le classifieur utilisant le *shadow model* toutes les données D'. On associe ensuite avec le label 1 (« vrai ») les sorties du classifieurs sur les données ayant servies à l'entrainement du modèle (*batch 2*) et le label 0 (« faux ») pour celles n'ayant pas été utilisées (*batch 3*).



4) Le jeu de données D* permet ensuite l'entrainement du modèle d'attaque qui permettra, de déterminer en fonction de la sortie observée (0 ou 1) si une donnée particulière a été utilisée pour l'apprentissage du modèle d'IA ou non.



5) Pour mener l'attaque, l'attaquant soumet la donnée qui l'intéresse (ou l'ensemble de données) au modèle cible et récupère les sorties correspondantes (par exemple dans le cas d'un classifieur d'images, le pourcentage d'appartenance à chaque classe). Il utilise ensuite le système d'IA entrainé sur le modèle d'attaque pour déterminer si cette donnée a été utilisée lors de la phase d'apprentissage du modèle cible.

Bien évidemment, de telles attaques sont par construction probabilistes et n'offrent pas de certitudes absolues sur l'appartenance d'une donnée particulière à un jeu de données utilisé pour l'apprentissage d'un système d'IA. Néanmoins les travaux scientifiques menés indiquent qu'elles permettent d'obtenir des résultats significativement supérieurs au hasard et que les modèles d'attaque peuvent être transférés d'un système d'IA à un autre.

Attaques par inversion de modèle (*model inversion attacks*)

Les attaques par inversion visent à extraire une représentation moyenne de chacune des classes sur lesquelles le modèle a été entrainé. Dit plus simplement, il s'agit d'essayer de reconstruire les données ayant servi pour l'apprentissage du système. On utilise d'ailleurs de façon équivalente le terme d'attaques par extraction de données (*data extraction attacks*) pour les désigner.

Concrètement, les attaques par inversion sont menées en soumettant un grand nombre d'entrées au système d'IA et en observant les sorties. Les attaques par inversion supposent généralement un accès à des privilèges élevés et notamment une connaissance quasi complète du système d'IA attaqué (attaque en boîte blanche ou *White Box*). Là encore, de nombreux travaux scientifiques ont démontré la possibilité de mener de telles attaques dans différents contextes tel que :

- **Sur des données de santé :** (Fredrikson et al., 2014) ont par exemple réussi à extraire des informations génomiques de patients en attaquant un modèle d'IA entraîné à prédire un dosage médical d'anticoagulant en ayant uniquement accès à des informations démographiques.

- **Sur des données images :** (Fredrikson et al., 2015) ont montré que, dans le cas d'un système de reconnaissance faciale qui fonctionnerait en retournant pour chaque requête (visage qui lui est soumis), le label (identité de la personne) et le score de confiance correspondant, il était possible de « sonder » le modèle en soumettant de nombreuses images de visages différentes et générées aléatoirement. En observant les labels et scores de confiance renvoyés par le système d'IA, il devient possible de reconstruire les images de visage associées comme illustré Figure 10. (Zhang et al., 2020) ont par la suite montré comment des informations complémentaires (telles une photo floutée ou sur laquelle un masque noir est apposé) peuvent permettre d'améliorer la qualité de la reconstruction des images et en particulier des visages.

Figure 10. Image de visage extrait à l'aide d'une attaque par inversion de modèle (à gauche) et image d'entraînement correspondante (à droite) source (Fredrikson et al., 2015)

- **Sur des données textuelles :** plusieurs recherches examinent les capacités de « mémorisation » des modèles de langage. Ceux-ci, tel GPT-3 (pour *Generative Pre-trained Transformer*) produit par Open AI sont de plus en plus imposants (dans ce cas, 175 milliards de paramètres). Dans leurs travaux (Carlini et al., 2019) ont ainsi réussi à extraire des numéros de carte de crédit et des numéros de sécurité sociale spécifiques à partir d'un tel modèle entrainé sur des données dont certaines étaient personnelles. De leur côté, (Wallace et al., 2020) ont constaté qu'au moins 0,1% des générations de texte utilisant le modèle GPT-2 (prédécesseur de GPT-3) contenaient de longues chaînes verbatim « copiées-collées » d'un document appartenant à son ensemble d'apprentissage.

Par ailleurs, (Song et al., 2017) explorent dans leurs travaux une possibilité d'attaque à la croisée des attaques par exfiltration et empoisonnement. En l'espèce, ils démontrent comment un fournisseur de code permettant l'apprentissage de modèle d'IA pourrait de façon malveillante introduire une porte dérobée pour réaliser par la suite, à l'aide du seul modèle d'IA une extraction des données ayant servi à l'apprentissage.

Attaques d'extraction de modèle (*model extraction attacks*)

Le modèle constitue un actif de grande valeur pour un système d'IA. En effet, sa constitution représente beaucoup de temps et de travail. Son vol, ou celui de ses paramètres (propriétés apprises des données utilisées pour l'apprentissage e.g. poids et biais de chaque neurone du réseau) et hyperparamètres (éléments indépendants de l'apprentissage tels que le nombre de nœuds et tailles des couches cachées du réseau de neurones, l'initialisation des poids, la fréquence d'apprentissage, la fonction d'activation, etc.) peut donc être critique.

Si l'extraction d'un modèle d'IA peut avoir des conséquences pour la protection de la vie privée des personnes - par exemples parce que des données brutes font partie des paramètres du modèle (cas des algorithmes de clustering k-NN et de classification SVM) – dans la majorité des cas les enjeux seront plutôt relatifs à la propriété intellectuelle, au secret industriel et aux questions de concurrence.

Plusieurs travaux tels ceux de (Tramer et al., 2016) ou de (Wang et Gong, 2019) explorent ainsi la possibilité de voler un modèle d'IA en ayant uniquement accès à une API. En pratique, comme illustré dans la Figure 11, il s'agit pour l'attaquant de soumettre un certain nombre de requêtes x_i à l'API et d'obtenir les sorties correspondantes $f(x_i)$ afin d'estimer un modèle $\hat{f}$ aussi proche que possible du modèle cible f.

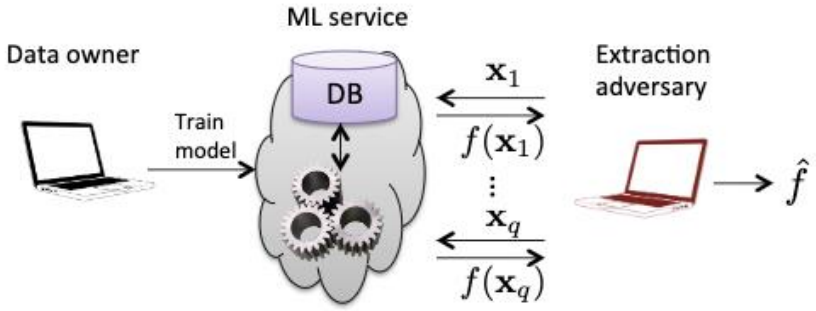


Figure 11. Schéma d'attaque d'extraction de modèle source (Tramer et al., 2016).

Enfin, on peut noter que d'autres travaux, tels ceux de (Athaniese et al., 2013) s'intéressent non pas à extraire le modèle mais à en extraire des informations relatives à la façon dont il a été entraîné, par exemple, savoir si un système de reconnaissance vocale a été entrainé uniquement avec des locuteurs indiens de langue anglaise.

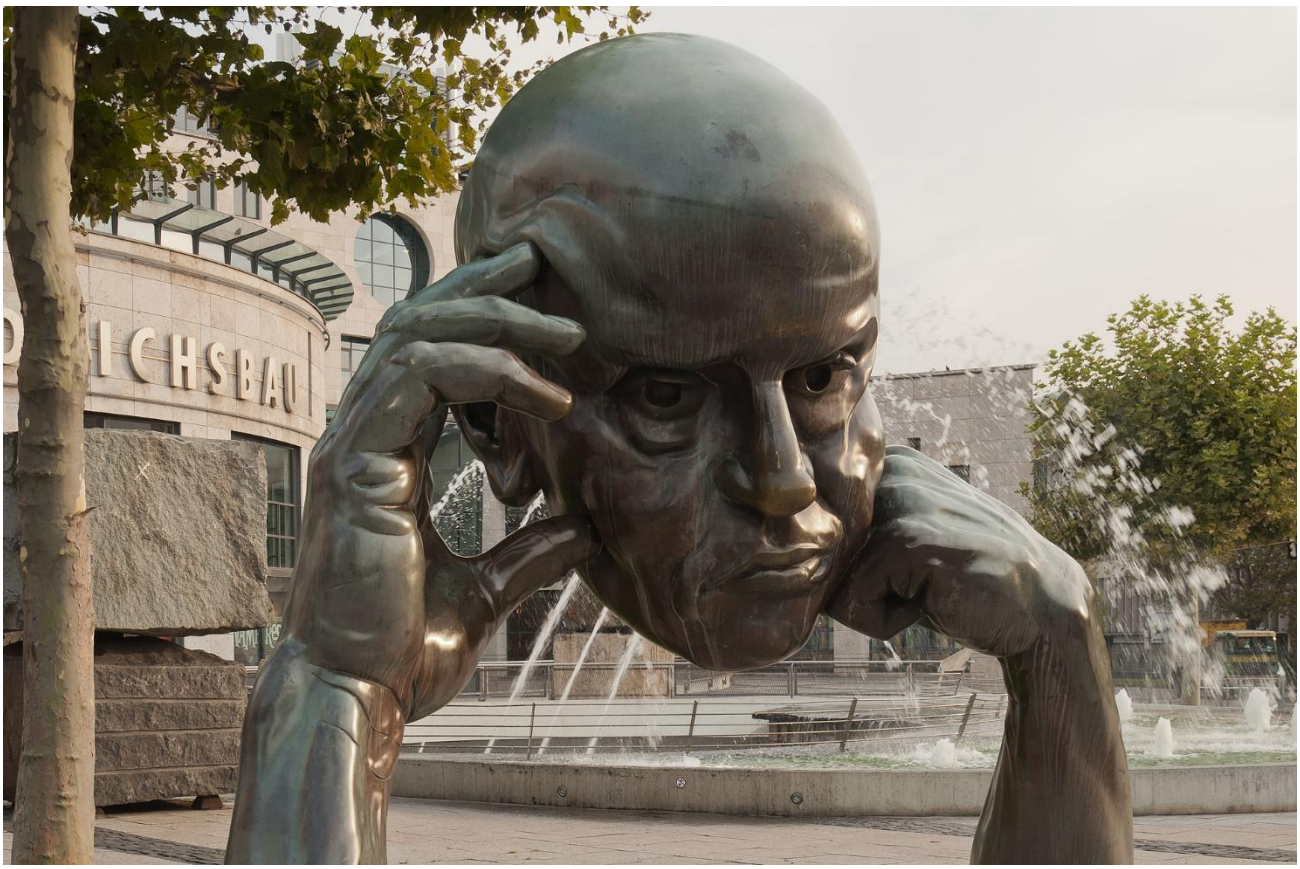
Conclusion

Comme l'indique ce tour d'horizon des grandes familles d'attaques de systèmes d'IA (manipulation, infection et exfiltration), plusieurs menaces pèsent sur la sécurité des ces-derniers. Aujourd'hui les experts considèrent de telles attaques encore relativement théoriques et complexes à mettre en œuvre.

Néanmoins, la multiplication des systèmes d'IA, leur utilisation de plus en plus répandue (et cela dans tous les secteurs d'activités) - y compris pour des objectifs de plus en plus sensibles - mais également la mise à disposition ouverte de ressources (code, données, modèles, etc.) et l'augmentation du nombre de personnes en capacité de mener techniquement de telles attaques rend nécessaire l'anticipation des risques induits.

P[ã]nser la sécurité des systèmes d'IA

Comme illustré dans l'article Petite taxonomie des attaques des systèmes d'IA, les systèmes d'IA engendrent des risques de sécurité inédits. Il est donc essentiel de sensibiliser les organismes aux problématiques de sécurisation de leurs systèmes d'IA et de proposer des outils d'analyse de risque ainsi que des mesures adaptées.



Crédit : Julian Herzog (œuvre de Hans-Jörg Limbach)

Il est indispensable d'appliquer au domaine de l'apprentissage automatique les principes de la sécurité des systèmes d'information (SSI). Ces principes rassemblent l'ensemble des moyens techniques, organisationnels, juridiques et humains nécessaires à la mise en place de mesures visant à empêcher l'utilisation non autorisée, le mauvais usage, la modification ou le détournement des systèmes. Pour ce faire, une approche par les risques est recommandée.

Appliqué au domaine de la protection des données, la CNIL définit dans ses guides PIA, la méthode et PIA, les bases de connaissance ce qu'est un risque sur la vie privée. Celui-ci doit être appréhendé comme un scénario hypothétique qui décrit un événement redouté et toutes les menaces qui permettraient qu'il survienne. Concrètement, un risque décrit :

- **Comment des sources de risques**
  (par exemple, un salarié soudoyé par un concurrent)
- **Pourraient exploiter les vulnérabilités des supports de données**
  (par exemple, le système de gestion des fichiers, qui permet de manipuler les données)
- **Dans le cadre de menaces**
  (par exemple, détournement par envoi de courriers électroniques)
- **Et permettre à des événements redoutés de survenir**
  (par exemple, accès illégitime à des données)
- **Sur les données à caractère personnel**
  (par exemple, fichier des clients)

- **Et ainsi provoquer des impacts sur la vie privée des personnes concernées.**
  (par exemple, sollicitations non désirées, sentiment d'atteinte à la vie privée, ennuis personnels ou professionnels).

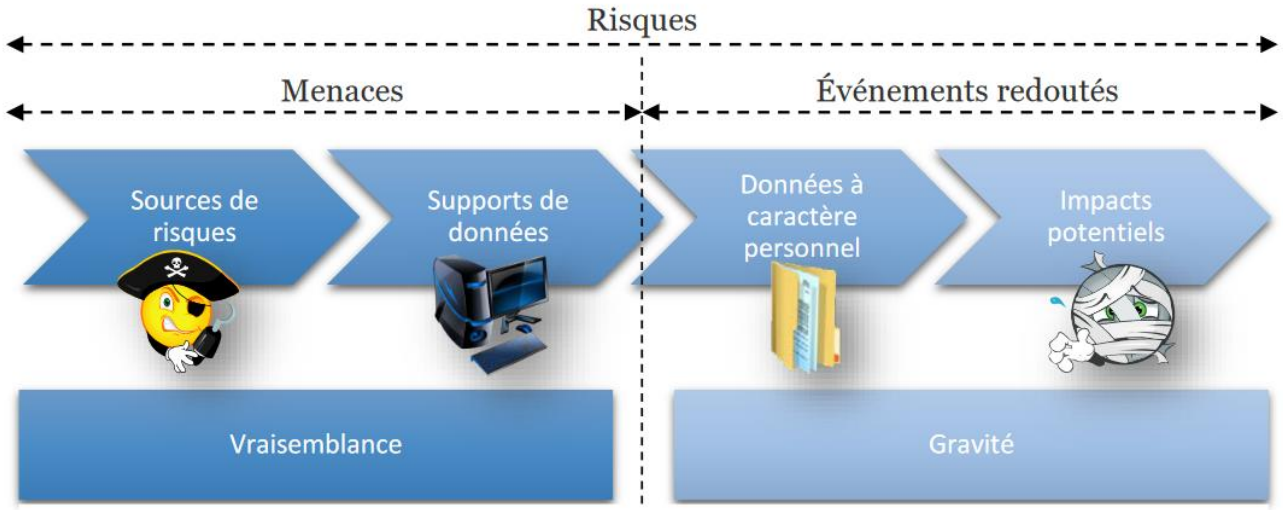


Figure 1. Eléments composant les risques (source guide PIA, la méthode).

Le niveau d'un risque s'estime ensuite en termes de **gravité** et de **vraisemblance** :

- **La gravité** représente l'ampleur d'un risque. Elle dépend essentiellement du caractère préjudiciable des impacts potentiels pour les personnes concernées.
- **La vraisemblance** traduit la possibilité qu'un risque se réalise. Elle dépend essentiellement des vulnérabilités des supports face aux menaces et des capacités des sources de risques à les exploiter.

Ainsi, pour procéder à une telle analyse appliquée au cas des systèmes d'IA, on peut se poser cinq grandes questions. Celles-ci peuvent être combinées avec certaines approches proposées par des sociétés spécialisées en sécurité des systèmes d'IA comme par exemple CalypsoAI. Les réponses à ces questions permettront ensuite d'identifier et évaluer les risques liés à la sécurité et d'élaborer la politique de sécurité adaptée :

- **Qui ?** [*sources de risque*]
  Renseigne sur le profil de l'attaquant
- **Comment ?** [*vulnérabilités des supports*]
  Renseigne sur la façon dont est menée l'attaque
- **Pourquoi ?** [*menaces et motivations des sources de risques*]
  Renseigne sur les raisons de l'attaque
- **Sur quoi ?** [*descriptions des données et des événements redoutés*]
  Renseigne sur les données ciblées lors de l'attaque
- **Avec quels impacts ?** [*conséquences pour les personnes*]
  Renseigne sur les conséquences pratiques de l'attaque

Qui ? [*sources de risque*]

Cette première étape de questionnement vise à déterminer les sources de risque pesant sur un système d'IA. En pratique, pour la gestion de risque sur la vie privée, on distingue les sources humaines (interne ou externe à l'organisme mettant en œuvre le système) et non-humaines (incendies, inondations, etc.). Ici, un focus particulier est mis sur les sources humaines et en particulier externes qui sont qualifiées « d'attaquant » du système d'IA. Les profils d'attaquant peuvent être nombreux et variés et peuvent dépendre de nombreux facteurs : but du système d'IA, organisation qui le met en place, etc. Voici quelques exemples :

- Un utilisateur du système, notamment s'il souhaite en démontrer l'inefficacité (par exemple dans le cas où le système d'IA vient remplacer une tâche manuelle)
- Un chercheur ou un ingénieur éprouvant un système commercial d'apprentissage automatique (potentiellement pour un travail de recherche, comme les attaques sur Metamind, Google, Amazon et Clarifai)
- Un hacker éthique (*white hat*), par exemple participant à un *bug bounty*
- Un expert en sécurité informatique (*pentester*, *red teamer, blue hat*) réalisant des tests (intrusion, etc.)
- Un hacktiviste attaquant un système d'IA pour démontrer un point
- Un concurrent souhaitant réduire les performances d'un système
- Un hacker mal intentionné (*black hat*) attaquant un système d'IA pour obtenir un gain financier (que ce soit en déployant réellement l'attaque ou en la vendant à des tiers).
- Un groupe de hackers mal intentionnés (par exemple Anonymous, The Shadow Brokers et Legion of Doom).
- Une organisation financée par un État (dans ce cas, il s'agit principalement de cyberguerre).

Comment ? [*vulnérabilités des supports*]

Pour chaque adversaire peuvent être définies des types de connaissances et d'outils auxquels il peut avoir accès et donc donner une indication sur le niveau de sécurisation attendu. Il s'agit ici de savoir de quelle manière un attaquant pourra effectivement attenter au bon fonctionnement d'un modèle d'IA. En premier lieu, il faut noter qu'un système d'IA est un système d'information comme un autre. Il peut donc également être sujet aux attaques plus « traditionnelles ». Il est recommandé dans ces cas de mettre en œuvre les bonnes pratiques de sécurité et d'hygiène informatique comme proposé par la CNIL dans son guide sécurité des données personnelles ou l'ANSSI. Dans la suite, on s'intéresse **uniquement aux vulnérabilités spécifiques aux systèmes d'IA utilisant l'apprentissage automatique**. Les moyens de mener des attaques sur un système d'IA peuvent être analysés selon quatre dimensions :

- **Le moment de l'attaque**
- **Les connaissances dont dispose l'attaquant**
- **Les limitations de l'attaquant**
- **Les alternatives dont dispose l'attaquant**

Le moment de l'attaque

L'attaque d'un système d'IA peut se produire à différents moments dans la chaîne de traitement. En pratique, on considère deux étapes :

- **Pendant la phase d'apprentissage ou d'entrainement :** l'attaquant est en mesure d'altérer le processus d'apprentissage du modèle d'IA, notamment en introduisant des données corrompues. Pouvoir attaquer un système d'IA pendant la phase d'apprentissage peut s'avérer très efficace mais est également difficile à réaliser. De nombreuses recherches se sont penchées sur cette question comme par exemple (Kearns et al., 1993), (Biggio et al., 2011) ou (Kloft et al., 2010).
- **Pendant la phase d'inférence ou de production :** l'attaquant est uniquement en capacité de modifier la donnée d'entrée qu'il soumet au système. Une attaque à ce moment de la chaîne de traitement n'est pas nécessairement très puissante mais est beaucoup plus facile à réaliser. Là encore, on trouve de nombreux travaux scientifiques comme par exemple (Papernot et al., 2017).

Les connaissances dont dispose l'attaquant

Un attaquant sera en mesure de mettre en œuvre une attaque en fonctions de la connaissance des éléments internes du système d'IA dont il dispose. Plus précisément, on distingue :

- **Les attaques en boîte blanche (*White Box*) :** elles supposent que l'attaquant à accès à de nombreuses informations sur le système d'IA : la distribution des données ayant servi à l'apprentissage du modèle (potentiellement l'accès à certaines parties de celles-ci), l'architecture

du modèle, l'algorithme d'optimisation utilisé, ainsi que certains paramètres (par exemples les poids et les biais d'un réseau de neurones). De nombreux travaux de recherche font l'hypothèse d'attaques en boîte blanche comme par exemple (Szegedy et al., 2014) (Biggio et al., 2017) ou (Goodfellow et al., 2015).

- **Les attaques en boîte noire (*Black Box*) :** elles supposent que l'attaquant ne sait rien du système d'IA contrairement aux attaques en boîte blanche. L'utilisateur ne dispose que des informations de sortie du système. Celles-ci peuvent être de deux types :
  - **label :** l'attaquant reçoit uniquement les étiquettes (ou labels) prédits par le système
  - **score :** l'attaquant reçoit les étiquettes prédites accompagnées de scores de confiance du système

  De nombreux travaux, comme par exemple, (Dalvi et al., 2004) ou (Xu et al., 2016) partent de cette hypothèse parfois considérée plus réaliste que les attaques en boîte blanche. Une variation des attaques en boîte noire, dénommée *NoBox,* consiste à mener l'attaque sur un modèle de substitution, reconstruit sur la base de la compréhension (éventuellement limitée) par l'attaquant du système d'IA ciblé.
- **Les attaques en boîte grise (*Grey Box*) :** elles se situent quelque part entre les deux précédentes. Par exemple, l'attaquant peut disposer d'informations sur le modèle mais pas sur les données d'apprentissage utilisées (ou l'inverse).

De façon générale, il est évident que plus un attaquant dispose de connaissances sur un système d'IA et plus il sera en capacité de mettre en œuvre des attaques efficaces.

Les limitations de l'attaquant

Un attaquant devra également composer son action en fonction de contraintes spécifiques au système d'IA ciblé. On parle alors de limitations et celles-ci peuvent prendre de très nombreuses formes. Par exemple :

- Dans le cas de l'introduction d'un fichier malveillant dans un système d'IA, par exemple en essayant de la faire passer pour « bénin », un attaquant ne pourra modifier celui-ci qu'à certains endroits bien spécifiques, afin de conserver sa fonctionnalité malveillante.
- Pour attaquer un système d'IA déployé dans des dispositifs physiques (voitures, drones, satellites, caméras de surveillance, etc.), un attaquant pourra être limité à la modification de l'entrée dans le domaine physique (à moins de se placer en aval des capteurs physiques du dispositif).
- Pour qu'une attaque au niveau de la phase d'apprentissage du système d'IA soit efficace, deux conditions sont nécessaires :
  - Le système doit être régulièrement ré-entrainé sur la base de nouvelles données (sinon, l'attaquant ne sera pas en mesure d'injecter ses données corrompues).
  - Le système doit accepter de recevoir des données issues de sources externes (idéalement pour l'attaquant sans qu'un humain n'intervienne pour valider manuellement ces ajouts)
- Pour être menées à bien, de nombreuses attaques nécessitent de pouvoir réaliser des requêtes sans restriction. Par ailleurs, l'accès au score de confiance associé à un décision est bien souvent indispensable, ce qui peut s'avérer complexe à obtenir. Par exemple, la plupart des produits anti-virus du marché fournisse l'étiquette du fichier soumis au système (« malveillant » ou « bénin »), sans donner plus de précisions.

Les alternatives dont dispose l'attaquant

Enfin, il est essentiel de garder à l'esprit qu'un individu souhaitant attaquer un système d'information se focalisera sur les composants les plus faiblement protégés. En effet, l'attaque des composants au cœur d'un système d'IA n'est peut-être pas le moyen le plus facile pour un individu malintentionné d'arriver à ses fins. Un système d'IA est également un système d'information au sens classique du terme et peut donc également être sujet à des attaques plus « traditionnelles ». Ainsi, si l'objectif de l'attaquant est le vol de données personnelles, l'utilisation de modèles de substitution n'est peut-être pas la façon la plus simple d'y parvenir.

De plus, il peut demeurer une incertitude sur la réussite des attaques spécifiques aux systèmes d'IA, notamment celles qui visent à modifier le modèle ciblé et un attaquant pourra privilégier des attaques aux résultats plus garantis.

Pourquoi ? [*menaces et motivations des sources de risques*]

Pour cette étape, il s'agit de déterminer dans quel but et avec quelles motivations, un attaquant va s'en prendre à un système d'IA. La détermination du **but de l'attaquant** peut être réalisée en étudiant comment une attaque attente à un ou plusieurs éléments de la triade définie dans la norme internationale de sécurité des systèmes d'information ISO/CEI 27001 :

- **la confidentialité :** qui a pour but de s'assurer qu'une information n'est accessible qu'aux personnes autorisées.
- **l'intégrité :** qui a pour but de s'assurer qu'une donnée reste exacte et consistante à travers son cycle de vie ;
- **la disponibilité :** qui a pour but de s'assurer qu'un système ou une donnée est accessible en un temps défini ;

Une première catégorie d'attaques consiste ainsi à **extraire des informations du système d'IA** : ces attaques s'assimilent à une **perte de confidentialité**. Par exemple, un attaquant pourrait vouloir déduire si une donnée particulière (par exemple concernant une personne cible) faisait partie de l'ensemble de données d'entraînement utilisé, par exemple la sortie d'un hôpital comme illustré dans les travaux de (Shokri et al., 2017) sur le *Hospital Discharge Data Public Use Data File* du *Texas Department of State Health Services*.

Une deuxième catégorie d'attaques vise à **faire commettre au système d'IA des erreurs,** préférentiellement de façon discrète, ce qui correspondrait à une **perte d'intégrité**. Par exemple, un attaquant pourrait vouloir que le classifieur mis en œuvre par un antivirus prenne un fichier malveillant pour un fichier bénin, sans affecter ses performances globales, de sorte que celui-ci demeure non détecté. Les attaques en intégrité peuvent avoir plusieurs sous-objectifs plus ou moins complexes à mettre en œuvre, par exemple :

- **Mauvaise classification ciblée :** l'attaquant fait en sorte qu'une donnée appartenant à une classe spécifique (par exemple, « malveillante ») soit classée comme appartenant à une autre classe spécifique (par exemple, « bénigne »)
- **Erreur de classification :** l'attaquant fait en sorte qu'une donnée appartenant à une classe spécifique (par exemple « panneau stop ») soit classée comme n'importe quelle autre classe (par exemple « limite de vitesse 60 » ou « limite de vitesse 45 » ou « chien » ou « humain » ou autre, tant qu'il n'y a pas d'arrêt).
- **Mauvaise classification :** l'attaquant fait en sorte que les données soient systématiquement mal classées (proche des attaques en disponibilité décrites après)
- **Réduction de la confiance :** l'attaquant souhaite altérer la qualité des sorties du système d'IA et notamment abaisser les scores de confiance associés (cela peut s'avérer utile lorsqu'un seuil est utilisé, comme dans le cas des scores de fraude). Cela est à rapprocher de l'erreur de classification en tentant de la généraliser à toutes les données soumises.

À noter que si les exemples donnés plus haut sont directement appliqués à des algorithmes de classification, ils peuvent également être déclinés à d'autres types de systèmes comme des modèles de régression par exemple (Jagielski et al., 2018).

Enfin, la troisième catégorie d'attaque vise à **mettre le système d'IA complètement hors service**. De telles attaques correspondent à une **perte de disponibilité**. Par exemple, si suffisamment de données de mauvaise qualité, mal étiquetées, etc. sont insérées dans l'ensemble d'entraînement, le modèle appris ne sera plus pertinent et deviendra inutile. C'est l'équivalent de l'attaque informatique classique par déni de service (DoS pour *Denial of Service* en anglais) appliquée au domaine de l'apprentissage automatique.

Sur quoi ? [*description des données et des événements redoutés*]

Cette nouvelle étape vise à préciser les typologies de données, et notamment de données à caractère personnel, qui sont ciblées lors de l'attaque d'un système d'IA. Le Règlement général sur la protection des données (RGPD) et les différents textes relatifs à la protection des données explicitent plusieurs catégories de données distinctes :

- Les données à caractère personnel d'usage courant
- Les données à caractère hautement personnel (comme définies dans les Lignes directrices du Groupe Article 29 concernant l'analyse d'impact relative à la protection des données (AIPD) et la manière de déterminer si le traitement est «susceptible d'engendrer un risque élevé» aux fins du règlement (UE) 2016/679)
- Les données relatives aux personnes vulnérables
- Les données sensibles (au sens de l'article 9 du RGPD)
- Les données relatives aux condamnations pénales et aux infractions (au sens de l'article 10 du RGPD)

Il convient de relever que ces données peuvent être fournies par la personne, récupérées auprès d'un tiers qui les détenait préalablement ou encore, déduites. En effet, dans de nombreux cas, l'utilisation de systèmes d'IA vise à inférer sur la base des données utilisées pour l'entraînement (de la personne directement ou d'autres personnes), des informations sur la personne concernée. Il peut s'agir par exemple d'informations relative à l'affection ou non d'une pathologie. Il convient donc de s'assurer de l'exhaustivité du recensement des données personnelles en n'omettant pas ces données inférées.

Types de données	Catégories de données
Données à caractère personnel d'usage courant	Donnée d'état-civil, identité, données d'identification
	Données de vie personnelle (habitudes de vie, situation familiale, photos, etc.)
	Données de vie professionnelle (CV, scolarité formation professionnelle, distinctions, etc.)
	Données de connexion (adresses IP, journaux d'événements, etc.)
Données à caractère hautement personnel	Données de localisation (déplacements, données GPS, GSM, etc.)
	Données relatives aux communications électroniques
	Données d'ordre économique et financier (revenus, situation financière, situation fiscale, etc.)
	Données de traitement médico-administratif (numéro de sécurité sociale, etc.)
Données à caractère personnel relatives aux personnes vulnérables	Données concernant des personnes âgées, enfants, personnes en situation de handicap, etc.
Données à caractère personnel sensibles	Données révélant l'origine raciale ou ethnique, les opinions politiques, les convictions religieuses ou philosophiques ou l'appartenance syndicale
	Données génétiques et données biométriques utilisées aux fins d'identifier une personne physique de manière unique
	Données concernant la santé ou concernant la vie sexuelle ou l'orientation sexuelle
Données à caractère personnel relatives aux condamnations et aux infractions	Données à caractère relatives aux condamnations pénales, aux infractions ou aux mesures de sûreté

Table 1. Proposition de typologie des données à caractère personnel.

Attenter à ces données donne lieu à un ou des « événements redoutés ». Comme vu dans la section précédente, ces événements peuvent prendre trois formes lorsqu'appliqués à des données à caractère personnel :

- **Un accès illégitime aux données.** Elles sont alors connues de personnes non autorisées. Il y a atteinte à la confidentialité des données.
- **Une modification non désirée de données.** Elles ne sont plus intègres ou sont changées. Il y a atteinte à l'intégrité des données
- **Une disparition de données.** Elles ne sont pas ou plus disponibles. Il y a atteinte à la disponibilité des données.

Avec quels impacts ? [*conséquences pour les personnes*]

Comme indiqué » dans le guide de la CNIL PIA, les bases de connaissance, on distingue généralement quatre niveaux d'impacts pour les personnes à la survenue d'un événement redouté :

- **Négligeable :** les personnes concernées ne seront pas impactées ou pourraient connaître quelques désagréments, qu'elles surmonteront sans difficulté
- **Limitée :** Les personnes concernées pourraient connaître des désagréments significatifs, qu'elles pourront surmonter malgré quelques difficultés
- **Importante :** Les personnes concernées pourraient connaître des conséquences significatives, qu'elles devraient pouvoir surmonter, mais avec des difficultés réelles et significatives
- **Maximale :** Les personnes concernées pourraient connaître des conséquences significatives, voire irrémédiables, qu'elles pourraient ne pas surmonter

Ces impacts peuvent être précisés en caractérisant leur nature :

- **Corporelle :** préjudice d'agrément, d'esthétique ou économique lié à l'intégrité physique
- **Immatérielle :** perte subie ou gain manqué concernant le patrimoine des personnes
- **Morale :** souffrance physique ou morale, préjudice esthétique ou d'agrément

Appliqué à une attaque menée à l'encontre d'un système d'IA, il s'agit de mener une réflexion visant à mesurer ces impacts le plus honnêtement possible. Ainsi une attaque par inférence d'appartenance réalisée à l'encontre d'un système utilisé pour la caractérisation d'une maladie grave aura un impact moral, potentiellement important, alors qu'une attaque en disponibilité menée à l'encontre d'un système de vision par ordinateur visant à extraire des informations relatives aux émotions des images de visage pourra avoir un impact limité. Et ces impacts diffèreront encore d'une attaque par exfiltration permettant de recouvrir des informations ayant été utilisées pour entrainer un modèle de langage (numéro de carte bleue, de téléphone, adresse, etc.) !

Estimer le risque [*gravité + vraisemblance*]

Une fois que les source de risques et leurs motivations, les vulnérabilités des supports, les données exposées et les impacts potentiels pour les personnes concernées ont été caractérisés, il s'agit d'estimer le ou les risques identifiés. C'est la dernière étape du processus de qualification des risques. Ainsi, pour chaque événement redouté identifié, il s'agit de répondre aux deux questions suivantes :

- **« Que craint-on qu'il arrive aux personnes concernées ? » :** afin de déterminer les impacts potentiels pour les personnes concernées s'ils survenaient.
- **« Comment cela pourrait-il arriver ? » :** afin d'identifier les menaces sur les supports des données qui pourraient mener à cet événement redouté et les sources de risques qui pourraient en être à l'origine.

La réponse à la première question permet d'estimer **la gravité**, notamment en fonction du caractère préjudiciable des impacts potentiels et, le cas échéant, des mesures susceptibles de les modifier. La réponse à la seconde permet, elle, d'estimer **la vraisemblance**, notamment en fonction des vulnérabilités des supports de données, des capacités des sources de risques à les exploiter et des mesures susceptibles de les modifier. La Figure 2, présente la matrice des risques couramment utilisée pour visualiser ceux-ci ainsi que les conséquences de l'applications de mesures adaptées pour les réduire (il s'agit d'un exemple).

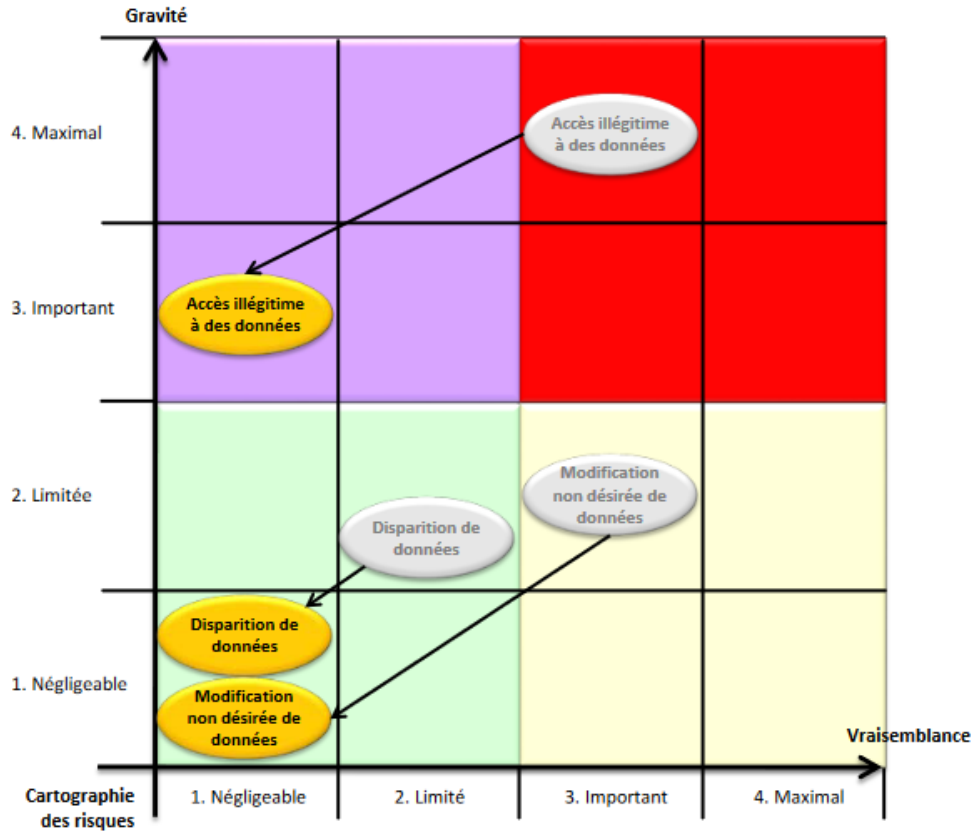


Figure 2. Exemple de cartographie des risques liés à la sécurité (source guide de la CNIL PIA les modèles).

La caractérisation des risques en fonction de leurs gravité et vraisemblance permettra de déterminer s'ils peuvent être jugés acceptables compte tenu des mesures existantes ou prévues. Si non, il conviendra de proposer des mesures complémentaires et réestimer le niveau de chacun des risques en tenant compte de celles-ci, afin de déterminer les risques résiduels.

Sécurité des systèmes d'IA, les gestes qui sauvent

Sécuriser un système d'IA s'avère être une tâche particulièrement complexe. Néanmoins, il est possible de mettre en œuvre des mesures tout au long du cycle de vie qui, si elles ne garantissent pas de façon certaine la sécurité du système et l'impossibilité qu'une attaque soit menée, réduisent néanmoins substantiellement les risques pour la vie privée des personnes concernées.

Crédit : Todd Lappin

Plusieurs publications, comme par exemple celle proposée par la société Wavestone, listent les bonnes pratiques à la fois techniques et de gouvernance à mettre en œuvre pour sécuriser un système d'IA. La sécurité des systèmes d'IA étant un domaine de recherche en plein essor, on observera qu'à des mesures élémentaires peuvent être ajoutés des mécanismes de défense beaucoup plus raffinés, voire encore expérimentaux et connaissant des évolutions très rapides.

Avoir un plan de déploiement

La mise en œuvre d'un système d'IA suppose avant toute chose d'avoir réfléchi à la façon de le faire. Si la mise en œuvre pratique peut bien souvent remettre en question certains choix de conception, la formalisation préalable des exigences permet de s'assurer que les solutions déployées sont bien cohérentes avec les exigences préalablement identifiées.

Penser l'architecture

En fonction de la tâche à réaliser (classer des images, extraire des informations de données textuelles, etc.) différents types de systèmes d'IA peuvent être déployés. Il convient donc de répondre à plusieurs questions afin de choisir celui qui est le plus approprié.

- **Quel type de problème veut-on résoudre ?**

Définir s'il s'agit de régression, de classification, de partitionnement de données, de segmentation, de génération de contenu, etc.

- **Quel type de système souhaite-t-on mettre en œuvre ?**
  En fonction du problème à résoudre, préciser si on veut mettre en œuvre un système d'apprentissage supervisé, non-supervisé, par renforcement, en continu, fédéré, distribué, centralisé, etc.
- **Quelles sont les implications de ces choix ?**
  Chaque système dispose de ses propres particularités et emporte donc des conséquences spécifiques. Les implications pour la vie privée d'un système d'apprentissage fédéré ou d'un système centralisé ne sont par exemple pas les mêmes, de même que les systèmes d'apprentissage en continu présentent des vulnérabilités différentes que les systèmes pour lesquels les modèles sont appris « une fois pour toute » (*once and for all*).

Séquencer le traitement

La puissance de l'apprentissage automatique (*machine learning*) tient à son pouvoir à établir des corrélations parmi de très grands volumes de données. Lors de la phase d'apprentissage, les modèles sont bien souvent entraînés avec plus de données que ce qui sera *in fine* strictement nécessaire, cela afin de déterminer les combinaisons qui s'avéreront les plus efficaces en pratique. Ensuite, une fois le modèle sélectionné et validé, il est possible de réduire le nombre et la nature des données utilisées au strict nécessaire. C'est ce sous-ensemble de données qui sera alors utilisé pour le passage en production du système d'IA.

Il convient donc de bien séquencer son traitement d'IA pour séparer clairement :

- **La phase de R&D :** phase au cours de laquelle une exploration sera menée pour caractériser le meilleur système permettant de résoudre la tâche étudiée (en termes d'architecture, de données utilisées, de choix de paramètres et d'hyperparamètres, etc.) et réaliser son entraînement.
- **La phase de production (ou opérationnelle) :** phase consécutive à celle de R&D et qui verra le système d'IA utilisé pour la tâche pour laquelle il a été entrainé.

Il est essentiel d'avoir une bonne idée de ce découpage en phases car celles-ci doivent répondre à des exigences différentes. Elles peuvent être rattachées à deux finalités différentes au titre de la protection des données (et donc recourir à deux bases légales différentes). En fonction des architectures choisies, ce découpage en phase peut s'avérer plus délicat à réaliser. A titre d'exemple, un système d'apprentissage en continu peut être vu comme une succession de phases de R&D et de production.

Avoir une approche *privacy by design*

En fonction de son domaine d'emploi, de son architecture et des données qui l'alimentent, un système d'IA peut présenter des risques plus ou moins importants pour la vie privée. Dans certains cas, ceux-ci peuvent être atténués par la mise en œuvre de mesures protectrices dès la conception du système. Il est possible de jouer sur différents leviers en pratique :

- **Explorer la possibilité de déployer des architectures respectueuses de la vie privée.** Certaines pratiques de mises en œuvre des systèmes d'IA peuvent apporter des garanties pour la confidentialité des données. Les approches d'apprentissage fédéré (*federated learning*) par exemple peuvent parfois apporter plus de garanties qu'un système centralisé en limitant la circulation des données (voir à ce sujet l'article *Chacun chez soi et les données seront bien gardées : l'apprentissage fédéré*).
- **Renforcer l'environnement d'exécution du système d'IA**. Les modalités relatives aux conditions pratiques de déploiement des systèmes doivent garantir une bonne sécurisation. A titre d'exemple, des environnements de confiance (*trusted execution environment*) peuvent être mis en œuvre.
- **Utiliser les ressources offertes par la cryptographie.** Les progrès scientifiques récents dans le domaine de la cryptographie peuvent permettre d'obtenir des garanties fortes pour la protection des données. En fonction des cas d'usage, il pourra par exemple être pertinent d'explorer les possibilités offertes par le calcul multipartite sécurisé (*secure multi-party*

computation), le transfert inconscient (*oblivious transfer*) ou encore le chiffrement homomorphe (*homomorphic encryption*).

- **Utiliser des méthodes d'anonymisation.** Certaines méthodes d'anonymisation comme la confidentialité différentielle (*differential privacy*) peuvent être introduites et utilisées à différents endroits de la chaîne de traitement. Le recours à des techniques d'anonymisation réduit en pratique les risques de violation de données et de vol ou de rétro-ingénierie du modèle. Celle-ci peut par exemple être appliquée pour :
    - Perturber les données utilisées lors de la phase d'apprentissage
    - Perturber les paramètres du modèle appris, par exemple, injecter du bruit dans le processus de mise à jour des paramètres comme proposé par (Long et al, 2017)
    - Perturber la fonction de perte utilisée lors de la phase d'apprentissage
    - Perturber l'entrée soumise au système en phase de production
    - Perturber la sortie fournie par le système en phase de production

A noter qu'un compromis entre utilité et protection du modèle contre les attaques en confidentialité doit être trouvé (Rahman et al, 2018).

- **S'assurer que les règles élémentaires de SSI sont bien mises en œuvre.** Il existe des *frameworks* spécialisés pour le développement de système d'IA sécurisés comme SecML proposé par (Melis et al, 2019). Il est par ailleurs essentiel de garder à l'esprit que les systèmes d'IA sont avant tout des systèmes informatiques et qu'ils doivent ainsi répondre aux mêmes exigences de sécurisation. Le Guide de la sécurité des données personnelles de la CNIL présente les mesures élémentaires à mettre en œuvre, y compris pour un système d'IA.

Être vigilant aux ressources utilisées

Un système d'IA implique l'exploitation de différents objets. On peut en particulier en distinguer trois :

- **Les données :** utilisées à la fois pour l'élaboration du modèle en phase de R&D et l'exploitation en phase de production
- **Les modèles :** créés à partir des données, ils peuvent parfois être le résultat d'un enrichissement et d'une adaptation de modèles existants en utilisant les technologies d'apprentissage par transfert (*transfer learning*)
- **Le code :** « incarnation » informatique du système d'IA, il peut être produit spécifiquement pour la tâche à remplir ou être dérivé de ressources existantes (code disponible en *open* source par exemple). Il fait quasi systématiquement appel à des briques existantes (librairies, exemples disponibles, etc.)

Les données

S'assurer de la légalité

Les données étant au cœur du fonctionnement des systèmes d'IA, il est essentiel de s'assurer que la collecte et le traitement de celles-ci ainsi que leur usage sont conformes à la réglementation en vigueur, et notamment à celle liée à la protection des données lorsque ces données sont à caractère personnel (RGPD, Loi Informatique et Libertés, directive ePrivacy, directive « Police-Justice », etc.).

Une analyse de conformité concernant l'accès à ces données pour le développement et l'utilisation d'un système d'IA est donc indispensable. Comme tout traitement de données, celui-ci doit donc être caractérisé et décrit comme le veut la réglementation en vigueur (finalité, base légale, durée de conservation, exercice des droits, mesures de sécurité). En fonction de la finalité du traitement, la réalisation d'une Analyse d'impact relative à la protection des données peut s'avérer indispensable.

S'assurer de la qualité

Les données, utilisées dans des jeux d'entraînement, de validation ou de test, doivent répondre à des exigences de qualité afin de construire le meilleur système possible. Il convient donc d'être attentif à différents aspects, par exemple :

- L'adéquation des données au problème à résoudre
- La manière dont les données ont été collectées, qui peut apporter un éclairage sur celles-ci

- La représentativité des données pour le problème à résoudre
- La présence d'éventuels biais dans les données
- La disponibilité et la quantité de données disponibles
- Les éventuels problèmes dans les données (données manquantes, non-renseignées, aberrantes, etc.)

La phase de nettoyage des données (*data sanitization*) est une étape essentielle pour la constitution d'un jeu de données utilisable par un système d'IA. Un suivi (*monitoring*) des sources de données et des mécanismes de seuillage visant à limiter l'utilisation de données issues d'une même source (zone géographique, individu, IP, etc.) peuvent également être mis en œuvre pour réduire les risques de sur-représentation de certaines catégories de données.

Il est par ailleurs primordial que l'intégrité des données collectées et utilisées par le système d'IA soit assurée. Cela est valable tant pour la phase de R&D que pour la phase de production, et les données ne doivent pas pouvoir être altérées de façon non prévue par le concepteur du système (principe d'exactitude). La chaîne d'acquisition des données doit donc être conçue pour garantir une protection de bout en bout, en limitant les canaux d'entrée, en appliquant une gestion d'accès stricte ou encore en chiffrant les flux de données.

Dans le cas de jeux de données réutilisés, et en particulier en ce qui concerne ceux disponibles en open source, un devoir de vigilance s'impose, ces jeux pouvant ne pas apporter de garanties de qualité satisfaisantes mais également avoir été infectés dans le but de mener une attaque sur le système d'IA.

S'assurer de la désensibilisation

Le fonctionnement des systèmes d'IA, basés sur l'utilisation de quantités de données importantes voire très importantes, met en tension certains principe du RGPD, notamment celui de la minimisation des données. Par ailleurs, et comme présenté précédemment, la phase de conception du système d'IA ou phase de R&D peut nécessiter l'accès à des typologies de données plus larges que celles qui seront finalement utilisées en production.

Toutefois, il est indispensable de faire en sorte que les données ne soient pas excessives. Il ne faut pas conserver dans les jeux de données des informations dont il est certains qu'elles ne doivent pas être exploitées. Ainsi, l'élaboration d'un système d'IA visant à produire un diagnostic médical ne devra jamais être réalisé sur des données directement nominatives. Le recours à des mécanismes de pseudonymisation (tel que pour des documents textuels) ou de filtrage/obfuscation (par exemple, numéro à 16 chiffres pour les cartes bleues) est donc indispensable.

Par ailleurs, afin de s'affranchir de certaines contraintes relatives à la protection des données personnelles, le recours à des données anonymisées ou de synthèse peut s'avérer très efficace. L'anonymisation étant une notion mouvante et complexe, il convient toutefois de s'assurer au préalable que celle-ci a été bien réalisée (voir à ce sujet la Fiche pratique de la CNIL).

S'assurer de la traçabilité

Il est indispensable d'être en mesure d'assurer la traçabilité des données utilisées par les systèmes d'IA (en particulier en phase de R&D) et de documenter leur provenance et les conditions de leur collecte. En plus de permettre d'assurer la légalité, ces éléments de connaissance permettront en effet au concepteur du système d'IA de disposer d'informations pour améliorer la qualité de son système.

Par ailleurs, certaines technologies actuellement en développement visent à opérer une traçabilité des données et ainsi à être en mesure d'observer si elles ont été utilisées pour l'apprentissage d'un modèle. On parle ainsi de données « radioactives » (Sablayrolles et al., 2020). Ces recherches, généralement plutôt motivées par des questions de propriété intellectuelle, pourront trouver à s'appliquer dans le cas des données personnelles.

Les modèles

La conception d'un système d'IA peut dans bien des cas être opérée à l'aide de modèles d'IA existants. En effet, les contraintes introduites par l'apprentissage profond rendent celui-ci coûteux à la fois en ressources calculatoires, en quantité de données nécessaires et en temps. Dans de nombreux domaines d'application, on a donc recours aux technologies d'apprentissage par transfert (*transfer learning*) pour faciliter la création de modèle (par exemple, pour les modèles de langage, de vision, etc.).

Concrètement, il s'agit d'utiliser un modèle appris sur de très grandes catégories de données et de l'adapter au problème visé à l'aide de données spécifiques à celui-ci mais dont la quantité nécessaire est bien moindre que si un apprentissage complet devait être réalisé. A titre d'exemple, la détection de tumeurs cancéreuses peut se faire à l'aide du modèle généraliste GoogLeNet adapté à ce cas d'usage.

Comme décrit dans l'article *Petite taxonomie des attaques des systèmes d'IA*, un attaquant peut se servir d'un modèle en *open source* qu'il aura infecté pour introduire des portes dérobées (*backdoors*) ou des chevaux de Troie. Il est donc indispensable, lorsqu'on a recours à un modèle mis en œuvre par un tiers de s'assurer :

- De la fiabilité de la source auprès de laquelle on le récupère
- De disposer d'éléments de connaissance relatifs à la constitution de celui-ci
- De disposer de la dernière version du modèle en question

Le code

Comme l'élaboration de tout programme informatique, la conception de systèmes d'IA repose sur le développement de code. Cependant, la conception de systèmes d'IA étant une tâche très complexe, elle repose bien souvent sur l'utilisation de ressources existantes. Si comme dans d'autres domaines de l'informatique, il est quasiment impossible de se passer de l'utilisation de librairies de références (TensorFlow, Keras, Scikit-Learn, etc.), il est également fréquent de réutiliser des parties de codes sources produites par d'autres.

Comme pour les modèles d'IA disponibles en *open source*, il est nécessaire de s'assurer :

- De la fiabilité de la source auprès de laquelle on récupère ces éléments
- De disposer d'éléments de connaissance relatifs à leur production
- De s'assurer qu'ils n'aient pas été corrompus (en inspectant le code récupéré)

Le Guide RGPD du développeur de la CNIL précise les bonnes pratiques de développement à mettre en œuvre de façon générale pour satisfaire les impératifs de protection des données.

Par ailleurs, dans le cadre de contrat passé entre un organisme et un fournisseur externe de solutions d'IA, l'usage qui peut être fait du code source propriétaire peut être très précisément encadré. Ces questions sont abordées dans la dernière partie.

Sécuriser et durcir le processus d'apprentissage

La phase d'apprentissage des systèmes d'IA est une étape clé et une de leurs spécificités. Il s'agit donc de consacrer un effort tout particulier à l'élaboration de celle-ci afin d'assurer que l'apprentissage est correctement réalisé, que le modèle est robuste mais également qu'il n'offre pas de prises à un éventuel attaquant. Une protection du processus d'apprentissage doit donc être réalisée et cela à deux niveaux : au niveau des données utilisées pour l'entraînement du système et au niveau de la méthode d'apprentissage mise en œuvre.

Au niveau des données d'entrainement

Outre les questions relatives à la qualité des données collectées pour être utilisées lors de l'apprentissage d'un système d'IA (voir précédemment), il convient de s'assurer que celles-ci ne servent pas les desseins d'un attaquant.

Surveiller l'impact des données

Comme présenté dans l'article *Petite taxonomie des attaques des systèmes d'IA*, les attaques par infection, et notamment celle par empoisonnement (*poisoning attacks*), permettent aux attaquants d'exercer un contrôle des systèmes d'IA de façon dissimulée en contaminant les données (pour abaisser la qualité du modèle produit, pour intégrer une porte dérobée, etc.). L'état de l'art scientifique propose plusieurs méthodes pour se prémunir contre de telles attaques et consolider le modèle d'IA produit, telles que :

- **Contrôle itératif de l'apprentissage (*iterative learning control*) :** étudie l'impact de chaque donnée sur le fonctionnement du modèle. On parle également de défense RONI (*Reject On Negative Impact*) permettant de supprimer du jeu d'apprentissage les données ayant un impact négatif sur la précision du modèle (Nelson et al, 2008).
- **Apprentissage actif (*active learning*) :** fait intervenir un opérateur pendant le processus d'apprentissage pour lui demander de qualifier certaines données (Settles, 2010). Il s'agit d'une méthode d'apprentissage semi-supervisée.

Des méthodes moins complexes peuvent également être mises en œuvre pour assurer le contrôle des données pendant la phase d'apprentissage tel que le contrôle de l'intégrité des données ou la définition de listes noires (*blacklists*) recensant des mots clés ou patterns à retirer systématiquement du jeu d'apprentissage (par exemple le vocabulaire injurieux dans le cas d'un chatbot).

Consolider son jeu de données

En plus des attaques par infection, certaines attaques dites par manipulation visent à dégrader la qualité des sorties produites en fournissant au système en production des données corrompues. Pour se prémunir de telles attaques, la littérature scientifique propose différentes stratégies telles que :

- **Augmentation de données (*data augmentation*)** : cette méthode permet d'augmenter la quantité de données en ajoutant des copies légèrement modifiées de données existantes. Cela permet une « régularisation » du modèle d'IA. Cette technique est efficace, sauf si les données sont limitées. Il peut alors manquer certaines informations sur les données qui n'ont pas été utilisées pour l'entraînement, et les résultats peuvent donc s'avérer biaisés.
- ***Randomisation* :** ajoute un bruit aléatoire à chaque donnée utilisée pour l'entraînement. L'ajout de ce bruit rend plus difficile pour un attaquant de prédire la perturbation à ajouter à une entrée pour arriver à ses fins. L'intensité du bruit à ajouter doit être optimisée pour obtenir le meilleur compromis entre l'exactitude de l'algorithme et sa robustesse.
- **Entraînement contradictoire (*adversarial training*) :** utilise des exemples contradictoires (*adversarial examples*) la plupart du temps générées à l'aide de réseaux antagonistes génératifs (GANs pour *generative adversarial networks*). L'entrainement d'un modèle avec de telles données doit permettre au système d'ignorer le bruit susceptible d'avoir été ajouté par un attaquant et à n'apprendre qu'à partir de caractéristiques robustes.

Si ces méthodes visent en priorité à empêcher les attaques par empoisonnement des données d'apprentissage, aucune n'apporte de garanties formelles de la capacité à se prémunir de toutes. Même si un compromis acceptable entre performance et robustesse doit toujours être trouvé, l'idée est qu'elles permettent d'améliorer la capacité de généralisation du modèle entrainé.

Au niveau de la méthode d'apprentissage

Parallèlement au travail sur les données utilisées, il est également indispensable de sécuriser et durcir le processus d'apprentissage lui-même. Pour cela, des méthodes classiques peuvent être utilisées :

- **Validation croisée (*Cross validation*) :** permet d'estimer la fiabilité d'un modèle en utilisant une technique d'échantillonnage. Cette technique est efficace, sauf si les données sont limitées. Il peut alors manquer certaines informations présentes dans les données qui n'ont pas été utilisées pour l'entraînement, et les résultats peuvent donc être biaisés.
- **Amorçage *(Bootstraping)* :** permet d'estimer des valeurs statistiques sur une population, en faisant la moyenne des estimations obtenues sur de nombreux échantillons issus de cette population (par exemple pour estimer la moyenne, l'écart-type ou même un intervalle de

confiance pour les paramètres du modèle, ou pour estimer ses performances). Cette méthode consiste à tirer de manière aléatoire uniforme des observations l'une après l'autre, et en les remettant dans l'échantillon d'origine une fois qu'elles ont été choisies.

- **Normalisation de lot (*Batch normalization*) :** cette technique spécifique à l'entrainement de réseaux de neurones profonds standardise (centre-réduit) les entrées soumises à une couche du réseau pour chaque mini-batch (la couche d'entrée, mais également les couches cachées). Cela a pour effet de stabiliser le processus d'apprentissage et réduire la durée de l'entraînement du réseau. A noter que d'autres types de normalisation existent : la normalisation des poids (*Weight normalization*), des couches (*Layer normalization*) ou encore par groupe (*Group normalization*).
- **Quantification (*Quantization*) :** cette technique, également utilisée pour l'apprentissage profond, est un processus d'approximation d'un réseau de neurones complexe qui utilise des nombres à virgule flottante par un réseau de nombres d'un ensemble « discret » d'assez petite taille. Cette troncature réduit considérablement les besoins en mémoire et le coût de calcul des réseaux neuronaux.
- **Élagage (*Pruning*) :** cette technique consiste à supprimer certaines connexions dans un réseau de neurones profond afin d'augmenter la vitesse d'inférence et de réduire la taille de stockage du modèle. En général, les réseaux neuronaux sont très sur-paramétrés. L'élagage d'un réseau peut être considéré comme la suppression des paramètres inutilisés.
- **Décrochage ou abandon (*Dropout*) :** cette technique de régularisation vise à réduire le surapprentissage dans les réseaux de neurones. Pour cela, des neurones sont sélectionnés aléatoirement et ignorés pendant la phase d'apprentissage. Ces neurones ignorés sont temporairement supprimés lors de la passe avant (*forward*), et leurs poids ne sont pas mis à jour lors de la passe arrière (*backward*). Contrairement à l'élagage, les neurones ne sont pas définitivement supprimés.

Par ailleurs différentes stratégies d'apprentissage ont pu être proposées dans la littérature scientifique pour durcir les modèles et minimiser les prises offertes à un éventuel attaquant. De nombreuses méthodes d'apprentissage ensemblistes ont en particulier été élaborées et prolongent les méthodes classiquement utilisées (*bagging, boosting* ou forêts aléatoires). Ces techniques reposent sur la combinaison de multiples algorithmes pour accroître les performances du modèle, et parvenir à un niveau de précision bien supérieur à celui qui serait obtenu si on utilisait n'importe lequel de ces algorithmes pris séparément :

- **Micromodèles :** technique permettant de réaliser l'entrainement sur de multiples modèles et évaluer leur pertinence (et la possibilité que certains soient infectés par des données corrompues) par un vote majoritaire au moment de l'inférence (Cretu et al., 2008).
- **Distillation défensive (*defensive distillation*) :** technique qui s'éloigne des approches d'apprentissage ensembliste mais utilise un modèle de référence (maître ou *teacher*) afin d'entraîner un second modèle « lissé » introduisant une faible incertitude (élève ou *student*). Le second modèle est plus robuste et donne moins de prise à un attaquant cherchant à exploiter des vulnérabilités (Papernot et al., 2016).
- ***Private Aggregation of Teacher Ensembles* (PATE) :** technique ensembliste utilisant la confidentialité différentielle (*differential privacy*). Des modèles maîtres bruités permettent l'entrainement, par un vote majoritaire sur les sorties qu'ils produisent, d'un modèle élève qui sera celui utilisé par le système. L'idée sous-jacente est que ce modèle ne laissera que très peu de prise à une attaque en confidentialité (Papernot et al., 2017).
- **Apprentissage contradictoire ensembliste (*ensemble adversarial training*) :** technique visant à contrer les attaques par exemples contradictoires et en particulier leur propriété de transférabilité. Pour cela, le modèle est entraîné en utilisant des exemples contradictoires créés à partir du modèle lui-même, mais également des exemples transférés à partir de modèles pré-entraînés (Tramer et al., 2020).

Enfin, on peut noter que de plus en plus de travaux s'intéressent à la façon de permettre à un individu d'exercer son droit d'opposition sur un modèle d'apprentissage automatique comme prévu par le RGPD. L'exercice de ce droit suppose qu'un apprentissage présentant des propriétés appropriées soit effectué. On parle ainsi de *machine unlearning* et plusieurs approches d'apprentissage comme par exemple SISA (*Sharded, Isolated, Sliced, and Aggregated*) commencent à être proposées (Bourtoule et al., 2020).

Fiabiliser l'application

S'il est essentiel de prendre en compte les risques spécifiques à un système d'IA avant de le mettre en production, il faut également garder à l'esprit que celui-ci est avant toute chose un système informatique au sens classique du terme. Sa sécurisation et sa robustesse passent donc également par l'application des mesures de sécurité « classiques ». Les bonnes pratiques de développement web présentées dans le Guide RGPD du développeur de la CNIL et dans d'autres initiatives comme l'Open Web Application Security Project (OWASP) trouvent donc à s'appliquer. En particulier, il s'agira à la fois de protéger le processus d'alimentation du système d'IA en phase de production et de maîtriser les sorties qu'il produira, ces deux étapes pouvant donner lieu à des actions malveillantes. En pratique, le recours à des experts en sécurité informatique (*pentester*, *red teamer*, *blue hat*, etc.) est recommandé. Il s'agit ainsi de s'assurer de la fiabilité et de la robustesse du système d'IA en réalisant différents tests et attaques (intrusion, contournement, etc.).

Contrôler les entrées

En premier lieu, il est essentiel qu'un système d'IA n'expose que ce qui est strictement nécessaire à son bon fonctionnement. L'accès au système doit donc être strictement limité aux seules personnes ayant nécessité d'y accéder. Un fonctionnement en mode « boîte noire » (*black box*) ne permettant aux utilisateurs que de fournir des entrées et d'observer des sorties est donc généralement à privilégier.

Des « sas de décontamination » peuvent être mis en œuvre afin de :

- **S'assurer du bon format des fichiers soumis :** il s'agit de vérifier le type de données, l'exhaustivité des informations entrées ou extraites, etc.
- **Vérifier la cohérence des données :** il s'agit de mesurer un éventuel écart par rapport aux données anticipées, à des données historisées, etc.
- **Détecter l'ajout de bruit dans les données (*noise prevention*) :** il s'agit de détecter l'éventuel présence d'une entrée corrompue par exemple en comparant la prédiction obtenue de la donnée nettoyée avec la donnée soumise (Akhtar et al, 2018).
- **Compresser les caractéristiques (*feature squeezing*) :** il s'agit de réduire l'information dans la donnée soumise à un minimum de caractéristiques suffisantes pour réaliser la tâche et ainsi de ne pas permettre l'ajout d'informations corrompues. La comparaison des sorties produites sur la donnée d'origine et sur sa version compressée permet ainsi de détecter avec une haute précision les exemples contradictoires puisqu'une grande différence de comportement est observée. Cela est en particulier adapté aux traitements de vision par ordinateur (Xu et al, 2017).

Par ailleurs, il est vivement recommandé de durcir les API (interface de programmation applicative) qui permettent l'utilisation d'un système d'IA en ligne par exemple afin de :

- Limiter le nombre de requêtes qui pourraient être soumises par un utilisateur
- S'assurer que l'utilisateur est un humain par exemple par l'utilisation de CAPTCHA (*Completely Automated Public Turing test to tell Computers and Humans Apart*)
- Imposer à chaque requête un coût calculatoire important (à la manière des fonctions à dérivation de clé utilisées utilisées pour le hachage en cryptographie)
- Analyser les comportements des utilisateurs. Les comportements suspects peuvent faire l'objet de détection et de blocage. On parle alors d'analyse UEBA (analyses comportementales des utilisateurs et des entités ou *User and Entity Behavior Analytics*).

Enfin, si les données soumises en phase de production doivent ensuite être réutilisées pour un ré-apprentissage du modèle d'IA, la mise en place d'un « bac à sable » (*sandbox*) peut permettre de vérifier que le réentrainement apporte bien les gains de performance escomptés. Il s'agit de suivre l'évolution des performances du modèle et donc de ne pas exposer ce dernier de façon définitive à des risques de dérive (*drift*) et d'infection, par exemple à des attaques par empoisonnement ou porte dérobées.

Maîtriser les sorties

S'il est indispensable de contrôler les entrées soumises à un système d'IA, de façon symétrique, il faut également s'assurer que les sorties produites sont bien protégées et n'offrent pas de prises à un éventuel

attaquant. En effet, comme illustré dans l'article *Petite taxonomie des attaques des systèmes d'IA*, un attaquant pourra exploiter les sorties d'un système d'IA pour de multiples buts : voler le modèle, l'inverser, mener une attaque par inférence d'appartenance, etc. Pour cela, il est conseillé de :

- **Réduire la « verbosité » des sorties :** il s'agit par exemple de n'exposer que des labels inférés et non pas les scores de confiance du système ou alors de produire une version grossière de ceux-ci (e.g. confiance faible / moyenne / forte) et pas des scores bruts. Il est alors plus compliqué pour un attaquant d'inférer le fonctionnement du système d'IA visé. Si elles ne permettent pas de contrer toutes les attaques, les techniques de masquage de gradient (*gradient masking*) sont une étape importante de défense des systèmes d'IA (Papernot et al, 2017).
- **Adapter les sorties :** un modèle d'IA fournissant par définition toujours une réponse aux requêtes qui lui seront soumises, il s'agit de proposer une classe « abstention » / « ne sait pas » lorsque la prise de décision est incertaine.
- **Détecter les sorties suspectes :** il s'agit de comparer un résultat produit par rapport à des indicateurs de référence et lever une alerte en cas de doute (par exemple un historique d'interactions passées). La manière de traiter les anomalies doit ensuite être définie au cas par cas : arrêt du traitement, demande de réauthentification, alertes du superviseur du traitement, etc.
- **Proposer une modération manuelle :** dans certains cas, il peut être utile de soumettre les sorties produites à un opérateur afin que celui-ci s'assure du bon fonctionnement du système d'IA avant de renvoyer une réponse.

Penser une stratégie organisationnelle

En fonction de la finalité du système d'IA, de son domaine d'emploi, des personnes à qui il se destine, de la stratégie d'apprentissage déployée (continue, ponctuelle, « une fois pour toutes »), etc. des risques différents sont susceptibles d'advenir. Il est donc recommandé de i) documenter les choix de conception, ii) superviser le fonctionnement du système, iii) identifier les personnes clés et encadrer le recours à des sous-traitants et iv) mettre en œuvre une stratégie de gestion des risques. L'article *P[ã]nser la sécurité des systèmes d'IA* précise comment peut être pensée une telle stratégie.

Documenter les choix de conception

La documentation est la pierre angulaire de la mise en œuvre d'un système d'IA sûr et résilient. Cette documentation doit refléter le cheminement qui a permis d'aboutir aux choix de conception mis en œuvre dans le système en production. Elle doit également être alimentée et mise à jour régulièrement et cela tout au long de l'utilisation du système d'IA. Pour ce faire, la CNIL propose aux professionnels un Guide d'auto-évaluation de son système d'IA avec un focus particulier sur l'utilisation de données à caractère personnelles. Par ailleurs, différentes ressources peuvent être mobilisées pour documenter son système et s'assurer de l'absence « d'angles morts ». On peut citer à titre d'exemples :

- La Certification de processus pour l'IA du Laboratoire national de métrologie et d'essais (LNE)
- L'outil ALTAI (*Assessment List on Trustworthy Artificial Intelligence*) développé par le groupe d'experts de haut niveau sur l'IA (GEHN IA) mis en place par la Commission européenne
- Le référentiel d'évaluation data sciences responsables et de confiance de l'association Labelia
- Le guide pratique pour des IA éthiques du syndicat professionnel Numeum
- L'outil d'évaluation de l'incidence algorithmique mis en œuvre par le gouvernement du Canada

Superviser le fonctionnement du système

Il est essentiel de s'assurer que le système d'IA ne dévie pas du comportement attendu et cela tout au long de sa durée de vie. Pour cela, il est indispensable d'anticiper dès le stade de la conception, la manière de suivre les évolutions du système et les indicateurs qui permettront d'opérer ce suivi. Développer une méthode d'évaluation rigoureuse est donc essentiel. Par ailleurs, il est indispensable de déployer un processus de maintien en conditions opérationnelles. Il s'agit d'assurer la conformité de la fonctionnalité d'IA aux spécifications définies après son déploiement et tout au long de sa phase d'exploitation.

Cette supervision est particulièrement critique pour la mise en place de systèmes d'apprentissage en continu, dans lesquels le modèle est mis à jour au gré de l'utilisation du système. Une mesure de l'évolution des performances du système au regard de paramètres clés ou la mise en place d'audits réguliers doivent être envisagés avant d'activer un apprentissage en continu.

Des exigences de traçabilité doivent donc être mises en œuvre. Les bonnes pratiques de traçabilité traditionnelles ne prennent pas en compte les complexités induites par l'apprentissage automatique. Il est ainsi essentiel de définir des exigences de traçabilité spécifiques, permettant notamment de garder des informations concernant les données soumises au système, paramètres qui conduisent aux décisions prises par les systèmes, etc.

Identifier les personnes clés et encadrer le recours à des sous-traitants

La mise en place d'une chaîne algorithmique complète est un exercice délicat qui mobilise les compétences de nombreux individus alliant des savoir-faire complémentaires. Il est donc nécessaire de déterminer ces compétences qui participent à la capacité des processus de conception, au développement du système d'IA, à son évaluation et au maintien en condition opérationnelle, au paramétrage des fonctionnalités d'IA et à la capacité à atteindre les résultats attendus.

Par ailleurs, un organisme peut dans certains cas décider d'avoir recours à des solutions d'IA développées par des fournisseurs externes. Il convient dans ce cas d'être vigilant car l'utilisation du système peut engager la responsabilité de l'organisme, notamment en matière de protection des données personnelles. Il convient donc d'encadrer par un contrat la relation et les engagements du ou des sous-traitants afin notamment de préciser :

- Le cadre dans lequel le système d'IA doit évoluer
- Les exigences attendues et les modalités de contrôle et d'évaluation
- Les modalités de prise en compte des impératifs de protection des données personnelles
- Les mécanismes de gestion des risques déployés
- Le niveau de documentation souhaité
- Les modalités d'accès et d'utilisation des ressources (possibilité pour le sous-traitant de réutiliser les données, le modèle appris, etc.)
- Les besoins de réversibilité en fin de contrat
- La gestion des questions de propriété intellectuelle

Mettre en œuvre une stratégie de gestion des risques

Comme détaillé dans l'article *P[ã]nser la sécurité des systèmes d'IA*, une stratégie de gestion des risques et de résilience aux erreurs et éventuelles attaques doit être mise en place tout au long du processus de conception, développement, déploiement et mise en production du système d'IA. Elle doit être adaptée spécifiquement au problème à résoudre et alignée avec les impacts potentiels. En pratique, cette stratégie doit être élaborée en fonction de la sensibilité du traitement, des typologies de données traitées (données structurées, image, parole, texte, etc.), des stratégies d'apprentissage (continue, ponctuelle, « une fois pour toute »), des modalités d'exposition du système (interne, public, etc.), des étapes d'évaluation du système, etc.

Une telle stratégie doit permettre de responsabiliser les organismes en leur permettant de construire des systèmes d'IA sûrs et résilients et de démontrer leur conformité aux réglementations en vigueur comme par exemple le Règlement général sur la protection des données (RGPD). Il s'agit donc d'allier analyse de nature juridique concernant la mise en œuvre du système et mise à plat de mesures techniques et organisationnelles nécessaires pour le protéger ainsi que les actifs associés (données, modèles, paramétrisation, etc.). Cette stratégie doit permettre de :

- recenser de manière exhaustive et précise les différents systèmes d'IA déployés dans l'organisme
- sensibiliser et responsabiliser les équipes impliquées
- valider les choix de conception
- mesurer les risques impliqués relativement à la protection des données personnelles (éventuellement présents de façon résiduelle)
- préciser les modalités de sauvegarde et de gestion des données

- assurer la confidentialité, intégrité et disponibilité en déployant les mesures adéquates (restriction d'accès, chiffrement, etc.)
- définir un plan de continuité de l'activité

Les procédures et organisations mises en place pour assurer la conformité au RGPD ou la sécurité des systèmes d'information peuvent notamment être mobilisées pour inclure la gestion des risques spécifiques liés à l'utilisation de systèmes d'IA.

Auteur : Félicien Vallet, responsable IA (CNIL)