Explicabilité des modèles profonds et méthodologie pour son évaluation : application aux données textuelles de Pôle emploi

Gaëlle Jouis

COLLEGE ED MATHEMATIQUES
DOCTORAL STIC
PAYS DE LA LOIRE

Nantes Université

THÈSE DE DOCTORAT DE

NANTES UNIVERSITÉ

ÉCOLE DOCTORALE N° 641
Mathématiques et Sciences et Technologies du numérique de l'Information et de la Communication
Spécialité : *Informatique*

Par
Gaëlle JOUIS

Explicabilité des modèles profonds et méthodologie pour son évaluation

Application aux données textuelles de Pôle emploi

Thèse présentée et soutenue à Nantes Université, le 14 Février 2023
Unité de recherche : umr6004 – LS2N
Thèse N° : 2019-0989

Rapporteurs avant soutenance :

Céline HUDELOT Professeure des universités - CentraleSupélec
Philippe LENCA Professeur des universités - IMT Atlantique

Composition du Jury :

Président :	Gilles VENTURINI	Professeur des Universités - Université de Tours
Examinateur :	Richard DUFOUR	Professeur des Universités - Nantes Université
Dir. de thèse :	Harold MOUCHÈRE	Professeur des Universités - Nantes Université
Co-dir. de thèse :	Fabien PICAROUGNE	Maître de conférences - Nantes Université

Invité(s) :

Alexandre HARDOUIN Scientifique des données - Pôle emploi

Remerciements

Je tiens à remercier tout d'abord mes encadrants. Harold et Fabien, merci de m'avoir guidée, poussée, écoutée. Alexandre, pour m'avoir soutenue depuis mon arrivée à Pôle emploi jusqu'à aujourd'hui. Je ne pouvais espérer meilleur trio pour m'épauler sur ces trois années.

Je remercie chaleureusement mes rapporteurs Céline Hudelot et Philippe Lenca ainsi que les membres du jury Richard Dufour et Gilles Venturini, pour leurs retours constructifs et leurs apports intéressants.

Cette thèse n'aurait jamais vu le jour sans Nicolas Greffard, merci pour tes conseils et nos nombreuses discussions. Une pensée toute particulière pour mes collègues de Pôle emploi. Laurent, pour avoir été présent à chaque instant. À ceux qui ont défendu mon projet, à L'ADS, ceux qui y sont aujourd'hui et ceux qui en sont partis. Je salue la tribu Socle pour m'avoir écoutée tous les jours, même quand je racontais 5 jours de suite exactement la même chose en mêlée. Je remercie également les collègues du réseau pour leur temps, et leurs retours enrichissants.

Merci à l'équipe IPI pour les moments de convivialité, entre grilles et échanges scientifiques. Neslihan et Sophie, pour votre soutien. Je remercie les doctorants, en particulier Noémie, Tristan, Mathieu et Hippolyte, pour l'entraide et les expériences partagées. Merci aux collègues de Capacités pour l'accompagnement dans les expérimentations, et la bonne humeur dans les couloirs. À Polytech de façon plus globale, merci Jean-Pierre et Philippe, pour les petits-déjeuners et les discussions enrichissantes. Je salue également les étudiants que j'ai eu le plaisir d'encadrer en PRED et PTRANS.

Un grand merci à mes parents, mon frère et ma soeur, pour leur amour à toute épreuve et les petits mots qui égaient la semaine. Julie, merci d'avoir été là pour moi malgré mon indisponibilité et ma fatigue. Merci d'avoir rendu mon quotidien plus léger. Merci pour les moments de respiration, Chloé, Solal, Mathilde, Estelle. Pour les nombreuses soirées de confinement seule devant mon ordinateur, mais jamais vraiment isolée, merci à la communauté de Cast&Play et à la Gilde.

Sommaire

INTRODUCTION

L'intelligence artificielle (IA) est aujourd'hui partout dans nos quotidiens. Les modèles complexes tels que les réseaux de neurones sont notamment utilisés pour traiter automatiquement les textes qui nous entourent. La figure 1 présente différentes solutions d'intelligence artificielle. Elle met en avant la corrélation entre meilleures performantes et explicabilité décroissante. De bas en haut, les solutions sont de plus en plus performantes ; on retrouve alors les trois ères de l'IA. La première période des années 1960 est celle des systèmes experts ou "la bonne vieille IA" (*Good Old Fashioned AI*, *GOFAI*), avec des systèmes tels qu'ELIZA [115]. Vient ensuite celle de l'apprentissage automatique avec des systèmes simples tels que les régressions linéaires. Enfin, l'avènement de l'apprentissage profond a contribué à l'entraînement de modèles complexes, résolvant des problèmes pointus [63]. Ce manque d'explicabilité est un problème auquel s'intéressent les communautés scientifique et industrielle.

L'explicabilité est la capacité d'un système à être compris par un humain étant donné un contexte. L'explication peut prendre diverses formes et s'adapter à son receveur. Les informations brutes comme le code source d'un algorithme ou la structure d'un modèle ne sont pas des explications mais de la transparence. Celle-ci ne suffit pas à rendre un modèle explicable, surtout si ce dernier est complexe.

Pour la communauté scientifique la performance des modèles a longtemps été mesurée par des scores de prédiction : précision, rappel, F1-score permettent les comparatifs de modèles d'IA. Toutefois, les scores faibles n'indiquent pas les faiblesses d'un modèle. Le développement de ces modèles est alors contraint aux essais-erreurs. On pourrait se contenter de modèles très performants. [25] dit que les explications sont nécessaires quand les résultats d'un modèle ne correspondent pas aux suppositions des utilisateurs. Pourtant, un modèle avec de bonnes prédictions peut les effectuer pour de mauvaises raisons. Le domaine de l'Intelligence Artificielle Explicable (XAI) a vu son crédit augmenter au fil de la croissance de la taille des modèles profonds. L'explicabilité permet d'améliorer un modèle, ainsi que son adoption par les utilisateurs.

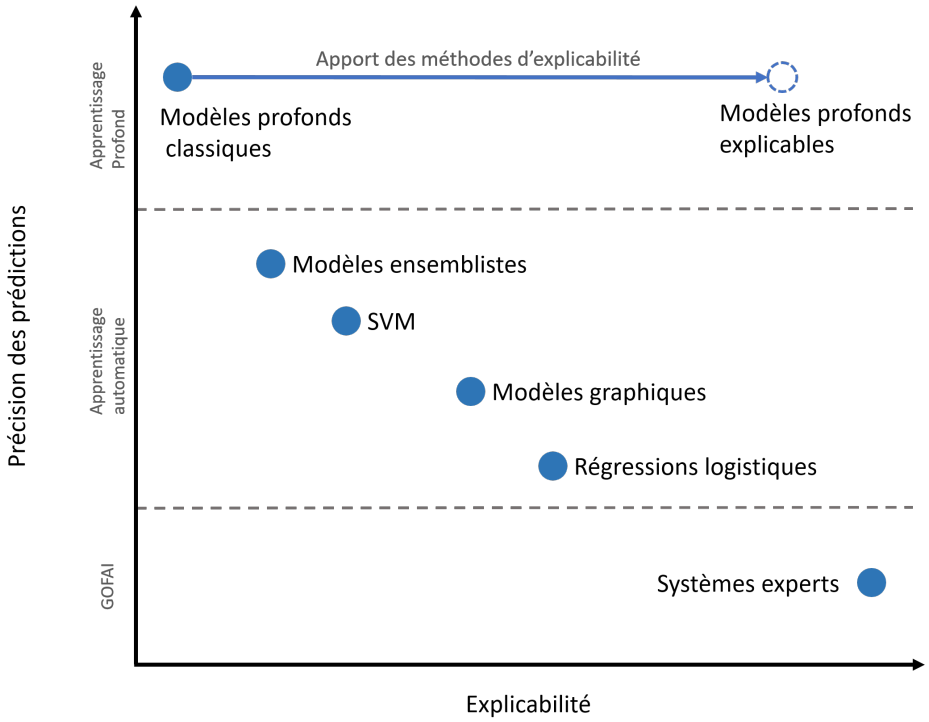


FIGURE 1 – Différentes familles de solutions d'intelligence artificielle, selon leur précision de prédiction, et leur explicabilité. Figure inspirée de [25].

Pour l'industrie et les services publics l'explicabilité est également importante. Un manque d'acceptabilité freinera l'adoption d'outils, tandis qu'une confiance totale rendra difficile la détection de dysfonctionnements d'un modèle d'IA. Cette problématique est d'autant plus présente dans des domaines tels que la conduite autonome, les soins médicaux, et l'accompagnement des personnes. Le besoin de transparence et d'explications est également transcrit dans les législations [28, 34]. Ainsi le Règlement Général sur la Protection des Données (RGPD) [23] indique que tout traitement de données personnelles des citoyens et citoyennes européennes doit être communiqué de manière accessible, en langage clair et net. Toutefois, cette contrainte juridique est aujourd'hui limitée [112]. La loi pour une république numérique [64] exige quant à elle que les traitements concernant un usager en particulier, soient communiqués sur demande de ce dernier. La question se pose alors de trouver le bon compromis entre la performance du modèle et son explicabilité.

Contributions

L'objectif de la thèse est de rendre plus explicables les modèles profonds permettant de traiter des textes. La figure 2 présente l'environnement de notre contribution. Des utilisateurs veulent rendre un service ou bénéficier d'un service. Iels vont pour cela utiliser un outil, lequel peut appeler un modèle pour obtenir le résultat d'une prédiction, classification ou recommandation. Ce modèle complexe est issu d'un entraînement défini par un algorithme et un large ensemble de données, dans notre cas des textes.

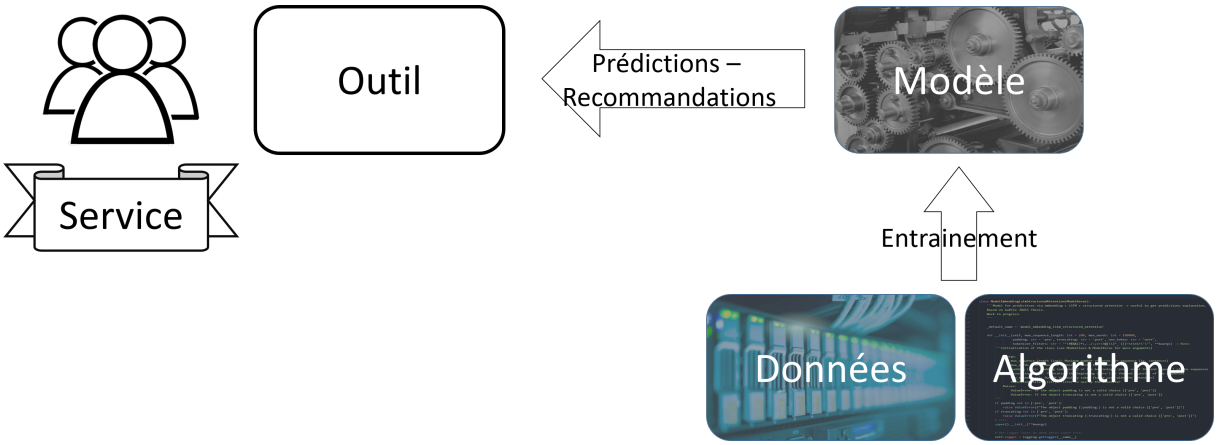


FIGURE 2 – L'environnement des travaux. Des utilisateurs veulent bénéficier d'un service. Iels utilisent un outil qui présente les prédictions d'un modèle. Le modèle est lui même entraîné à partir d'un algorithme et de données.

Nous proposons des explications permettant à l'utilisateur de comprendre les résultats obtenus et le modèle. Ces explications sont présentées sous forme d'interfaces. Les modèles cibles sont les plus complexes : les réseaux de neurones profonds. Nous nous concentrons spécifiquement sur les architectures d'analyse de textes et sur des visualisations prenant en compte les spécificités du langage naturel. Ces travaux sont intégrés à un outil commun aux scientifiques des données de Pôle emploi, et accessibles en open source.

Notre contribution est donc constituée des éléments suivants :

— La création d'un jeu de données de classification de textes avec les explications locales associées,
— L'application de deux méthodes de génération d'explication locales de la littérature : les ancres et le mécanisme d'attention,
— La définition d'un protocole de comparaison de ces explications, avec et sans utilisateurs,
— Un démonstrateur permettant de présenter des interfaces adaptées aux utilisateurs, interfaces définies grâce à un test d'utilisabilité réalisé directement auprès des utilisateurs,
— Une méthode modulaire de génération d'explications globales, via l'aide à la création d'un modèle mental par les utilisateurs.

Plan

Dans le chapitre 1 nous nous intéresserons à la littérature du domaine. Cet état de l'art commence par la définition des méthodes d'explications et présente la diversité des explications possibles. Ensuite nous abordons les méthodes de génération d'explications, et leur évaluation.

Ensuite, dans le chapitre 2 nous présentons les pré-requis à nos travaux, à savoir la collecte et définition d'un jeu de données d'explications de référence. Nous appliquons deux méthodes de génération d'explications locales issues de l'état de l'art à nos données de référence. Enfin, nous présentons notre démonstrateur, permettant la visualisation de ces explications locales.

Le chapitre 3 définit le protocole de comparaison d'explications locales. Dans un premier temps nous évaluons la forme des explications grâce à un test d'utilisabilité auprès d'un panel restreint d'utilisateurs. Nous évaluons ensuite le contenu des explications. Deux contextes majeurs sont pris en compte : la présence ou absence d'utilisateurs experts dis-

ponibles.

Dans le chapitre 4 nous introduisons une méthode modulaire d'explication globale d'un modèle, par l'aide à la définition d'un modèle mental pour l'utilisateur. Nous implémentons une stratégie avec un effort de réduction des calculs nécessaires. L'application sur nos données réelles permet de prouver son potentiel et d'en déterminer les limites.

Enfin, nous mettons en avant dans le chapitre 5 le lien fort entre les travaux effectués et leur contexte industriel. Nous présentons le cadre logiciel de développement de Pôle Emploi, auquel sont ajoutées les fonctionnalités d'explications locales et globales présentées. La charte éthique de l'établissement est présentée ainsi que l'intégration de nos travaux à sa mise en œuvre. Enfin, nous proposons une ébauche de guides destinés aux industriels, notamment gestionnaires de projets, leur permettant de mieux définir leurs besoins en explicabilité, et les méthodes de la littérature qui seraient les mieux adaptées à ces besoins.

Chapitre 1

Génération et évaluation des explications dans la littérature

Dans ce chapitre

Ce chapitre présente l'état de l'art en XAI, divisé en trois sections. La première partie reprend la typologie des explications d'IA de la littérature. La seconde porte sur la génération de ces explications, et la troisième porte sur leur évaluation. Nous mettons en avant la multiplicité des méthodes de création des explications. En effet, il n'y a pas une méthode qui conviendrait à tous les usages, ce qui a conduit à de nombreuses propositions de la communauté XAI. De même, il n'existe pas de consensus sur la manière d'évaluer les explications, que ce soit sur le protocole ou les métriques présentées. Nous présentons un large panel de propositions, ainsi que leurs avantages et inconvénients.

Ce chapitre présente diverses méthodes de génération d'explication de la littérature, ainsi que des méthodes d'évaluation de ces explications. Cet état de l'art aborde deux problématiques de l'explicabilité de l'intelligence artificielle (XAI). La première est : comment obtenir un outil d'intelligence artificielle explicable ? La seconde est : comment évaluer la méthode d'explicabilité d'un outil ?

Une explication peut être créée via des méthodes variées et peut prendre diverses formes. Pour un même outil, il est possible de chercher des explications différentes répondant à des objectifs variés. Ces besoins différents ont amené à la création d'un grand nombre de solutions dans la littérature.

Évaluer une explication nécessite donc de s'assurer que celle-ci est bien conçue par l'expliqueur, mais également bien perçue par le receveur. Les méthodes d'évaluation sont

tirées des explications elles-mêmes, basées sur le ressenti subjectif d'un utilisateur ou sur l'impact objectif d'une explication dans une tâche. Le choix d'une évaluation adaptée rend compte non seulement de la qualité d'une explication mais aussi de son adéquation avec les objectifs de l'outil global dans la réalisation d'une tâche.

Nous définissons d'abord les différents types d'explications en section 1.1. Nous présentons ensuite en section 1.2 les principales méthodes d'explication, et en section 1.3 les évaluations de ces explications.

1.1 Typologie des méthodes d'explication

Le choix d'une méthode est notamment guidé par des contraintes techniques ou d'organisation du projet : nature du modèle à expliquer, détection de la problématique d'explicabilité avant ou après conception du modèle, etc. Cette section passe en revue quatre caractéristiques permettant de définir une méthode d'explication : la portée, la stratégie, le format, et les données d'application. La typologie est inspirée d'états de l'art fondateurs dans le domaine [6, 15, 25, 39, 44, 49, 77].

1.1.1 Portée : globale vs. locale

La portée des explications données par une méthode peut être locale ou globale. Ces deux types d'explications n'ont pas le même objectif. Les facteurs permettant de choisir entre une explication globale ou locale sont :

— le but de l'explication,
— les questions auxquelles elle doit répondre,
— le public ciblé,
— le contexte de réception de l'explication.

Pour chacun de ces points, l'une ou l'autre portée sera à privilégier.

Globale Une explication globale s'applique au modèle dans son ensemble. Elle est vraie pour toute entrée, et permet de comprendre le comportement du modèle dans les cas nominaux et extraordinaires. Son but est de donner une vision globale du modèle, pour tout résultat possible. Elle permet de construire la confiance dans le modèle et valider la mise en place de ce dernier.

Les questions abordées sont :

— Quels sont les points faibles de mon modèle ?

— Dans quels cas puis-je l'utiliser sans crainte ?
— Dans quels cas dois-je me méfier ?
— Comment a-t-il appris ?
— Quelles sont les données d'entraînement pertinentes ?
— Quel est l'intérêt/la contribution de cette sous partie de mon modèle pour la réalisation de la tâche ?
— comment modifier mes données d'entraînement afin de gagner en performances ?

Le public cible de ces explications est constitué de scientifiques des données, d'experts du domaine d'application, de décideurs et d'utilisateurs finaux. [74] Les explications globales sont particulièrement adaptées pour mesurer les performances et améliorer le modèle, définir ses limites d'applications, valider la mise en production du modèle, et instaurer une phase d'appropriation pour les utilisateurs finaux.

Locale Une explication locale s'applique pour un modèle et une instance donnée. Elle permet de comprendre pourquoi pour une entrée définie, le modèle donne un résultat et non un autre. Elle met en lumière le raisonnement du modèle dans un cas précis, voire dans un groupe d'instances plus étendu mais défini. Elle permet de construire la confiance dans le résultat, sans généraliser au reste du modèle.

Les questions auxquelles elle doit répondre sont :

— Pourquoi j'obtiens ce résultat avec cette entrée ?
— Que changer en entrée pour changer le résultat en sortie ?
— Que se passe-t-il si ma donnée d'entrée change sur cette variable ?

Ces explications ciblent en priorité les utilisateurs finaux, ainsi que les experts du modèle. Ce format est adapté pour une intégration aux conditions réelles d'utilisation d'un modèle. Il se concentre sur le résultat obtenu par l'utilisateur. Il est également indiqué si l'utilisateur a peu de temps pour lire l'explication.

En résumé, une explication globale demande plus de temps qu'une explication locale. Mais la première permet de faire la lumière sur tout le modèle, là où l'explication locale donne confiance dans un exemple en particulier. Ainsi, une explication globale sera plus pertinente pour un audit ou une évaluation globale du modèle. Une explication locale sera plus efficace à l'usage en conditions réelles, ou pour l'étude approfondie de cas spécifiques. Outre la portée, différentes stratégies d'explicabilité peuvent être mises en place.

1.1.2 Stratégie

On peut classer ces méthodes d'explication par la transparence du système étudié, ce qui donne les catégories suivantes.

— Expliquer un modèle boîte noire au travers de ses entrées et sorties, soit en observant directement l'influence des premières sur les secondes, soit en créant un modèle interprétable mimant la boîte noire. Ces méthodes sont totalement indépendantes de la structure interne du modèle boire noire.

— Observer les mécanismes internes d'un système boîte grise après son entraînement ; afin d'y détecter des schémas et les interpréter, ces méthodes sont donc dépendantes de l'architecture interne du modèle observé.

— Concevoir un modèle ou une solution transparente boîte blanche de par son architecture ; en lui associant des contraintes compréhensibles pour un humain, sous forme de règles ou en générant des explications en plus du résultat attendu.

Ces diverses stratégies sont possibles selon le moment de vie du projet.

1. Le projet est en phase d'initialisation. Alors les modèles transparents seront à privilégier, puisque les explications générées seront les plus fidèles au raisonnement du modèle.
2. Le modèle est déjà conçu, et accessible. Il est possible d'utiliser une méthode dépendante de l'architecture, si cette dernière le permet.
3. Si un modèle est hors d'atteinte, car c'est un outil propriétaire appartenant à un développeur tiers, ou l'outil est accessible mais l'architecture ne permet pas d'utiliser une méthode type boite grise. Dans ce cas un fonctionnement boite noire est possible.

Il est possible de remplacer l'utilisation d'un modèle d'apprentissage profond par un modèle transparent. Toutefois, cette solution sort des limites d'études des travaux présentés ici. Ainsi, la stratégie est à adapter selon l'accessibilité et la phase de conception du modèle. Elle suit également les éventuelles contraintes du modèle, si par exemple le compromis ne peut être fait sur le taux d'erreur du modèle et que l'architecture ne peut être modifiée. Une fois la stratégie déterminée, il reste le format d'explication à sélectionner.

1.1.3 Format d'explication

L'explication peut prendre différentes formes, regroupées en quatre grandes familles :

1. basées sur les variables d'entrée,
2. sous formes de règles,
3. les exemples,
4. les explications générées qui ne sont ni des règles ni des exemples.

La figure 1.1 présente ces quatre type d'explications, en se basant sur un cas d'usage de Pôle emploi pour illustration.

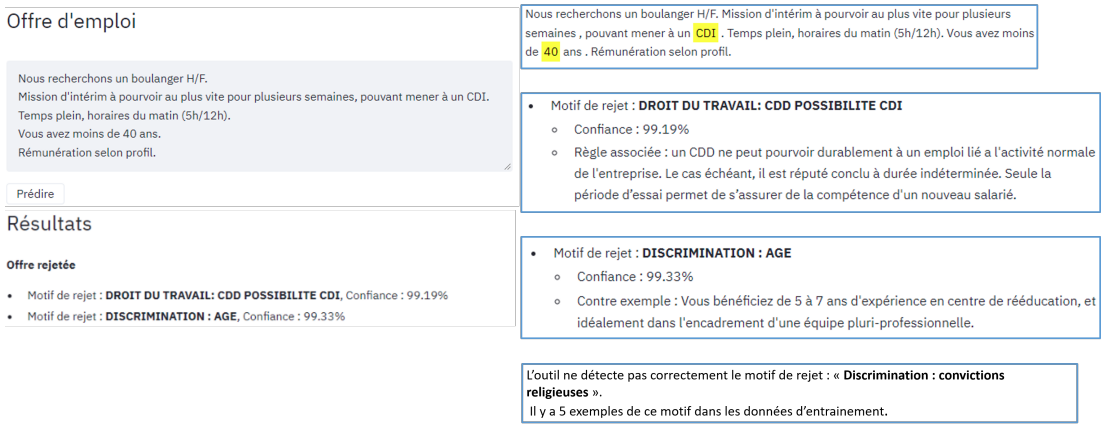


FIGURE 1.1 – A gauche, exemple d'offre d'emploi rejetée et motifs associés. A droite, les différents formats d'explications possibles. De haut en bas : les variables, règles, exemples, et explications générées.

Les explications basées sur les variables d'entrée mettent en avant une corrélation ou un lien de causalité entre une entrée et une sortie. Ces variables d'entrées peuvent être une colonne de données tabulaires, un n-gramme dans un texte, ou un pixel ou superpixel dans une image. Ce type d'explication est très utilisé pour l'explicabilité locale, car il s'appuie directement sur une instance donnée en entrée. Ces explications se font également par contraste : en se concentrant sur les variations menant à des décisions algorithmiques différentes.

Une explication peut se faire en explicitant une règle logique. Cette règle peut mettre en avant une relation de cause à effet du phénomène modélisé, ou sur une corrélation entre une variable et une sortie associée. Dans ce dernier cas, la règle est plus généralisable que l'explication basée sur les variables. Les explications sous forme de règles peuvent notamment être extraites d'un arbre de décision, ou de contraintes appliquées au système de décision, auquel cas ce sont des explications globales. Elles peuvent aussi être des explications valides dans un périmètre local précis.

Les explications basées sur les exemples sont basées sur des instances réelles du phénomène modélisé. Elles évitent ainsi de s'appuyer sur des données hors distribution, adversaires ou jamais vues par le modèle lors de l'apprentissage. Lorsque c'est un contre-exemple qui est présenté, c'est à dire une instance menant à un résultat différent, alors l'explication est contrastive. Il est également possible de montrer une instance la plus différente possible mais menant au même résultat qu'une instance d'origine, on parle alors d'exemple semi-factuel.

Enfin, les explications générées ne sont ni des variables, ni des règles, ni des exemples. Elles peuvent correspondre à de la documentation sur le modèle, telle que l'analyse des erreurs du modèle, ou des indications sur ses limites. Elles peuvent prendre la forme de motifs générant de fortes activations de couches d'un réseau de neurones, ou encore des textes et images générés, sans que ces derniers ne ressemblent à un exemple.

1.1.4 Données d'applications spécifiques

Les explications peuvent s'appliquer aux textes, images et données tabulaires. Les textes et images sont des données spécifiques, car analysés directement par les interlocuteurs humains.

Ainsi, le langage naturel est lu, entendu et parlé, et ce dès l'enfance, pour la plupart des humains. Un mot manquant ou une règle de grammaire non respectée sera alors jugée sévèrement. D'un point de vue technique, les textes sont divisibles en mots ou en n-grammes. Un n-gramme est un ensemble de n entités syntaxiques consécutives, des mots ou des caractères. La figure 1.2 illustre ce concept sur un court exemple. Au-dessus de la phrase sont extraits deux bigrammes (ou 2-grammes) de lettres : "un" et "hr", parmi les 16 possibilités. En dessous sont extraits les bigrammes de mots.

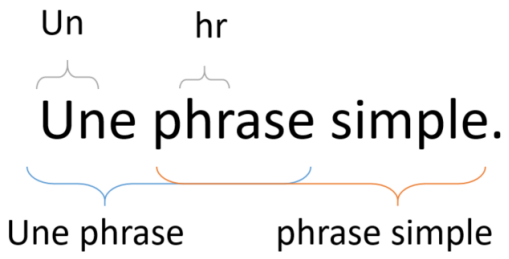
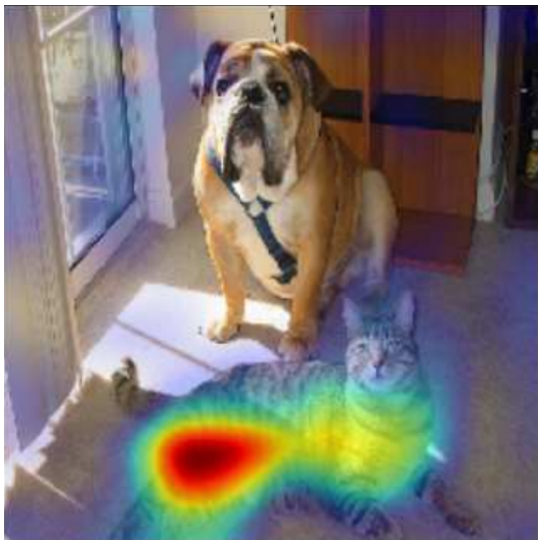


FIGURE 1.2 – Exemples de bigrammes de caractères en haut et de mots en bas.

Pour être traités par un algorithme, les textes doivent être transformés en vecteurs. Les mots peuvent être utilisés directement pour vectoriser le texte, avec des méthodes

d'encodage à chaud ("one-hot encoding") ou des techniques de sac de mots telles que le TF-IDF ("Term Frequency - Inverse Document Frequency"). Enfin, de façon plus avancée, les textes peuvent être vectorisés via un plongement de mots, réseau de neurone spécialisé dans cette tâche.

Les images sont représentables par un ensemble de pixels de couleurs, sur les trois canaux rouge, vert, bleu. Techniquement une image est donc une matrice en trois dimensions : hauteur, largeur, couleur. Il est possible de représenter une image en considérant chaque pixel comme une unité. Les pixels peuvent également être rassemblés par blocs, selon leur similarité. Ces blocs sont des *superpixels*. Ils permettent de découper l'image en blocs porteurs de sens. La figure 1.3 montre deux images avec des explications au niveau des pixels et superpixels.



(a) Explication au niveau des pixels avec la méthode Grad-CAM [99]

(b) Explication au niveau des pixels avec la méthode des ancres [91]

FIGURE 1.3 – Différentes visualisations d'une explication sur une image.

Enfin, les données tabulaires ont l'avantage de nécessiter moins de transformation entre leur format lisible par un humain et leur intégration dans les modèles. Toutefois, ces données sont moins naturelles à lire comparées aux textes et images.

Les données ne sont pas au choix, à moins de modifier la structure du problème initial. Toutefois elles imposent des contraintes de visualisation et possèdent leurs spécificités propres. Les portées, stratégies, formats, et les données à disposition pour générer des explications sont désormais définies.

Ces différentes caractéristiques des explications impliquent un nombre important de

méthodes de génération d'explications. Ces méthodes sont adaptées à certaines caractéristiques.

1.2 Comment générer une explication ?

Les méthodes connues sont présentées en se basant sur le découpage par stratégie, à savoir :

— expliquer un modèle boîte noire via des méthodes indépendantes du modèle,
— observer un modèle boîte grise avec des méthodes dépendantes du modèle,
— concevoir un modèle transparent.

1.2.1 Explications indépendantes du modèle

Ce type de méthode a pour principal avantage de pouvoir être utilisé sur tous les modèles. Cela évite de se contraindre techniquement dans le choix d'un modèle. Les explications sont le plus souvent basées sur les variables d'entrées. Les explications sont générées en considérant le modèle comme une boite noire. Elles relèvent donc de la corrélation plus que de la causalité.

Distillation de connaissance Une manière de rendre les modèles interprétables est de se concentrer sur des architectures simples. La distillation de connaissances dans les modèles permet d'entraîner un ensemble complexe de réseaux de neurones et transférer les structures apprises à un réseau de neurones plus simple [47]. Instinctivement, il est tentant de se dire qu'un réseau plus simple pourrait être plus interprétable. Les auteurs de [30] proposent de mesurer l'interprétabilité comme étant le ratio entre la performance du modèle simple et du modèle complexe. En appliquant ces mesures sur un cas d'étude avec un réseau de neurones complexe et un réseau de neurones simplifié, ils en déduisent que la distillation améliore la robustesse, c'est à dire à la résistance aux attaques par modifications subtiles des entrées du modèle, au détriment de l'interprétabilité. En revanche, la distillation est également utilisée pour créer des arbres de décision [68, 118] ou des arbres boostés par le gradient [20] à partir de réseaux de neurones, ce qui revient à créer un module d'explication sous forme d'arbres de décisions. Si l'approche est décrite pour des réseaux de neurones, elle est applicable à d'autres modèles.

Importance des variables De nombreuses méthodes fonctionnent sur la quantification de l'importance des variables. Les vecteurs d'explication locale quantifient l'importance de chaque variable d'entrée pour une instance donnée [9]. Un vecteur d'explication indique dans quelle direction changer une variable pour que le modèle change de résultat. En observant un ensemble de vecteurs, on peut également avoir une visualisation plus globale du modèle. Toutefois, ces vecteurs sont difficilement interprétables pour les utilisateurs, surtout dans le cas de l'analyse sémantique où la dimension est la taille du vocabulaire du corpus.

Les valeurs de Shapley donnent un aperçu de la contribution d'un élément dans un ensemble par rapport à un résultat final [104]. La dimension des explications obtenues est égale à la dimension des données en entrée. Elles nécessitent donc un traitement a posteriori pour en retirer des explications claires. Le calcul des valeurs de Shapley nécessite d'avoir accès au jeu de données d'entraînement du modèle, ce qui est une limitation forte et sous-entend un calcul éventuellement long. Pour limiter le temps de calcul, une estimation des valeurs de Shapley est donnée dans le module SHAP ("SHapley Additive exPlanation") [71].

La méthode Randomized Input Sampling for Explanation ("RISE") masque aléatoirement des variables en entrée du modèle, pour déterminer les variables importantes [82]. Ces travaux sont approfondis pour créer des cartes de saillance (saliency maps) adaptées à chaque classe du modèle [106]. Les mesures d'influence quantitative des entrées donnent également le lien entre les données en entrée et les sorties d'un algorithme [27]. Cette méthode est intéressante pour sa rapidité de calcul. La méthode COnstrained feature perturbation and COunterfactual instances ("COCO"), calcule ces poids de variables en contraignant les perturbations dans une direction choisie [36]. Malgré ces contraintes, cette méthode nécessite un long temps de calcul. Les contraintes sur les perturbations, basées sur les connaissances fonctionnelles des données, permettent également de créer des explications plus concises [43].

Approximation linéaire locale La méthode Local Interpretable Model-agnostic Explanations ("LIME") mesure l'importance des variable en effectuant des perturbations aléatoires [90]. LIME fonctionne par approximation linéaire du modèle autour d'une instance donnée. C'est ce modèle linéaire qui est ensuite utilisé pour générer des explications. Ces explications correspondent aux variables d'entrée qui impactent le plus la sortie du modèle. Pour l'analyse de texte, ce sont les mots du texte associés à une quantification

de l'influence, positive ou négative, sur la réponse du modèle.

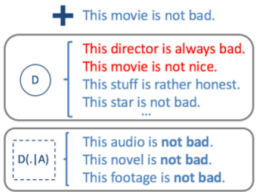
Dans le même article, les auteurs présentent SP-LIME, une méthode d'explication globale se basant sur les explications locales de LIME. Un ensemble d'exemples et explications locales associées est "sélectionné judicieusement"[90], étant donné un nombre B prédéfini d'instances à présenter à un utilisateur. Cette sélection s'appuie sur la recherche d'un ensemble optimal d'exemples couvrant les variables d'entrées les plus importantes. L'ensemble des exemples doit être expliqué par Lime avant d'être sélectionné par SP-Lime. Cette méthode est intéressante car elle est généralisable au-delà de LIME [91]. Par contre, elle nécessite de générer en amont toutes les explications des instances d'un ensemble de données pour pouvoir les sélectionner ensuite.

L'avantage de cette approximation linéaire est la simplicité de la création de l'explication. Toutefois il faut faire l'hypothèse forte que le modèle se comporte linéairement autour de l'instance expliquée. De même, les explications manquent de stabilité, et deux entrées très proches peuvent avoir des explications très différentes. Il est ainsi difficile pour un utilisateur humain de savoir à quel point ces explications sont généralisables. Les perturbations de LIME sont adaptables à des domaines spécifiques, comme cela est montré dans le cadre de l'analyse d'images [103].

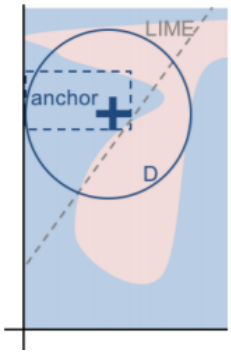
Ancres Les auteurs de LIME ont proposé une amélioration de leur méthode, les Ancres [91]. En conservant l'idée d'approximation locale du modèle, les auteurs sont passés d'une approximation par un modèle linéaire à une explication sous forme de règle. L'idée est de mieux définir le contexte dans lequel l'explication générée est valable. Soient un modèle $f : X \to Y$, une instance $x \in X$, un résultat $y \in Y$ choisi, et une ancre A associée. A est une condition telle que si x respecte cette condition, alors la probabilité que $f(x) = y$ est grande. L'ancre est construite de sorte à maximiser cette probabilité, il est toutefois possible que $f(x) \neq y$. On note $D(.|A)$ l'ensemble des $x \in X$ qui respectent la condition A. Une ancre intéressante s'applique à un ensemble $D(.|A)$ le plus grand possible relativement à X. Si les entrées sont des textes, une ancre est un ensemble de mots ou n-grammes.

Dans l'exemple illustré en Fig. 1.4, le modèle étudié classe des phrases selon deux catégories : "positive" et "négative". L'instance d'origine est la phrase "This movie is not bad", et est classée "positive". L'ancre associée est la règle $A = \{not, bad\} \to Positive$. L'ensemble $D(.|A)$ de la Fig. 1.4a, représenté par un rectangle dans la Fig. 1.4b, regroupe les entrées possédant les variables de l'ancre. Dans notre exemple, $D(.|A)$ correspond

aux textes comprenant les mots “not” et “bad” de l’ensemble des variations de l’instance d’origine (ensemble D de la Fig. 1.4). L’ensemble D est obtenu en appliquant des variations cohérentes à l’instance d’origine. Appliqué à l’exemple, cela correspond à remplacer un ou plusieurs mots de la phrase par des mots de nature similaire. Remplacer un adjectif par un autre adjectif est une variation cohérente de l’instance d’origine.



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



(b) Représentation de l’entrée (croix), et des ensembles D (cercle) et $D(.|A)$ (rectangle)

FIGURE 1.4 – Textes similaires à une entrée D et son sous ensemble $D(.|A)$ correspondant à l’ancre $\{not, bad\} \rightarrow Positive$. Source : [91]

Pour sélectionner une ancre, les auteurs choisissent de maximiser deux paramètres. Le premier est la précision, qui est maximale si les éléments de l’ensemble $D(.|A)$ ont la même sortie que l’instance d’origine. Le second est la couverture, soit la taille de l’ensemble $D(.|A)$ par rapport à l’ensemble D. Une ancre avec une forte précision est une explication fidèle au modèle boîte noire. Si elle a une bonne couverture, une ancre est assez généralisable.

Le travail autour des ancres met en avant la relation entre l’utilisateur et l’explication. La limite de LIME contournée par les Ancres est la difficulté pour l’utilisateur à déterminer la validité d’une l’explication donnée. Ce faisant, l’explication donnée, à savoir une règle, est également plus facile à aborder qu’un ensemble de poids comme le résultat de base de LIME.

Les explications indépendantes des modèles sont une première approche en considérant les modèles boîtes noires, mais il s’agit plus de mettre en avant des corrélations entre les

entrées et les résultats du système. Pour aller plus loin, il est possible de s'appuyer sur la structure du modèle pour générer des explications. La *boîte noire* devient alors une *boîte grise*.

Génération de contre-exemples Les méthodes basées sur la perturbation telles que LIME [90] et RISE [82] créent des données hors distribution (*out of distribution*, OOD). Ces échantillons OOD peuvent être mal classés par les modèles, mais peuvent aussi être assez inhabituels pour les utilisateurs finaux, perdant ainsi la clarté des explications. Ceci est particulièrement vrai dans le domaine du langage naturel, où le texte OOD devient très rapidement dénué de sens. Cette déviance par rapport à la distribution originale des données peut être mesurée [87]. Des travaux tentent de contraindre les perturbations, en veillant à ce que les données générées soient cohérentes avec la distribution des données d'apprentissage. Cela peut être réalisé en trouvant le meilleur masque de perturbation [17] et en remplissant ce masque de la meilleure façon possible pour créer une entrée cohérente [3].

Pour aller plus loin, des entrées cohérentes peuvent être générées directement via des Réseaux antagonistes génératifs (Generative Adversarial Networks, GAN), en produisant des exemples factuels, des exemples semi-factuels et des exemples contrefactuels [18, 59]. Ces points de données générés sont similaires à des données d'entraînement par conception, évitant la plupart des problèmes d'OOD, selon les performances du GAN. De plus, ces exemples aident les utilisateurs finaux à reconstruire un modèle mental, c'est à dire une représentation mentale du modèle. En effet, ils rendent tangibles les limites de décision, situées entre les exemples semi-factuels et contrefactuels. La mise en évidence du delta nécessaire pour franchir cette frontière peut être obtenue en calculant la différence entre les instances factuelles et contrefactuelles [18].

Cependant, le gain en explicabilité apporté par certaines méthodes de pointe se fait au détriment du temps de calcul, et de la sobriété énergétique. A titre d'illustration, citée par [17], une approche d'explication avec un masque de perturbation cohérent peut prendre "environ une minute sur un seul GPU pour terminer une image".

1.2.2 Explications dépendantes du modèle

Les explications dépendantes du modèle sont basées sur l'observation des paramètres du modèle après son entraînement. Cette observation, pour conserver la métaphore de la boite noire, revient à ouvrir cette boite et regarder à l'intérieur. Les explications sont

alors plus fidèles au fonctionnement du système étudié que les méthodes indépendantes du modèle. Toutefois, cette approche contraint fortement le choix du modèle.

Réseaux à convolutions Dès 2013, des travaux sur les réseaux de neurones proposent des visualisations de leur fonctionnement, dans le cadre de la classification d'images. Dans [123], les auteurs proposent de mieux appréhender le fonctionnement des réseaux à convolution (Convolutionnal Neural Networks, CNN), par une visualisation directe des motifs d'activation des neurones par couche. Ils utilisent pour cela un réseau de neurones appelé *Deconvolutionnal Network*. Ils obtiennent des motifs reconnus par chaque couche, motifs simples sur les couches basses et plus semblables aux classes détectées sur les couches hautes. Les auteurs vérifient également le comportement de leur modèle en masquant certaines parties des images et en observant les éventuelles variations de classification induites.

Toujours sur les CNN, la méthode de cartographie d'activation de classe, (Class Activation Mapping, CAM) [124] est proposée. Elle permet de générer des cartes de chaleur des endroits de l'image aidant à la détection d'une classe en particulier. . La méthode Grad-CAM [99] mélange les visualisations à celles de [102, 123], afin d'obtenir les parties de l'image ainsi que les motifs précis permettant la classification (Fig 1.3a). Ces travaux sont repris par [19] pour donner l'amélioration *Grad-CAM++*. Dans la continuité de cas travaux, des concepteurs ont été associés à ces trois méthodes [86]. Un concepteur est un motif de changements d'états de neurones. Les auteurs proposent grâce aux concepteurs de prendre en compte les éléments en faveur et en défaveur de la classification. Toutefois, les cartes de chaleur issues de cette méthode sont plus diffuses. Des travaux sont également effectués sur les cartes de saillances (saliency maps) des images [101]. Ces travaux permettent d'avoir une indication fidèle du fonctionnement du modèle. Ces techniques sont plutôt orientées analyse d'image.

Long-Short Term Memory (LSTM) Dans la même philosophie, les auteurs de [58] essaient de comprendre les forces et les limites des réseaux de neurones de type LSTM, appliqués à l'analyse de textes. Pour leur analyse, ils génèrent des visualisations sur des motifs spécifiques dans les données entraînant les activations de certaines cellules. Un exemple donné par les auteurs et présenté en figure 1.5 concerne l'activation de certaines cellules en fonction des caractères rencontrés dans un texte. La visualisation met en évidence la détection du texte entre guillemets.

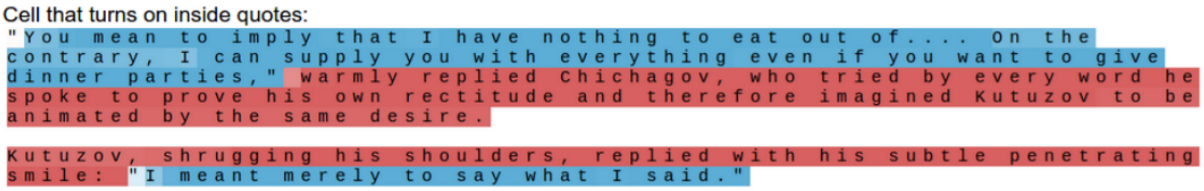


FIGURE 1.5 – Activation d'une cellule en fonction des guillemets dans le texte. Source : [58]

Ce type de réseau de neurones est largement utilisé dans l'analyse sémantique. Les visualisations peuvent donner une idée précise du fonctionnement du modèle. Toutefois, elles sont complexes ou nécessitent un travail de recherche et d'analyse, d'autant plus si on considère chaque cellule. Pour donner un ordre d'idée, des structures de LSTM de la littérature peuvent avoir 300 [66] ou encore 512 [58] cellules. Il convient alors de toutes les explorer pour trouver, pour une petite partie d'entre elles, des activations significatives. Si ce type d'approche permet de mieux comprendre les indices utilisés par le réseau, il ne s'agit pas forcément d'une explication de la décision finale. L'inspection neuronale profonde (Deep Neural Inspection, DNI) repose sur les activations de la couche d'états cachés d'un LSTM de taille restreinte, afin de vérifier le respect de fonctions hypothèses définies par des experts des données, telles que la recherche d'éléments spécifiques dans les données : ponctuation pour du texte, couleurs pour les images, etc. [98].

Décomposition du modèle La décomposition pixel par pixel (*Pixel-Wise Decomposition*, PWD) est une stratégie permettant d'expliquer les résultats d'un classifieur d'images en créant une carte de chaleur des pixels les plus pertinents pour une prédiction donnée [8]. Pour calculer la pertinence R (*Relevance*) de leurs variables d'entrée, dans leur exemple des pixels. Ils décomposent la prédiction $f(x)$ comme étant la somme des contributions des neurones de la couche précédente et appliquent itérativement cette propriété jusqu'à arriver à la couche d'entrée de leur réseau. Les auteurs décrivent deux manières d'y parvenir. La première méthode est la propagation de pertinence couche par couche (*Layer-wise Relevance Propagation*, LRP), un concept regroupant diverses solutions de décomposition respectant certains critères. La seconde est une approche basée sur la décomposition de Taylor, qui permet une approximation de la propagation de pertinence couche par couche, en s'appuyant sur le principe de décomposition de fonctions pour décomposer directement le classifieur f. Ces travaux sont approfondis dans [79] où les auteurs proposent la *Deep Taylor Decomposition*, qui est une adaptation de la décomposition de Taylor, appliquée non pas au modèle entier mais à chaque fonction de pertinence $R_j(x_i)$ entre un neurone

j d'une couche n et les neurones x_i de la couche $n-1$. Dans ce dernier article les auteurs illustrent leurs travaux avec des réseaux de neurones classant des images, mais ce type de méthode peut être appliqué à d'autres modèles et peut également être élargi à d'autres types d'entrées, comme les textes dans [79]. L'analyse des éléments du modèle peut être encore plus visuelle comme avec les graphes de flux de données intégrés à la librairie de développement de modèles Tensorflow [117]. L'apprentissage profond de variables importantes (Deep Learning Important FeaTures, DeepLIFT) [100] utilise la rétropropagation des contributions des neurones, similairement à la LRP [8]. Les auteurs définissent ces contributions comme positives ou négatives, relativement à une contribution de référence [100]. Il est également possible de déterminer des vecteurs concepts, permettant de proposer des exemples correspondant à des concepts sémantiques pertinents sur le plan fonctionnel [60].

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

1.2.3 Modèle Interprétable

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

Architectures simplifiées Dans le courant de simplification des réseaux, des expérimentations mettent en évidence l'efficacité d'architectures de réseaux de neurones simplistes mais capables de résoudre des problèmes complexes, comme le stationnement d'une voiture miniature [45]. Ce type de réseau possède une topologie inspirée du système nerveux du ver *C. elegans*. Le papier définit un réseau ainsi constitué de 12 neurones, en l'entraînant sur la tâche de stationner un robot. En observant les activations des neurones en fonction des phases, les auteurs mettent en lumière le rôle des neurones dans chaque phase de la tâche accomplie, en mettant en évidence par exemple les neurones s'activant lorsque le robot doit tourner à droite. Ces activations sont interprétables notamment parce que le réseau est composé de peu de neurones ; cela facilitant grandement l'analyse des activations. Cette approche permet ainsi de réaliser un travail similaire à [58], qui analyse

les activations de cellules LSTM, mais sur un nombre réduit de neurones.

Considérant la représentation de modèles sous forme de fonctions, l'approche Model Learning with Personalized Interpretability Estimation (ML-PIE) permet à l'utilisateur de choisir successivement les fonctions qui lui semblent les plus explicables, permettant la génération d'un modèle spécifiquement adapté à leur besoin [111].

Mécanismes d'attention Les mécanismes d'attention dans les réseaux de neurones sont une manière de rendre les modèles directement plus interprétables[10]. L'attention est appliquée dans diverses architectures de réseaux, dont les transformeurs qui reposent essentiellement sur ce mécanisme [1, 109]. Dans [66], les auteurs créent un plongement de mots via un réseau avec une partie LSTM et une partie basée sur l'attention. Chaque phrase est représentée par une matrice $M = AH$, où A est la matrice d'attention et H les états cachés de la couche LSTM. Les vecteurs d'attention a composant A vont se concentrer sur des aspects différents de la phrase. En sommant et normalisant (par une fonction softmax) tous les vecteurs d'attention a, les mots fortement considérés par le plongement ressortent avec les poids les plus forts. Cette solution permet donc d'avoir une visualisation claire des variables importantes en entrée du réseau, en observant les paramètres du modèle. Un exemple de visualisation est présenté dans le cadre de la traduction de textes dans [81] (Fig. 1.6). Elle met en avant l'inversion des mots entre l'entrée en anglais ("*european economic area*") et la traduction française ("*zone économique européenne*"). La visualisation des poids d'attention est également utilisée dans [114] afin de valider l'intérêt de leur topologie de réseau, également composé d'un LSTM et d'une couche d'attention. Le principal intérêt de l'analyse de l'attention est qu'il n'y a pas besoin de calculs supplémentaires d'une métrique spécifique une fois le modèle entraîné, contrairement à [8] par exemple.

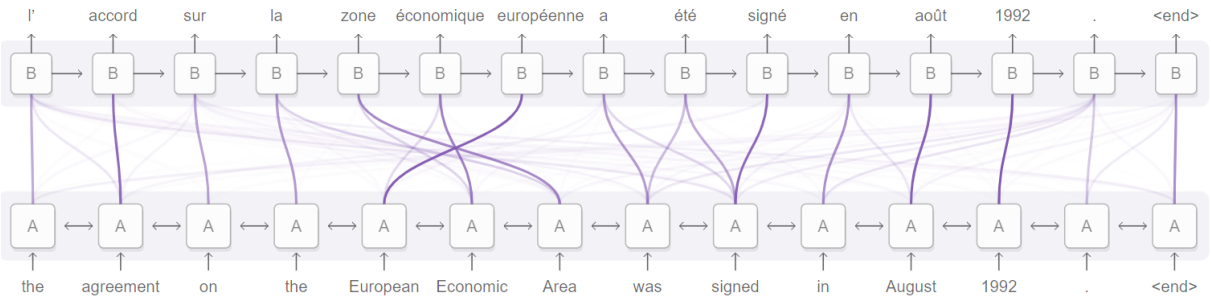


FIGURE 1.6 – Visualisation de l'attention pour une tâche de traduction. Source : [81]

Un modèle peut également utiliser l'attention pour enrichir les données d'apprentissage. Une méthode possible est d'associer les éléments d'attention à des prototypes de données connus qui permettent la prise de décision [13]. Le modèle est ainsi basé sur le raisonnement humain. Une autre façon de faire consiste à superposer des données et des cartes d'attentions issues de l'attention visuelle d'humains effectuant la même tâche que le modèle à entraîner [80]. Des mécanismes similaires ont été développés sur les modèles linéaires généralisés (Generalized Linear Models, GLM) afin d'en extraire des poids d'attention [92].

Génération d'explications Certains systèmes peuvent également générer d'eux même des explications autour d'une décision. C'est le cas de [24] où les explications d'un système de recommandation sont générées sous forme de critiques d'utilisateurs ("*I wouldn't recommend it.*") via un LSTM. Le principe de l'expérimentation est de reconstruire une critique que produirait un utilisateur avec un texte le plus naturel possible. Les auteurs évaluent leurs explications avec des métriques de lisibilité de textes tel que le score *Flesch-Reading-Ease*. Si le système de recommandation n'est pas nécessairement un réseau de neurones, un système similaire peut être appliqué à tout système de prédiction basé sur des textes. Ces explications générées peuvent être plus globales, comme les cartes de modèles, reprenant un ensemble structuré d'informations concernant un modèle [35].

D'autres architectures de modèles apprennent des concepts non supervisés dans les données. Les réseaux de neurones auto-explicatifs (Self-Explaining Neural Network, SENN) utilisent l'architecture encodeur-décodeur pour apprendre ces concepts [5]. Certains montrent l'activation neuronale associée à des éléments de concept, tels que les couleurs dans le cadre de l'analyse d'images [12]. Ces concepts peuvent être supervisés, c'est le cas des modèles de goulots d'étranglement conceptuels (Concept Bottleneck Models, CBM), qui apprennent à associer une entrée à un ensemble de concepts, présents dans les données d'entraînement, puis les ensembles de concepts à une classe [61]. Une fusion des concepts supervisés et non supervisés de [5, 61] intègre à la fois l'expertise métier et la découverte de nouveaux concepts [96]. Similairement aux concepts, des prototypes de granularité modulable peuvent être présentés, issus des motifs présents dans les données d'entraînement du modèle [107]. Le dictionnaire de prototypes associés peut être directement appris par le modèle, [108, 120]. Dans le cadre de l'analyse des séries temporelles, la méthode eXplainable Representation of Complex System Behavior (XR-CSB) extrait ces concepts correspondant à des états [57]. XR-CSB permet de générer des explications sous la forme

d'automates à états.

En résumé, les trois types d'approches possèdent leurs avantages et inconvénients, synthétisés dans le tableau 1.1.

TABLE 1.1 – Avantages et inconvénients des différentes approches en XAI.

Approche	**Avantages**	**Inconvénients**
Boîte noire	Pas de contrainte technique sur le modèle Précision du modèle non impactée	Explications non basées sur le fonctionnement interne du modèle Coût supplémentaire post-entraînement
Boîte grise	Explications basées sur le fonctionnement du modèle Précision du modèle non impactée	Contraintes techniques sur le choix du modèle Coût supplémentaire post-entraînement
Modèle transparent	Explications basées sur le fonctionnement du modèle Coût supplémentaire post entraînement limité	Contraintes techniques sur le modèle Risque de compromis entre explicabilité et précision

Les principales méthodes d'explicabilité ont été abordées, proposant un aperçu de la multiplicité des propositions de la communauté. Chaque méthode possède ses avantages et inconvénients. La section suivante couvre l'état de l'art de l'évaluation de ces méthodes.

1.3 Comment évaluer une explication ?

Évaluer les méthodes d'explicabilité n'est pas systématiquement réalisé dans la littérature. Toutefois de nombreuses méthodes sont présentées dans ce chapitre, associées aux points qu'elles permettent d'évaluer.

Les méthodes d'évaluation sont basées sur des mesures tirées des explications elles même, ou s'appuient sur le ressenti utilisateur ou l'impact d'une explication dans une tâche [7, 31]. Le choix d'une méthode adaptée rend compte non seulement de la qualité d'une explication mais aussi de son adéquation avec les objectifs de l'outil global dans la réalisation d'un acte métier.

Cette section présente les différentes méthodes d'évaluation des explications, à l'état de l'art. La section 1.3.1 définit les propriétés souhaitables pour une explication, recoupant et regroupant les noms parfois différents dans la littérature. Il vise à aider la sélection d'une ou plusieurs propriétés à maximiser, ce qui guidera le choix d'une méthode d'évaluation

par la suite. La section 1.3.2 définit les avantages et inconvénients de deux types de protocoles, ceux faisant appel à des utilisateurs et ceux qui s'en détachent. L'intégration d'humains dans un protocole expérimental apportant un travail conséquent, ce choix doit être réfléchi et motivé. Les sections 1.3.3 et 1.3.4 présentent les méthodes d'évaluation de la littérature, les liant aux propriétés qu'elles permettent d'évaluer. Enfin, la section 1.3.5 met en avant les challenges du domaine dans le lien entre évaluation technique et évaluation humaine.

1.3.1 Propriétés souhaitées

Les propriétés souhaitées des explications sont nombreuses. Cette section les regroupe en harmonisant les notations de l'état de l'art quand cela est possible. Dans un premier temps, les propriétés qui conviennent à toutes les explications seront citées avant d'être détaillées.

Les propriétés des explications sont souvent appelées avec des noms variables dans la littérature. Toutefois, il est possible de les regrouper sous des adjectifs suffisamment génériques.

Fidèle La fidélité correspond à la capacité d'une explication à être en accord avec le phénomène ou modèle expliqué [21, 44, 48, 62, 69, 71, 91]. Pour [22] cela signifie qu'elle est argumentée ; elle s'appuie sur des éléments vérifiables par le receveur. Pour une tâche de classification d'images de chien, cela signifierai souligner des éléments que le receveur sait être différenciant : forme des oreilles, du museau, etc. Pour [29, 82] elle comporte donc les éléments minimaux permettant la prise de décision.

Interprétable L'interprétabilité d'une explication est une notion plus complexe. Pour [44, 62, 111] cela correspond à la taille d'un modèle élève générant les explications, donnant ainsi une explication de taille raisonnable. [90] abonde dans ce sens en indiquant qu'une explication interprétable possède un nombre limité d'éléments et s'appuie sur des notions adaptées à son receveur. Pour [39] cela signifie que l'explication est suffisamment simple pour être comprise. Il est ainsi possible de découper l'interprétabilité en trois propriétés plus spécifiques : la concision, la complétude et l'adaptation.

Concise Une explication concise comporte un nombre limité d'éléments. Le receveur n'est pas noyé par la quantité d'informations. Si de nombreux travaux y font référence [62,

82, 85, 90, 121], [73] va plus loin, mettant en avant le fait que les receveurs ont une forte tendance à favoriser les explications les plus courtes. [39] met en garde contre le biais humain qui, en favorisant les explications courtes, risque de préférer des systèmes d'explications persuasifs plutôt qu'interprétables. Ces systèmes s'appuient alors sur des connaissances et préférences des receveurs, quitte à être moins fidèles [46].

Auto-explicative Une explication est auto-explicative lorsqu'elle ne nécessite pas de connaissances supplémentaires pour être comprise [22, 39]. Par exemple, une équation sera facilement interprétable pour un expert de la donnée, mais pas pour un receveur n'ayant pas l'habitude de manipuler ce format d'information [90]. Elle comporte donc tous les éléments nécessaires à sa compréhension sans qu'il soit nécessaire de les détailler [48].

Adaptée aux receveurs Une explication adaptée aux receveurs est conçue dans un vocabulaire connu de ces derniers, par exemple en intégrant le vocabulaire spécifique à la tâche ou domaine métier associé [22]. [25, 29] abondent en ce sens, estimant qu'elle pourrait ressembler aux explications que fourniraient des humains qui auraient la même tâche. Dans cette même dynamique mais spécifique au TALN, les travaux de [121] montrent que les utilisateurs ont une préférence pour les explications de textes basées sur des phrases plutôt que sur des mots.

Représentative Une propriété associée est la non-ambigüité, définie par [62] pour un ensemble de règles, comme le respect à la fois d'une bonne couverture des règles, et d'un faible empiètement des règles les unes entre les autres [62].

Utile Enfin, l'utilité des explications met en avant son intérêt pour les receveurs selon leurs attentes [48, 73]. Si le receveur est un scientifique des données, manager ou régulateur, cela correspond à détecter des biais [39]. Pour un expert du domaine ou une personne concernée par l'usage du modèle, l'utilité peut être une meilleure représentation mentale du comportement du modèle, le modèle mental [6, 25, 76, 91], notamment en détectant mieux ses erreurs [33].

Pour un utilisateur cela peut l'amener à améliorer sa performance dans l'acte métier associé, ou aller plus vite [30, 39, 52, 76, 91]. Pour tout utilisateur, l'utilité crée ou améliore sa confiance dans le modèle [48, 76]. Dans cette même optique, [89] propose de favoriser la mise en avant de critères actionnables, c'est à dire ceux sur lesquels l'utilisateur peut agir ou conseiller une action.

Il est difficile de maximiser chaque propriété, notamment parce que certaines sont incompatibles. Ainsi l'auto-explicabilité empêche la concision. Pour chaque projet, un ensemble de propriétés sur lesquelles doit se porter l'effort peut être défini. Ces propriétés étant définies, les sections suivantes définissent les modalités d'évaluation du respect de celles-ci. Dans la section 1.3.2 est discutée la présence ou non des utilisateurs humains lors de l'évaluation des explications.

1.3.2 Évaluation avec ou sans utilisateur

Les modalités d'évaluation des explications sont dépendantes à la fois des contraintes du projet : temps, argent, disponibilité des différents acteurs, mais aussi des propriétés souhaitées, vues en section précédente. Nous présentons les trois niveaux d'évaluation de la littérature : avec des utilisateurs réels, avec des systèmes simulant le comportement des utilisateurs, ou sans utilisateurs, ni réels ni simulés.

Avec des utilisateurs réels L'évaluation avec utilisateur permet de valider la bonne adéquation entre l'explication et son receveur. Elle assure notamment que les propriétés d'interprétabilité et d'utilité sont valides. [33] divise explicitement les évaluations avec utilisateurs en deux types, observés dans la littérature. Ces évaluations sont :

1. l'évaluation applicative,
2. l'évaluation humaine

Pour l'évaluation applicative, les conditions d'utilisation du modèle doivent être les plus proches possible de la réalité. L'évaluation porte alors sur la compréhension et la préférence de l'utilisateur recevant l'explication. Le profil de cet utilisateur doit correspondre à celui ou ceux auxquels sont destinés les explications. Il est ainsi possible de demander à effectuer l'action métier, avec et sans explications, ou avec des explications différentes, et comparer les performances et la vélocité de l'utilisateur [25]. Une évaluation similaire consiste à demander aux utilisateurs de donner une explication au même format, et comparer les deux. Toutefois elle nécessite de travailler sur des modèles simples, elle est donc moins adaptée à l'apprentissage profond [25, 71].

L'évaluation humaine se fait sur des tâches simplifiées, et non l'acte métier cible. De même elle ne nécessite pas de faire appel spécifiquement au public cible des explications, ce qui est intéressant lorsque ce public est composé d'experts peu disponibles. Une tâche simplifiée peut être :

— la comparaison de paires d'explications [25, 62],
— la comparaison de modèles simples tels que des petits arbres de décision[4],
— l'évaluation qualitative d'une explication pour un cas fonctionnel connu [57],
— retrouver le même résultat que le modèle, à l'aide de l'entrée et l'explication associée [25, 65, 91, 121],
— sélectionner le "meilleur modèle" selon les résultats et explications associées [90].

Ces évaluations se font avec un ordre de grandeur de une à plusieurs dizaines d'humains participant : 33 pour [62], 15 pour [121]. Quelques expérimentations sont menées à plus grandes échelles comme celle de [4] dont qui fait appel à 100 personnes, ou celle de [65] qui rassemble 158 personnes.

Présenter les explications à un receveur permet de s'assurer de la bonne adéquation entre ce dernier et le format d'explication proposé. Ce type d'évaluation est le plus proche des conditions réelles d'usage des explications, et est donc le plus précis. C'est la seule modalité d'évaluation qui permet de vérifier que le vocabulaire est adapté. Cette précision a un coût non négligeable [25, 33]. Ce type d'évaluation nécessite également de prendre du temps, notamment pour récolter les évaluations des receveurs. Dans le cas où des professionnels sont requis, iels sont parfois très peu disponibles : experts, médecins etc. Ces évaluations sont également sensibles aux biais humains, avec une préférence pour les explications courtes, qui ressemblent à celles que fournirait un humain [39, 72, 82].

Avec des utilisateurs simulés Il est également possible d'évaluer une explication pour son niveau d'informations et son adéquation avec l'acte métier associé. C'est alors la performance au regard de la tâche à accomplir qui est mesurée. Pour mesurer cette performance, quand les utilisateurs sont peu disponibles, il est possible de les remplacer par des algorithmes simulant leur comportement. Les utilisateurs simulés sont des modèles plus ou moins complexes, tels que des forêts aléatoires ou des réseaux de neurones [91]. Ces utilisateurs simulés peuvent effectuer, à l'instar des utilisateurs réels, des tâches complexes ou simplifiées. Il est ainsi possible d'entraîner un classifieur qui va prédire une classe en s'aidant de l'explication pour la retrouver [30, 90, 91], ou encore choisir le meilleur modèle, grâce aux explications [90].

Ce type d'expérimentation est intéressant dans la mesure où son coût est moindre comparé aux expérimentations avec utilisateurs réels. En particulier, si les utilisateurs ciblés sont des experts, les utilisateurs simulés ont l'avantage de ne pas être limité en nombre ou en quantité de données d'expérimentation traitées. Ils permettent de rester proche

de l'acte métier, en simulant cet acte par exemple. Enfin, ils permettent une bonne reproductibilité, facilitant les comparaisons entre méthodes, contrairement aux expériences avec des utilisateurs réels. Toutefois, il n'est pas possible de valider le vocabulaire ou l'acceptation d'un système comme c'est le cas avec des utilisateurs réels.

Sans utilisateurs Enfin, il est possible de valider les propriétés des explications sur de larges jeux de données, sans recourir aux utilisateurs. Dès lors que la propriété est mesurable (définie par une équation), elle peut alors être directement évaluée et comparée.

Les évaluations sans utilisateurs se font par la mesure d'une propriété. La concision pourra être associée au nombre de variables remontées dans une explication basée sur ces dernières [121]. La fidélité pourra correspondre au taux d'erreur d'un système d'explication en regard du phénomène ou modèle qu'il doit expliquer [62]. Dans [82], les éléments constituant l'explication sont supprimés un à un de l'entrée du modèle et l'observation du changement de résultat indique la pertinence de l'explication : plus celle-ci se fait tôt, plus les éléments constituant l'explication étaient nécessaires pour mener au résultat.

Les expérimentations sans utilisateur ont un coût moindre, et une bonne reproductibilité. Ils sont encore plus adaptés à l'évaluation comparative que les utilisateurs simulés, car ils ne nécessitent pas l'entraînement de ces derniers. Toutefois, ils sont dé-corrélés de l'acte métier, et doivent servir à la comparaison de propriétés formalisées mathématiquement.

Les différentes modalités d'expérimentations influent sur les types d'évaluation. La section 1.3.3 détaille les possibles évaluations objectives, tandis que la section 1.3.4 détaille les possibles évaluations subjectives.

1.3.3 Évaluation objective

Les évaluations objectives permettent de mesurer impartialement les performances du système explicatif, ou des explications en elles-mêmes. Il se fait principalement sans utilisateurs, par la mesure de propriétés telle qu'abordée en section 1.3.2. Toutefois, les mesures objectives peuvent également être extraites des expériences utilisateurs.

Chaque mesure met en avant un aspect des explications, et peut permettre de valider le respect d'une propriété. Toutefois, il n'existe pas aujourd'hui une mesure universelle qui permettrait d'évaluer et comparer tous les systèmes explicatifs existants [33]. Cette section présente les mesures objectives de la littérature, permettant d'évaluer les systèmes d'explications. Les mesures sont associées aux propriétés auxquelles elles sont associées. Les évaluations présentées sont respectivement centrées sur l'explication et l'acte métier.

Centrée sur l'explication Lorsque le système d'explication est un modèle, les propriétés de ce modèle forment une première source de mesures objectives. C'est le cas avec LIME [90] Pour mesurer la concision du modèle, il est possible d'utiliser sa taille en tant que nombre de paramètres [44]. Toutefois cette mesure ne rend pas compte de ce qu'a appris le modèle. On peut alors y associer des mesures portant sur la fidélité du modèle, caractérisée par les mesures communes en IA : exactitude ($exactitude = \frac{\#\text{bonnes réponses}}{\#\text{réponses}}$) précision etc. [48, 62, 69, 71, 91].

En se concentrant non pas sur le système d'explication mais sur les explications en elles-mêmes, il est possible de vérifier leur conformité avec des propriétés désirées. Pour une propriété, il existe parfois de nombreuses mesures dans la littérature. Chaque propriété évaluée par des mesures objectives est reprise, cette fois en détaillant les principales manières de les mesurer.

Fidèle La mesure de fidélité des explications est rapidement mise en place en altérant les données d'entrée à partir des explications [78, 93, 94, 97]. Pour [82] la présence d'éléments minimaux permettant la prise de décision est caractérisée par une faible Deletion Area Under Curve (DAUC). La DAUC est l'aire sous la courbe de score de classification, pour la classe cible, lors de la suppression d'éléments de l'explication, du plus important au moins important, selon un score d'importance attribué par la méthode d'explication évaluée. Cette évaluation est particulièrement adaptée aux cartes de saillances sur les images. Une instance dénuée des éléments expliquant le plus une classe sera dans cette logique, très rapidement détectée comme n'appartenant pas à cette classe. Similairement, [82] présente l'Insertion Area Under Curve, (IAUC). L'IAUC est caractérisée par l'aire sous la courbe de score de classification, pour la classe cible, lors de l'insertion d'éléments de l'explication. [41] propose une amélioration de la DAUC et l'IAUC qui favorise cette propriété : la corrélation de suppression (Deletion Correlation, DC) et la corrélation d'insertion (Insertion Correlation, IC), qui prennent en compte les pondérations des éléments de l'explication. Cette modification permet de pénaliser les explications mal calibrées par rapport au modèle expliqué. Les explications courtes sont pénalisées si le modèle s'appuie sur un nombre élevé d'éléments, et favorisées si le modèle s'appuie sur un nombre limité d'éléments. Pour [2], la fidélité est caractérisée en mesurant l'impact des changements de paramètres du modèle sur l'explication reçue.

Concise La concision est mesurée par la taille d'une explication [62, 85, 90, 121], une explication concise comporte un nombre limité d'éléments.

Adaptée aux receveurs Pour vérifier, comme le préconisent [25, 29] qu'une explication ressemble aux explications que fourniraient des humains qui auraient la même tâche, il est possible d'utiliser des mesures telles que l' intersection sur l'union (Intersection over Union, IOU) [29]. Le calcul de l'IOU donne un score de similarité de 0 à 1, et se calcule, pour deux ensembles A et B : $IOU_{AB} = \frac{A \cap B}{A \cup B}$. Une IOU de 1 signifie que deux ensembles sont identiques. Cette métrique est utilisée dans [14] pour évaluer les explications sur les images. Si seules quelques explications sont possibles, le cas peut être considéré comme un problème de classification, et les mesures de précision habituelles peuvent être utilisées [22].

Pour mesurer la clarté et lisibilité des explications sous forme de textes, le score *Flesch-Reading-Ease* peut être appliqué [24]. Il est calculé via la formule suivante :

$$FRE = 206,835 - 84,6 * M - 1,015 * P \tag{1.1}$$

où le terme $M = \frac{\#\text{nombre de syllabes}}{\#\text{mot}}$ correspond à la longueur moyenne d'un mot, en nombre de syllabes, et $P = \frac{\#\text{mots}}{\#\text{phrases}}$ et la longueur moyenne d'une phrase en nombre de mots. Les coefficients sont déterminés de façon totalement empirique, de sorte qu'un texte facilement lisible obtienne un score proche de 100, et un texte expert soit associé à un score proche de 0. Toutefois ce score est spécifique à l'anglais et il n'existe pas d'équivalent français faisant figure de référence.

Représentative La représentativité ou couverture d'une explication A est définie dans [91] par l'équation

$$\text{cov}(A) = \mathbb{E}_{D(z)}[A_{(z)}] \tag{1.2}$$

. La représentativité de A est donc la probabilité $\mathbb{E}$ qu'elle s'applique aux éléments z d'un ensemble d'instances D. En d'autres termes, A est représentative si elle est valide pour un grand nombre d'éléments de D où elle est appliquée.

Dans le cadre de la représentativité d'une règle, [62] considère la couverture comme le nombre d'éléments concernés par une règle.

Centrée sur l'usage Au-delà de l'explication en elle-même, l'évaluation peut porter sur l'intégration de l'explication et son intérêt pour l'acte métier ou l'usage final du modèle expliqué. Puisque les mesures présentées sont objectives, il n'est pas question ici de préférence utilisateur, mais de critères de performance ou de réussite pour le cas d'usage. Ainsi, cette section se rapporte à la propriété d'utilité vue en section 1.3.1.

Pour un expert du domaine ou une personne concernée par l'usage du modèle, l'utilité peut être une meilleure représentation mentale du comportement du modèle, aussi appelée modèle mental [6, 25, 76]. Cette représentation peut être mesurée au travers de la précision humaine, soit la capacité d'un humain à anticiper le comportement d'un modèle sur des instances inconnues. La précision humaine est alors définie comme étant la proportion d'instances pour lesquelles les utilisateurs arrivent à anticiper le résultat du modèle [33]. Cette mesure de précision peut aussi être effectuée avec des utilisateurs simulés, rendant l'expérience reproductible [91]. Pour les utilisateurs réels, la bonne représentation du modèle mental passe aussi par sa couverture, correspondant à la part de prédictions pour lesquelles l'utilisateur choisit de proposer une réponse autre que "je ne sais pas" lorsqu'on lui demande d'anticiper le résultat du modèle [91].

Dans [50], les utilisateurs doivent prédire la suite d'une vidéo selon quatre possibilités. Un groupe reçoit l'entrée du modèle sous forme d'images, et un second groupe reçoit une explication sous forme de carte de saillance de ces mêmes images. Les participants doivent alors prédire la sortie du modèle. Considérant les réponses des utilisateurs comme des résultats de classificateurs binaires, les auteurs calculent une courbe ROC (*Receiver Operating Characteristic*, ou caractéristique de fonctionnement du receveur), son aire sous la courbe donnant la mesure du succès de leurs cartes de saillance.

Les auteurs de [25] proposent une méthode d'évaluation dans laquelle les humains doivent copier la décision d'un modèle en ayant une instance et son explication. Il est aussi possible de présenter aux utilisateurs, une instance, la décision y et l'explication associées, et leur demander comment iels changeraient le modèle pour avoir une décision donnée $\bar{y} \neq y$. Cette tâche nécessite à l'utilisateur de bien comprendre le modèle.

Pour un utilisateur, l'explication peut l'amener à améliorer sa performance dans l'acte métier associé. Dans les travaux de [52], la tâche des utilisateurs est d'accepter ou refuser des dossiers de demande d'asile et permis de séjour pour les Pays Bas. Elle est présentée sous forme de texte décrivant la situation, reprenant des données tabulaires. Deux types d'explications sont proposés ici : un basé sur des règles (*"Valid reason for fleeing the country"*, "la raison pour fuir le pays est valide" [52]), l'autre donnant un pourcentage

de confiance du modèle pour un cas similaire. Le nombre moyen de dossiers correctement traités augmente si les participants, novices ou experts, reçoivent une recommandation accompagnée d'une explication avec le dossier.

L'utilité peut également rendre l'utilisateur plus rapide pour effectuer sa tâche. Il est alors possible de mesurer sa vélocité avec et sans explication, et mesurer le gain de temps [91, 121]. [91] propose cette évaluation sur la classification de données tabulaires, avec des explications basées sur les variables.

Enfin, [89] propose la mise en avant de critères actionnables dans les explications. Pour des données tabulaires, il s'agit de variables dont les valeurs peuvent être changées, correspondant à des éléments logiquement actionnables dans la réalité. Cette contrainte est directement intégrée dans sous la forme de contraintes et de coûts. Dans leur exemple, l'âge d'une personne est contraint à ne pas pouvoir diminuer.

Les différentes évaluations objectives permettant d'évaluer les explications, selon les propriétés souhaitées, ont été passées en revue. Ces mesures permettent la comparaison de système d'explications sur des éléments techniques et sans tenir compte du ressenti humain. La section suivante présente les mesures subjectives, axées cette fois sur la préférence des utilisateurs.

1.3.4 Évaluations subjectives

Lorsque l'objectif de l'explicabilité est de favoriser l'acceptation d'un modèle par un humain, les évaluations subjectives permettent de prendre en compte la préférence des personnes recevant l'explication. Ces évaluations se font donc nécessairement avec des humains, afin de rendre compte de leur jugement. De par leur plus grande complexité de mise en place, les évaluations subjectives sont moins présentes dans la littérature, comparées à l'évaluation objective [62]. Ce type d'évaluation prend en compte le lien entre l'explication et les personnes les recevant, et notamment le ressenti de ces dernières. Ces évaluations sont coûteuses et requièrent du temps [25, 33]. Elles sont également sensibles aux biais humains [82]. De même, contrairement aux mesures objectives, les évaluations subjectives ne sont pas reproductibles et conviennent moins pour la réalisation d'études comparatives. Les différents types d'études comparatives sont détaillés dans cette section, et associés aux propriétés souhaitées des explications. La section se divise en deux parties, une première portant sur les études des tâches réelles, et la seconde sur les tâches annexes.

Sur une tâche réelle Il est recommandé que la tâche effectuée lors de l'évaluation représente au mieux l'environnement fonctionnel dans lequel est utilisé le système créé, avec le public ciblé [25]. Ceci permet de prendre en compte le contexte inhérent aux explications et les spécificités du public ciblé. Ainsi, les auteurs de [25] proposent de mesurer la satisfaction ressentie lorsqu'une personne utilise un outil d'IA avec les explications associées pour effectuer un acte métier.

Les auteurs de [91] réalisent une mesure subjective de préférence avec un questionnaire demandant à des utilisateurs quelle explication iels préfèrent entre deux systèmes explicatifs : LIME et les Ancres. De même, les auteurs de [62] effectuent la comparaison de leur solution avec LIME.

Sur une tâche annexe Lorsque le public cible n'est pas disponible, ou que l'acte métier du système d'IA n'est pas utilisable ou simulable pour les évaluations, il est possible de passer par l'évaluation sur une tâche annexe.

Une tâche annexe simple consiste à faire deviner à un humain le résultat d'un modèle d'IA. La nuance ici est qu'il n'est pas souhaité d'effectuer la tâche réelle du modèle, mais de reproduire ses bonnes comme ses mauvaises réponses. Dans [62], les humains reçoivent une explication d'un système d'explication parmi trois. Cinq questions sont posées pour évaluer leur compréhension du modèle dans différents sous espaces. La précision des participants est mesurée et permet la comparaison des différents outils de génération de règles.

Une autre évaluation consiste à demander à un humain de choisir entre deux classifieurs, l'un étant significativement meilleur que l'autre, compte tenu uniquement de leurs explications [90]. L'expérimentation de [4] fonctionne sur le même principe en effectuant une comparaison par paires de modèles d'apprentissage automatique réputés compréhensibles : moteurs de règle, arbres de décision etc. Les participants ont les bases nécessaires pour comprendre les éléments présentés. La question posée est de dire si un des deux modèles présentés est plus compréhensible que l'autre. Le retour des participants se fait via une échelle de Likert, où l'échelon 1 correspond à une perception égale des modèles, et l'échelon 9 un des deux modèles est bien plus compréhensible que l'autre. La tâche de l'utilisateur n'est plus l'acte métier mais la sélection du meilleur outil pour accomplir cette dernière. Ces évaluations sont appropriées lorsque l'objectif est d'améliorer l'acceptation d'un modèle.

Les évaluations subjectives, basées sur la préférence humaine, ont été décrites aussi bien sur des tâches réelles que sur des tâches annexes. Les systèmes explicatifs ainsi que

la manière de les évaluer ont pu faire l'objet de débats et critiques dans le domaine de l'XAI. La section suivante en présente un ensemble non exhaustif.

1.3.5 Évaluations techniques et humaines

Le domaine de l'XAI est récent, et la question de l'évaluation et de la qualité de ces explications l'est plus encore. Certaines méthodes, définitions et modalités d'évaluations sont ainsi critiquées, au fil des avancées du domaine. Cette question présente les questionnements soulevés par le rapprochement entre considérations techniques et humaines.

L'interprétabilité d'une explication ou d'un modèle est souvent définie comme sa taille [62, 82, 85, 90, 111, 121]. Dans la section 1.3.1, il est précisé que les humains ont un biais et favorisent les explications les plus courtes [73], ce qui peut mener à la sélection de systèmes d'explication persuasifs [39]. Toutefois, [4] mène une expérimentation utilisateur sur la compréhension des modèles, sur deux jeux de données. Pour l'un, aucune corrélation entre taille du modèle et compréhension humaine n'est détectée. Pour l'autre jeu de donnée, c'est même l'inverse. Les humains ont trouvé plus compréhensible les modèles de plus importantes, à contre-courant des définitions de la littérature : *"Participants seemed to think that the larger and more complex models were more understandable (at least for the Labor data set)"* [4].

Les modèles à attention sont des modèles transparents utilisés pour générer des explications [11, 37, 42, 66, 95, 114, 119]. Toutefois, les auteurs de [51] alertent sur deux points qui font que, selon eux, le mécanisme d'attention ne peut être considéré comme fournissant des explications : "Attention is not explanation" [51]. Le premier point est l'accord avec des mesures alternatives d'importance de variables, telles que les explications basées sur le gradient. Leur expérimentation sur de multiples jeux de données montre que les variables mises en avant par le mécanisme d'attention ne sont pas corrélée avec celles mises en avant avec les méthodes basées sur le gradient. Le second point est la multiplicité des explications : pour une même instance en entrée, différentes distributions d'attention sur les variables, ou vecteurs d'attention, peuvent mener à une même sortie du modèle. Les auteurs génèrent des explications adversaires, soit des vecteurs d'attention les plus éloignés possibles de celui associé à un couple entrée sortie, et qui ne change pas la sortie du modèle pour l'entrée donnée.

En réponse à ces travaux, les auteurs de [116] reprennent le second point de [51] à savoir la multiplicité des explications. Ils montrent l'interdépendance entre la couche d'attention et le modèle au sein duquel elle est entraînée. Ils estiment également que la multiplicité

des explications n'est pas le signe que l'explication proposée est fausse. Leur réponse est intitulée "Attention is not not explanation" [116]. Toutefois malgré un tire évoquant un désaccord, les auteurs de [116] rejoignent les auteurs de [51] en ceci que le mécanisme d'attention échoue, dans certaines circonstances, à indiquer fidèlement le lien entre variables d'entrée et résultat en sortie d'un modèle, du fait de la multiplicité des explications et l'existence d'explications d'attention adversaires. Les explications basées sur l'attention sont donc interprétables pour l'humain, mais pas toujours fidèles au fonctionnement du modèle expliqué.

Les auteurs de [122] pointent également du doigt le manque de robustesse des explications sur des méthodes boite grise basées sur le gradient ou boite noires. Ils montrent que deux modèles aux performances similaires peuvent se baser sur des explications très différentes. Ils avancent que l'information permettant de relier une entrée à une sortie peut être redondante dans plusieurs variables d'entrées. Ces travaux font écho au point de vue des sciences sociales. Selon [72], il n'y a pas une unique explication pour un phénomène, qui peut avoir de multiples causes. Si ce fait est accepté d'un point de vue social, il questionne sur la fidélité des explications d'un point de vue technique. La variabilité des explications locales est considérée par les auteurs de [38, 67] comme un frein majeur. Ils proposent de construire la confiance des algorithmes sur de bons résultats plutôt que présenter des explications locales aux utilisateurs.

Les méthodes d'évaluation communes de fidélité de cartes de saillance, par insertion de variables ou suppression de variables, sont critiquées [41]. Les auteurs montrent que ces mesures passent par la création d'instances hors domaine d'entraînement, ce qui pose question sur leur qualité et pertinence. Les travaux des mêmes auteurs [40] vont plus loin et montrent que les mesures basées sur l'insertion de variables favorisent les méthodes d'explicabilité indépendantes des modèles, tandis que les mesures basées sur le masquage progressif des variables favorisent les méthodes basées sur l'attention.

Ces débats au sein de la communauté scientifique montrent bien la complexité inhérente à travailler sur un domaine récent, à mi-chemin entre la performance technique et le lien avec l'humain.

1.4 Conclusion

Ce chapitre présente l'état de l'art pour générer et évaluer des explications associées au fonctionnement des modèles d'apprentissage automatique et profond.

Dans un premier temps, nous avons traité la problématique de la génération d'explications. Les nombreuses méthodes ont été regroupées par stratégie, à savoir traiter les modèles comme des boites noires, s'appuyer sur leur fonctionnement, voire les concevoir en tant que systèmes transparents.

Nous avons ensuite abordé la complexité de l'évaluation des explications ; mettant en avant les propriétés des explications avec les divers choix stratégiques de conception d'une méthode d'évaluation. La présence d'utilisateurs ou non, et le choix d'évaluations subjectives ou objectives ont notamment été mis en avant. Enfin, les débats de la communauté ont été mis en avant, montrant la dynamique du domaine sur des questionnements à mi-chemin entre la technique et les sciences sociales.

Dans un domaine en plein essor, les propositions sont nombreuses et variées, abordant les multiples problématiques rencontrées derrière la vaste notion d'explicabilité. Cet état de l'art montre l'intérêt de prendre en compte les objectifs et les contraintes de la génération et l'évaluation d'explications des modèles profonds.

Le prochain chapitre présente le cas d'usage basé sur des données réelles, avec sa problématique fonctionnelle, les données associées et le modèle profond employé. Il servira de base pour illustrer la suite des travaux.

Résumé

- ✓ Les méthodes de génération d'explications sont nombreuses et se chargent de problématiques spécifiques
- ✓ La génération d'explication est contrainte par les modèles et les besoins des utilisateurs
- ✓ L'évaluation est conditionnée par les contraintes en budget et en disponibilité des personnes participantes
- ✓ Les évaluations vérifient l'adéquation avec certaines propriétés
- ✓ La rencontre entre technique et science sociale induit des débats dans la communauté
- ✓ L'état de l'art des méthodes de génération d'explications a donné lieu à une première publication [56]

Chapitre 2

Cas applicatif et prérequis

Dans ce chapitre

Nous présentons les travaux prérequis pour nos expérimentations présentées dans les chapitres 3 et 4. Nous développons le contexte applicatif servant d'illustration au travers d'un cas d'usage. Un point sera fait sur la création des jeux de données. Puis nous verrons la génération d'explications locales. Enfin les interfaces de visualisation seront énumérées.

Dans ce chapitre, nous définissons le contexte applicatif sur lequel nous menons nos expérimentations. Les prérequis tels que la création et collecte des données sont passés en revue. Nous générons des explications locales pour les experts du domaine, et proposons différentes visualisations d'explications.

Ce chapitre présente ainsi tous les prérequis nécessaires à la réalisation des expérimentations présentées dans les chapitres 3 et 4. Les matériaux présentés font appel à la labellisation manuelle de données, à la génération d'explications vues dans la littérature, mais aussi à la création d'un démonstrateur et la préparation des visualisations à y intégrer. Nous discuterons des problématiques rencontrées, telles que la définition d'une vérité terrain, et les avantages et inconvénients des méthodes de génération d'explications utilisées.

La première section est consacrée à la présentation du cas d'usage et des données associées, dont une partie est labellisée manuellement. La seconde section présente la génération des explications. Enfin, en section 2.3, nous détaillons l'implémentation des différentes visualisations dans un démonstrateur.

2.1 Présentation du contexte LEGO

Nous abordons l'explicabilité des modèles profonds dans le cadre spécifique du traitement de la langue naturelle. Afin de définir et appréhender ce contexte, nous illustrons nos travaux avec un cas d'usage de traitement de textes.

Un cas d'usage de Pôle emploi est le tri automatique des offres non conformes. Ce cas d'usage est appelé *LEGO*, pour LEGalité des Offres. Pôle emploi publie mensuellement 250 000 offres internes et maintient un stock d'environ 600 000 offres partenaires, agrégées de sites tels que Monster ou Jobijoba. Ces offres passent par un processus de vérification avant d'être publiées ; afin de répondre aux critères de qualité. À ce jour, il existe plus de 70 motifs de non-conformité. Parmi eux, le dépassement d'un nombre maximal de caractères et les champs manquants sont vérifiés automatiquement avec succès. Reste la vérification des points plus difficiles à détecter tels que les discriminations sur les candidats (âge) ou encore des points contractuels ("CDD pouvant évoluer vers un CDI").

Ce processus de vérification a longtemps été effectué par un système à base d'expressions régulières, formant un ensemble complexe de règles aux effets de bord inattendus. Ce système historique a été un premier pas vers l'automatisation. Toutefois, il a produit des erreurs que nous retrouvons dans les données les plus anciennes.

La DSI de Pôle emploi a automatisé le tri des offres non conformes, via un système composé de plusieurs modèles d'analyse sémantique. Ce système actuel lève des alertes non contraignantes pour les agents de Pôle emploi, et des refus contraignants pour les recruteurs et recruteuses. Donner au personnel les raisons des alertes de leurs offres permet l'harmonisation des bonnes pratiques. Pour les recruteurs, l'intérêt est double : leur permettre de corriger leur offre et donc de publier sur le site, et réduire leur frustration liée au rejet.

La figure 2.1 présente l'interface de travail des conseillers et conseillères à dominante entreprise (CDE) pour la création d'une offre d'emploi. L'alerte est présente sur un bandeau jaune en haut de page, en surlignant le champ incriminé "Descriptif de poste" et en affichant la phrase problématique et l'alerte associée : "Droit du travail : Type de poste". L'alerte est remontée grâce à l'outil de tri automatique des offres non conformes.

Le système actuel se concentre sur l'analyse sémantique des phrases des offres, afin de détecter un sous ensemble de motifs de non-conformité. La détection se fait au niveau des phrases d'offres d'emploi, sur le respect ou non des critères de conformité, avec 27 motifs possibles, ainsi que le motif "vide" correspondant aux offres légales. Nous avons

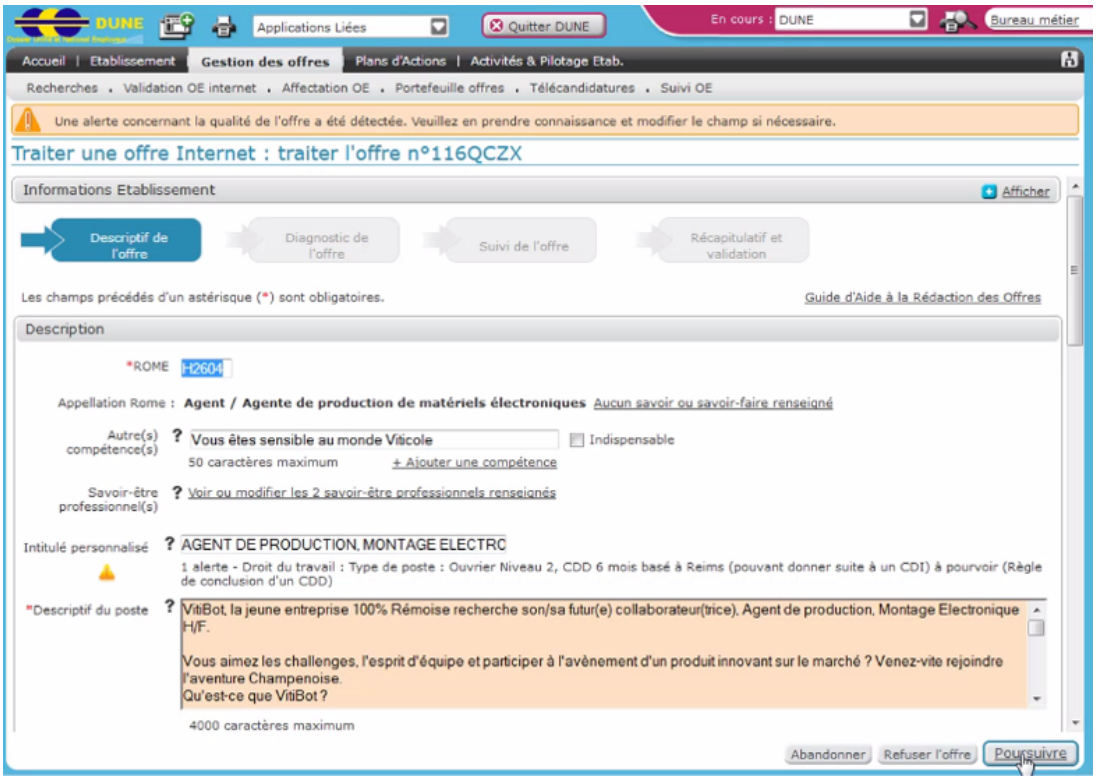


FIGURE 2.1 – Interface de l'outil de création d'offre avec une alerte liée au contrat de travail. L'alerte est accompagnée de la phrase incriminée. Le champ concerné, "Descriptif de poste", est surligné en jaune pour attirer l'attention de l'utilisateur. Iel peut ignorer l'alerte et cliquer sur le bouton "poursuivre".

ainsi un problème de classification multi-classes. Le cas multi-label, soit le cas où une même phrase possède plusieurs motifs de rejets, correspond à moins de 1% des cas. Pour simplifier, nous traiterons le cas d'usage comme un cas mono-label.

2.1.1 Jeux de données

Nos travaux concernent l'usage de modèles profonds tels que les réseaux de neurones. La phase d'apprentissage de ces modèles nécessite une quantité conséquente de données. Nous avons à notre disposition un large ensemble d'offres d'emploi, classées par le système historique. Le jeu de données d'entraînement contient 480000 phrases extraites d'offres réelles. Une majeure partie de ces phrases sont légales : 317843.

La distribution des motifs de rejet présentée en figure 2.2 montre une répartition inégale des classes, correspondant à la répartition des données en production. Certains motifs sont extrêmement rares, tels que les rejets liés à l'ethnie : 2 exemples sont disponibles.

Le jeu de données de test est constitué de données vérifiées manuellement, et les erreurs du système historiques y sont corrigées. De même, pour les phrases non conformes, les explications associées au motif de rejet sont indiquées sous forme d'ensemble de mots, à l'instar d'un surlignage. Ce travail manuel rend la création de ce jeu de données coûteux, il est donc restreint et composé de 208 phrases uniquement.

Deux sous-échantillons de cet ensemble de test de 208 phrases sont utilisés. Le jeu de test des bonnes prédictions (BP) est composé de 147 phrases du jeu de test, correctement prédites. Ce jeu de données permet de mesurer la performance des explications lorsque le modèle ne fait aucune erreur. Cela évite que la performance des explications soit affectée dans le cas d'un modèle peu performant. L'ensemble de test de phrases avec différentes explications (DE) est composé de 106 phrases de l'ensemble de test, avec des explications non-identiques entre elles. Ce jeu de données se concentre sur la tâche des utilisateurs qui comparent les explications et évite la situation où les utilisateurs doivent choisir entre des explications uniquement identiques. Ces deux jeux de données se recoupent tel qu'illustré en figure 2.3.

Les distributions des jeux de données sont présentées en annexe A.2. La création de ce jeu de test a soulevé des questionnements, tels que la difficulté de déterminer une explication idéale.

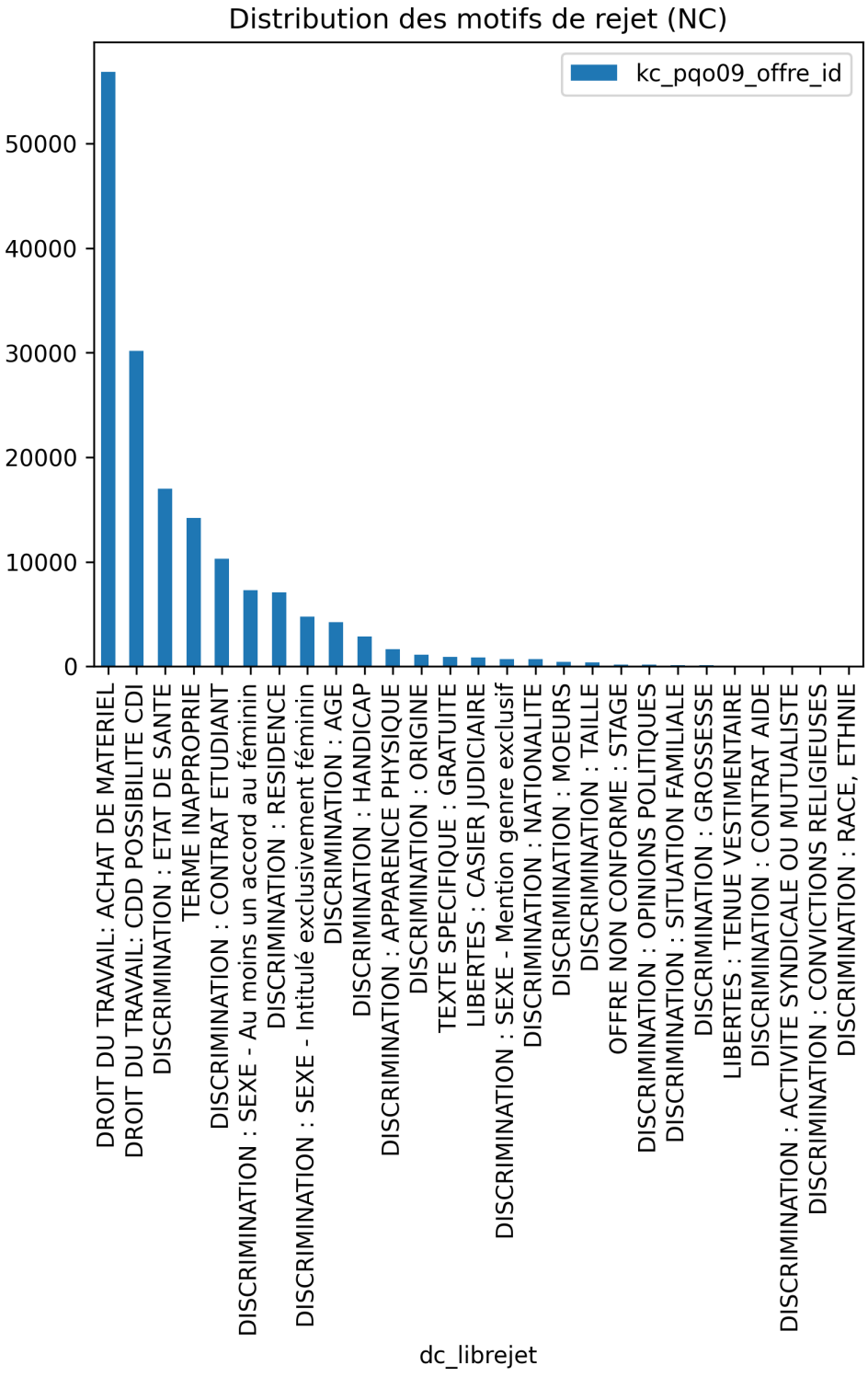


FIGURE 2.2 – Répartition des motifs de rejet dans les données d'apprentissage. Seules les phrases en alerte (non conformes, NC) sont prises en compte.

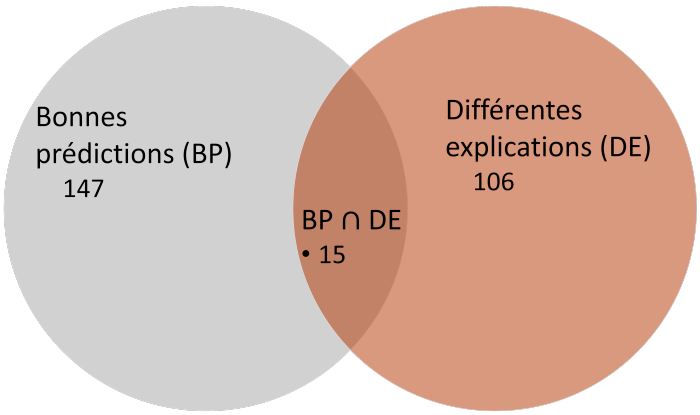


FIGURE 2.3 – Ensemble de test de 208 phrases labellisées manuellement. Deux sous ensemble de données en sont extraits : le jeu de test de bonnes prédictions (BP) et composés de différentes explications (ED)

2.1.2 Comment collecter les explications de référence

Générer une explication de référence ou explication attendue peut être fait en s'appuyant sur l'attention humaine [75]. Il est alors nécessaire de faire appel à des experts du domaine ou, à défaut, à une documentation de référence. On peut souhaiter obtenir une explication qui réponde à la question "qu'est-ce qui a été utilisé par le modèle pour donner sa réponse ?". Pour obtenir cette explication attendue, une première technique consiste à la demander aux utilisateurs, experts du domaine ou experts du modèle. Une seconde technique consiste à demander aux experts du domaine de vérifier les résultats des modèles et de voir où se concentre leur attention par oculométrie ou en rendant floues des parties d'une image [26]. Ces méthodes sont bien adaptées aux explications basées sur les variables d'entrées. Pour des systèmes à base de règles et d'exemples, il est possible, dans un esprit similaire, de demander de rédiger l'explication, ou de la choisir parmi un ensemble pré déterminé. Notre objectif est de comparer les explications générées par un outil à des explications de référence. Pour une instance particulière, nous pouvons séparer les explications humaines en "explication idéale" et "explication attendue". La différence entre ces termes est définie dans les deux paragraphes suivants.

Une explication idéale est une explication associée à une donnée d'entrée et sa classification correcte. Elle est idéale parce qu'elle s'applique lorsque la prédiction est juste. Cette explication est basée sur l'expertise du domaine. Si nous voulions entraîner un modèle à donner ces explications, nous les fournirions dans un jeu de données d'entraîne-

ment. Elles sont identiques, quel que soit le classifieur utilisé. Dans le cas d'une mauvaise classification, l'explication idéale ne reflète pas le comportement du modèle.

Une explication attendue est une explication associée à une donnée d'entrée et une classification, que celle-ci soit correcte ou non. L'utilisateur recevant le résultat peut alors espérer obtenir une explication qui soit fidèle au raisonnement, même si erroné, du modèle.

Explications de référence Nous collectons des explications de référence afin de mesurer des performances de méthode de génération d'explications, peu importe le résultat donné par le classifieur. Dans le chapitre suivant, les explications idéales sont utilisées comme référence dans les expériences réalisées. Les experts du domaine ayant de fortes contraintes de disponibilité, nous n'avons pas de personnes expertes à disposition pour associer des phrases avec leurs explications.

Documentation métier Afin de limiter les biais, nous déterminons les explications de référence grâce à la documentation métier. Cette documentation est constituée de documents techniques internes et du Guide d'Aide à la Rédaction des Offres (GARO). Le GARO est un document important pour les conseillers Pôle emploi, mais est peu pratique à utiliser lors des actes métier du fait de sa longueur : 128 pages. Ces documents nous permettent, à titre d'exemple, de déterminer que la discrimination sur le genre des candidats est détectée via l'intitulé de poste d'une offre. Cet intitulé sera alors considéré comme l'explication de référence en cas de rejet. Ces indications permettent de limiter les incertitudes dans la labellisation.

Spécificité du texte Des questionnements ont émergé lors de cette phase de labellisation manuelle. Les mots porteurs de sens devaient-ils être labellisés ? Quelle est la différence entre "Une assistante sociale" et "assistante sociale" ? À des fins de simplification, nous avons fait le choix d'ignorer ces mots non porteurs de sens.

Nous avons défini un cas d'usage et ses données associées. Nous pouvons désormais entraîner un modèle à résoudre notre problème de classification, et générer des explications pour ce modèle.

2.2 Génération d'explications par variables d'importance

Dans cette section, nous appliquons deux méthodes d'explication : la génération d'ancres sur n'importe quel modèle de [91] et l'utilisation de l'attention avec un modèle transparent de [66]. Ce modèle fournit directement des explications locales basées sur les variables d'entrées.

Pour rappel, notre choix a été guidé par les caractéristiques suivantes :

— nous ciblons les utilisateurs experts du domaine,

— l'explication est reçue lors de la rédaction d'une offre d'emploi,

— le modèle de classification est libre de toute contrainte de conception.

Nous optons pour des explications de portée locale, donnant des explications au cas par cas. La stratégie est libre, nous choisissons les deux extrêmes : modèle transparent et boîte noire. Nous appliquons deux méthodes d'explication basées sur les variables d'entrée.

Le modèle de référence sera un modèle transparent basé sur l'attention. Son avantage est de permettre la génération d'explications tout en conservant une architecture de réseau de neurone plutôt légère. La seconde méthode choisie est une méthode boîte noire, la méthode des ancres [91]. Ces travaux font suite à LIME [90], une méthode servant de référence en génération d'explications par approche boîte noire.

Les ancres seront générées sur les prédictions du modèle transparent. Ainsi, les deux explications seront basées sur le même modèle, limitant les différences entre explications aux seuls systèmes explicatifs.

En s'inspirant de [91] pour l'exemple, prenons la phrase "Ce film n'est pas mauvais", qui est classée "positive" par un modèle de classification binaire de sentiments, basé sur l'attention. L'explication des ancres serait $A = \{pas, mauvais\} \rightarrow Positive$. Chaque mot de la phrase posséderait un poids d'attention, et "pas" et "mauvais" auraient les poids les plus élevés. Les deux méthodes extraient des ensembles de mots d'intérêt.

Dans un premier temps nous présenterons le modèle transparent à attention, puis nous présenterons les ancres qui permettront d'expliquer ce même modèle.

2.2.1 Attention

Le mécanisme d'attention consiste à apprendre une couche spécifique d'attention à un modèle. Il est ainsi possible d'extraire des poids d'attentions associés aux variables

d'entrée d'une instance, qui correspondent à une pondération de focalisation du modèle.

Suivant les recherches de [66], nous concevons un réseau de neurones avec une architecture spécifique. Celle-ci combine un bi-LSTM, couche adaptée aux traitement de textes, et le mécanisme d'attention [66]. L'agencement des couches est détaillé dans la figure 2.4. La couche LSTM est la troisième en partant du haut. Les couches dense et dense1 forment l'attention, tandis que les deux couches suivantes créent la couche de représentation. Les détails de l'architecture sont présentés dans le tableau ci-dessous (cf. Table 2.1).

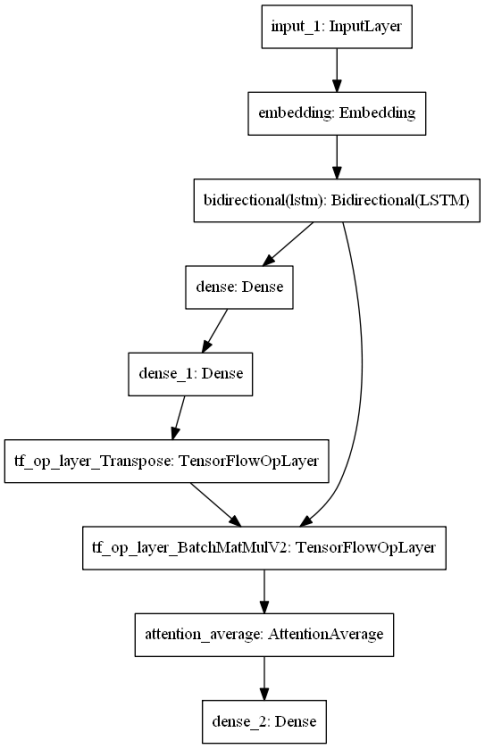


FIGURE 2.4 – Architecture du modèle à attention développé. Chaque couche ou calcul est représenté par un rectangle.

Le réseau a été adapté au cas d'usage, les dimensions des couches et autres commentaires sur l'architecture sont détaillés dans le tableau 3.1. Dans ce tableau, u est la taille du bi-LSTM, d_a la taille de la couche dense 0, r le nombre de têtes d'attention et M la matrice de représentation. Le mécanisme d'attention aboutit à une matrice d'attention A, qui est la sortie de la couche 5 (cf. Table 3.1). L'optimisation des hyper paramètres u, d_a a été effectuée en premier, puis le taux d'apprentissage a été ajusté sur l'intervalle $[5.10^{-5}, 10^{-1}]$. Les mots d'intérêt sont filtrés en utilisant un seuil t sur les valeurs d'attention. Lorsque le modèle ne prédit aucun rejet, l'explication est vide.

TABLE 2.1 – Architecture du réseau de classification à attention pour le cas d'usage LEGO.

ID	Type de couche	Paramètres	Commentaires
1	Couche d'entrée	80	La taille est le nombre de mots dans les textes
2	Plongement de mots	300	Plongement de mots GloVe
3	Bi-LSTM	u =50	La sortie est la matrice d'états cachés H
4	Couche dense 0	d_a =300	Activation $tanh$
5	Couche dense 1	r =1	La sortie est la matrice d'attention A
6	Représentation	sortie : $[2u, r]$	Combinaison de l'attention et de la couche cachée, $M = A^T * H$
7	Sortie	28	Couche de sortie

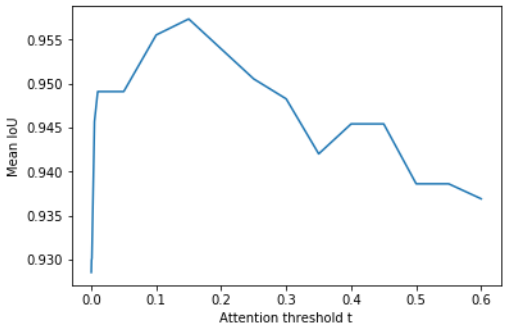


FIGURE 2.5 – Similarité (IOU) moyenne entre les explications générées et de référence dans l'ensemble de test, pour un seuil d'attention t variable. Les mots dont l'attention est supérieure ou égale à t forment l'explication. Le premier point est à 10^-4.

Pour ce classifieur, la matrice de plongement de mots utilisée est un plongement GloVe ("Global Vectors for Word Representation") de 300 dimensions [1]. Le taux d'apprentissage de $0,0005$ est déterminé par essais successifs. Ce réseau atteint un taux de reconnaissance de $83,67\%$ sur son ensemble de test. La matrice de confusion est présentée en annexe A.2.

Un seuil t est utilisé afin de filtrer les mots. t est déterminé en optimisant l'IOU sur un ensemble de test de 208 phrases, en comparant les explications à attention et la référence. Les résultats sont présentés dans le graphique de la Figure 2.5. Ils indiquent que les mots dont l'attention est supérieure ou égale à $0,15$ constituent une bonne explication. Nous générons ainsi des explications avec les seuils de valeurs $t = 0,15$.

Avantages et inconvénients Cette méthode de détection des mots importants a pour avantage de s'appuyer directement sur le fonctionnement interne du réseau. Le méca-

1. `https://dl.fbaipublicfiles.com/fasttext/vectors-crawl/cc.fr.300.vec.gz`

nisme d'attention permet de pondérer les entrées selon leur importance, avec pour objectif l'amélioration des performances du réseau. Ces pondérations sont ajustées lors de la phase d'entraînement du modèle, en optimisant les résultats de la tâche de classification. Ainsi, la génération d'explications ne nécessite pas de données supplémentaires. En faisant de l'anthropomorphisme, le réseau "apprend de lui-même à focaliser son attention sur certains éléments". Le second avantage de cette méthode est qu'elle ne nécessite pas de calcul après entraînement, hormis celui de l'inférence. Plus exactement, dans l'implémentation effectuée, le modèle est appelé une fois pour obtenir le résultat, et si la phrase est rejetée, une seconde fois pour extraire les poids d'attention.

Cette méthode souffre toutefois des critiques énoncées dans le Chapitre 1, en section 1.3.5. Elle n'est pas toujours fiable, l'attention apprise différant parfois fortement des autres méthodes d'explicabilité, qui ont acquis leur réputation dans la communauté scientifique. Si cette méthode limite les calculs lors de l'inférence, elle nécessite tout de même d'alourdir le modèle avec les couches 4, 5 et 6 du tableau 2.1 .

2.2.2 Ancres

Les ancres sont des règles extraites par analyse d'un ensemble de données générées et leurs sorties associées. Les données sont générées par perturbation de l'instance à expliquer. Les masquages permettent de déterminer quelles instances ont un impact sur le résultat final du modèle. Nous générons les ancres avec la bibliothèque python [2] développée par les auteurs de [91]. Afin de réduire les écarts d'explications aux seules performances des méthodes de génération des explications, les ancres sont générées en utilisant le modèle à attention développé en section 2.2.1. Les paramètres de génération des ancres sont détaillés dans la tableau ci-dessous 2.2.

TABLE 2.2 – Paramètre de génération des ancres pour le cas d'usage LEGO.

Paramètre	**Valeur**	**Commentaires**
Seuil	$0,95$	Précision minimale requise pour ajouter un nouveau mot
Delta	$0,1$	Marge d'erreur
Tau	$0,15$	Critère de précision de l'ancre
Taille de faisceau	4	Taille du faisceau de recherche, sélectionne N candidats à chaque tour
Taille de lot	100	Nombre de données générées

2. `https://github.com/marcotcr/anchor`

Dans un premier temps, rappelons quelques notations. Un domaine d'application est l'ensemble des instances pour lesquelles une règle A est vérifiée, $A(.) = 1$. A est une ancre si l'ensemble $\mathbb{E}_{D(Z|A)}$ des instances du domaine d'application sont classés comme l'instance à expliquer x, avec un taux supérieur à la précision souhaitée τ. On peut le formaliser par l'équation suivante :

$$\mathbb{E}_{D(Z|A)}[\mathbb{1}_{f(x)=f(z)}] \geq \tau \tag{2.1}$$

Calculer la précision d'une ancre n'est pas possible pour un modèle f et un ensemble D donnés. Nous calculons alors la probabilité que l'ancre corresponde au critère de précision τ en acceptant une marge d'erreur δ.

$$P(prec(A) > \tau) \geq 1 - \delta \tag{2.2}$$

On cherche enfin à conserver les ancres qui couvrent un plus grand nombre possible d'instances. La recherche s'effectue en partant d'une ancre vide, et en créant N ancres candidates par l'ajout de variables. Les paramètres du tableau 2.2 sont identiques aux paramètres de base des ancres. Toutefois nous avons dû augmenter la taille du faisceau de recherche, afin d'améliorer les performances.

Avantages et inconvénients La méthode des ancres permet de générer des explications sans s'inquiéter du fonctionnement interne du modèle. Les performances de ce dernier ne sont donc pas altérées. Les explications générées ont un cadre d'application défini.

Cependant cette méthode nécessite d'appeler le modèle un nombre important de fois. Elle est donc couteuse en calcul à chaque génération d'explication, et fonctionne mal sur un nombre trop important de variables d'entrées ; dans notre cas, sur les phrases longues. Enfin, il y a un risque de s'appuyer sur des comportements erratiques du modèle voire de créer des instances adversariales, en définissant le comportement sur un domaine d'application constitué de données hors distribution.

Explications générées Ces deux méthodes permettent de générer différentes explications présentées dans le tableau 2.3. Les explications par les ancres sont basées sur le modèle à attention, et deux explications reposent sur le mécanisme d'attention. Lorsque le modèle ne prédit aucun rejet, les explications générées sont forcées d'être vides.

3. Contrat à durée déterminée conduisant à un contrat à durée indéterminée

TABLE 2.3 – LEGO : phrases avec leurs différentes explications. Le texte est au-dessus des autres informations. Les phrases portent respectivement les numéros 0 et 73 du jeu de données LEGO - DE.

Rejet	**Référence**	**Ancres**	**Attention 0.15**	**Attention 0.5**
"[...] Notre agence de Saint-Medard-en-Jalles recherche une Assistante Administrative pour completer son equipe."				
Genre	['assistante administrative']	['recherche', 'Assistante', 'Jalles']	['assistante', 'administrative']	['assistante']
"Poste en CDD renouvelable en cdi."				
CDD possibilité CDI[3]	['CDD renouvelable en cdi']	['CDD renouvelable', 'cdi']	['cdi']	['cdi']

Nous avons détaillé les méthodes de génération d'explication permettant de détecter les mots importants dans les phrases d'offres d'emploi. Dans la section suivante, nous présentons différentes interfaces d'explications pour le cas d'usage LEGO.

2.3 Visualisations pour les explications locales

Les explications étant à destination d'humains, nous détaillons dans cette section la création d'un démonstrateur et des pistes de visualisations à présenter aux utilisateurs experts du domaine. L'objectif est de conduire un test d'utilisabilité à partir de ces visualisations pour déterminer quelles interfaces nous allons développer. Ces interfaces permettront la collecte des préférences utilisateurs.

Le démonstrateur est un prototype d'interface, reprenant des éléments de l'interface d'origine présentée en figure 2.1. Dans un premier temps, nous reprenons l'interface la plus simple possible, cf. figure 2.6. Le premier élément repris est le champ de texte pour le descriptif du poste. Nous ajoutons un bouton pour lancer la prédiction, remplaçant le bouton "poursuivre" de l'interface d'origine.

Lorsque l'utilisateur clique sur le bouton "Prédire", le résultat du modèle d'IA apparaît dans un encart sous le texte, présenté dans la figure 2.6b. Cette organisation diffère légèrement du bandeau explicatif situé au-dessus du texte dans l'interface d'origine en figure 2.1, mais permet une lecture de haut en bas. Le bandeau de résultat comprend un score de confiance du modèle. Plus le pourcentage est proche de 100%, plus la décision est tranchée. Un score plutôt faible, vers 50%, constitue une alerte. Les développements

(a) Démonstrateur avant prédiction

(b) Démonstrateur après prédiction, sans explications

FIGURE 2.6 – Démonstrateur présentant l'interface homme-machine du système d'explications locales, sans explications.

se basent sur les retours du test d'utilisabilité présenté dans le chapitre suivant, en section 3.1. Ils permettent la présentation aux utilisateurs des explications générées en section précédente, afin de collecter leurs préférences. Nous ajoutons ensuite à ce démonstrateur différents types d'explications locales aux utilisateurs, afin de leur permettre d'interagir en se rapprochant légèrement des conditions d'usage réel, tout en restant simple à développer et maintenir. Les visualisations proposées sont une explication par surlignage de la phrase puis des mots déclenchant le rejet, d'une règle métier associée et enfin d'un contre-exemple.

Les différentes pistes envisagées sont abordées avec leurs avantages et inconvénients. La première interface présentée est celle mettant en avant les phrases illégales. Puis vient l'explication basée sur les mots importants de cette phrase, suivie de celles basées sur les règles métiers, et les contre-exemples.

2.3.1 Préparation des illustrations

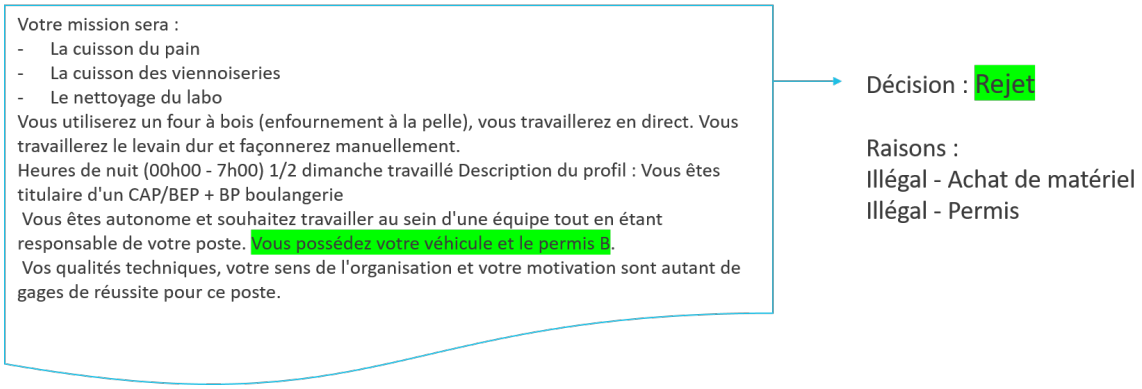
La préparation des illustrations est la première étape pour réaliser le test d'utilisabilité. Elles sont réalisées eu égard aux préoccupations du public cible ; les experts du domaine. Ces objectifs sont les suivants :

— corriger une offre rejetée,
— détecter si l'outil se trompe,
— créer de la confiance en l'outil.

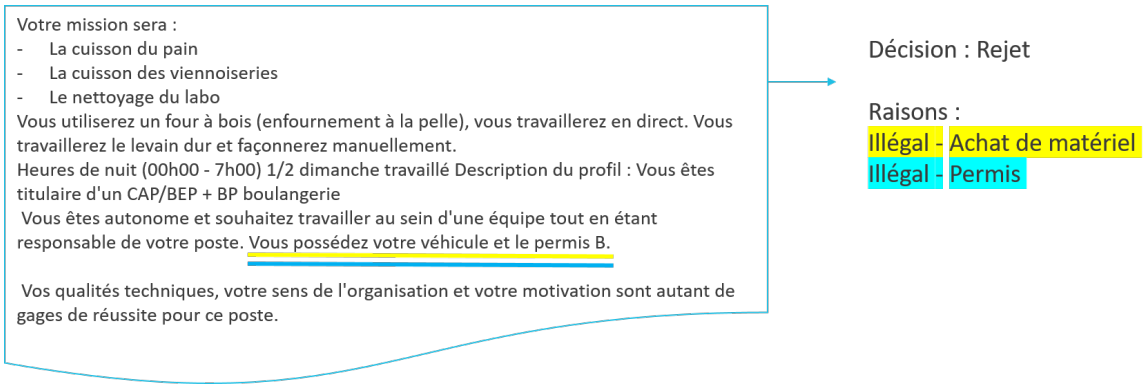
Les illustrations prennent en compte différents éléments de l'interface d'origine, présentée dans le chapitre 2. À savoir le bandeau d'alerte, la raison de l'alerte et le champ de descriptif du poste. Nous reprenons ces éléments et les repositionnons dans les illustrations, en partant sur quatre pistes, à savoir la mise en avant d'une phrase, d'un mot, d'un exemple et d'un contre-exemple. Chaque piste est illustrée et détaillée dans les paragraphes ci-après.

Illustration d'explication par phrase La première proposition d'illustration est la mise en avant de la ou les phrases entraînant le rejet d'une offre. Elle permet de donner le contexte du rejet, le texte étant analysé phrase par phrase. L'explication est donc à son niveau de contexte le plus large, laissant au receveur de l'explication le soin de comprendre le détail. Pour mettre en avant une phrase, de nombreuses options sont possibles. Nous en avons prédéterminé trois, présentées dans la figure 2.7. Il est ainsi possible de la surligner pour indiquer son rejet, comme dans la figure 2.7a. De même, pour faire apparaître plusieurs éventuels motifs de rejet pour une même phrase, nous pouvons la souligner comme présenté dans la figure 2.7b. Enfin, une autre proposition est d'afficher une bulle explicative lorsque le curseur de l'utilisateur survole la phrase, idée schématisée dans l'illustration en figure 2.7c.

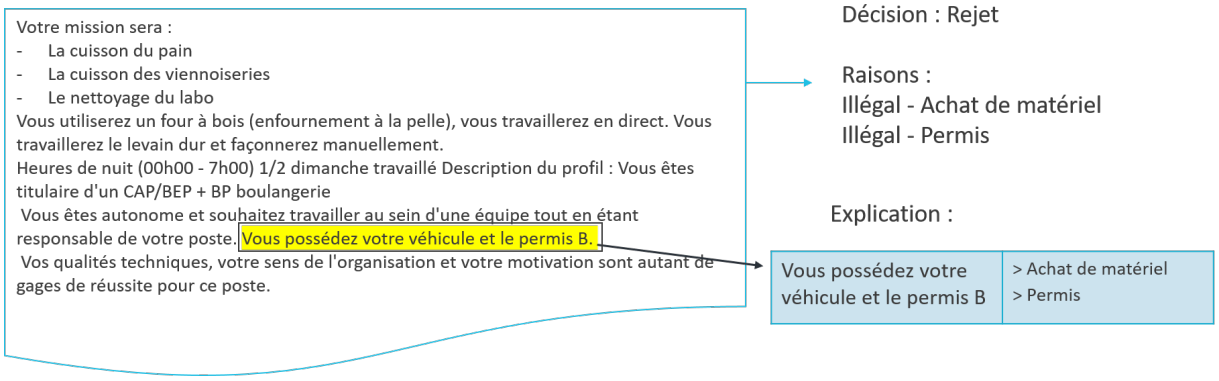
Illustration d'explication par mot La seconde proposition consiste à mettre en avant les mots de la phrase responsables du rejet. C'est l'explication avec le plus de précision, les phrases étant parfois longues. En contrepartie c'est l'explication qui donne le moins de contexte. Elle est également très proche du système expert d'origine, celui-ci étant basé sur des expressions régulières. Ce système original remontait donc des mots ou ensemble de mots prédéterminés et consignés dans un cahier des charges, sans tenir compte du contexte. Similairement aux phrases, nous proposons différentes variantes de maquette. Ces variantes sont illustrées dans la figure 2.8. La figure 2.8a montre les mots soulignés. Les



(a) Mise en avant d'une phrase rejetée par surlignage



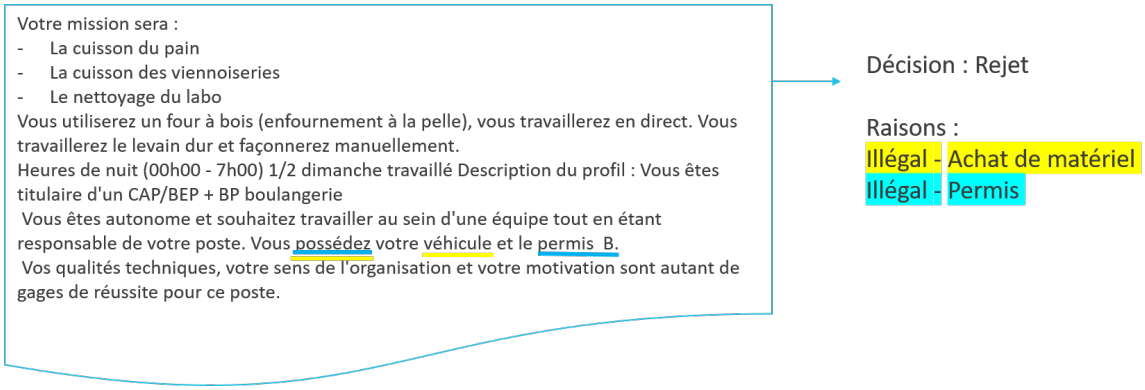
(b) Mise en avant d'une phrase rejetée par soulignage



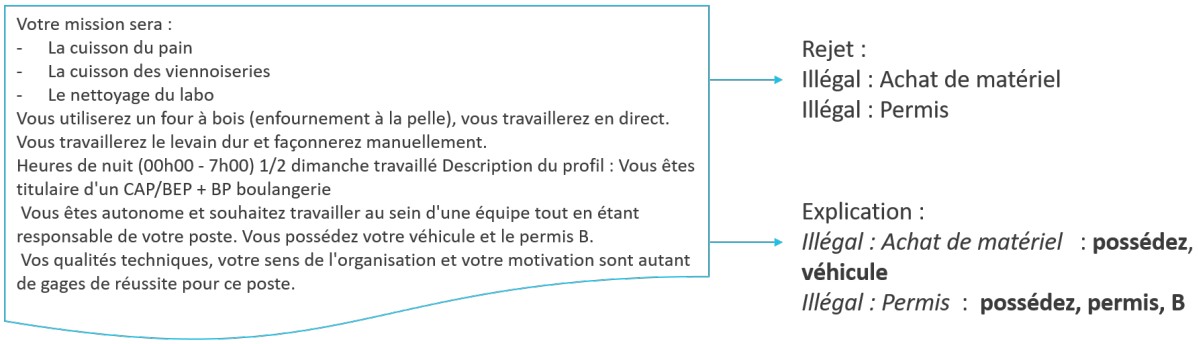
(c) Mise en avant d'une phrase rejetée via surlignage et bulle d'information apparaissant au survol de la phrase

FIGURE 2.7 – Illustrations de variantes d'une explication par la mise en avant d'une phrase.

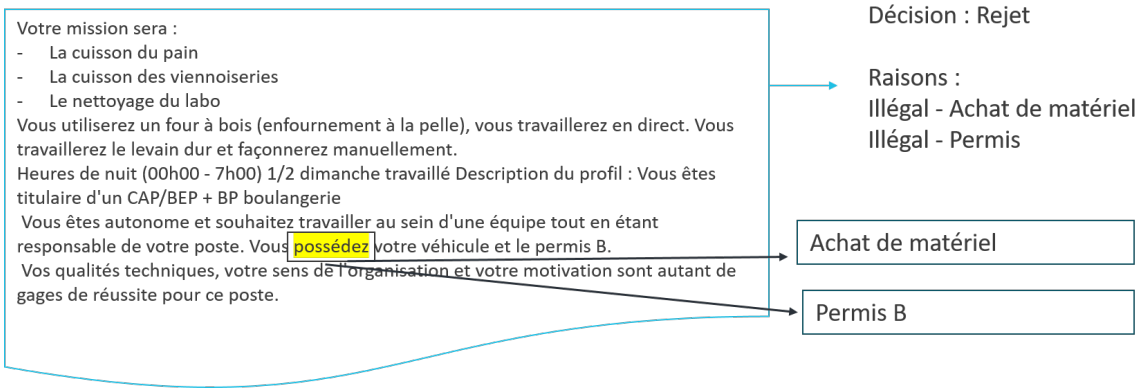
mots peuvent être surlignés, avec une bulle indiquant le motif de rejet affiché au survol du mot en question, comme illustré en figure 2.8b. Enfin, pour se rapprocher de l'interface d'origine, il est également proposé d'afficher le ou les mots en questions en dehors du champ texte, dans un bandeau à part. Cette maquette est illustrée en figure 2.8c.



(a) Mise en avant de mots par soulignage



(b) Mise en avant de mots par affichage dans un bandeau externe



(c) Mise en avant de mots via surlignage et bulle d'information apparaissant au survol des mots

FIGURE 2.8 – Illustrations de variantes d'une explication par la mise en avant de mots.

Illustration d'un exemple La troisième maquette proposée est l'explication par l'exemple, affichant pour chaque phrase rejetée une phrase proche, elle aussi classée sur le même motif de rejet. Ce type d'explication permet à l'utilisateur d'avoir un exemple de phrase qui entraîne un rejet, et qui comportera sans doute des mots identiques ou similaires à la phrase responsables du rejet. La figure 2.9 présente l'illustration proposée, avec une phrase proposée pour chaque motif de rejet détecté.

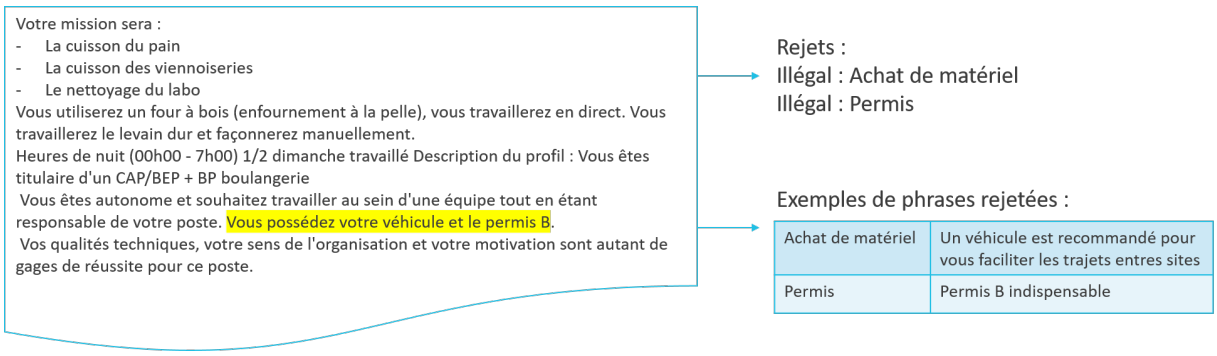


FIGURE 2.9 – Illustration d'explication par l'exemple

Illustration d'un contre-exemple Enfin, la dernière maquette que nous proposons est l'explication par le contre-exemple, affichant pour chaque phrase rejetée une phrase proche, mais cette fois qui n'est pas rejetée par le système. Ce type d'explication permet à l'utilisateur d'avoir un exemple de phrase acceptée, tout en étant proche sémantiquement de la phrase entraînant un rejet. Les contre-exemples sont illustrés en figure 2.10 avec comme précédemment une phrase par motif de rejet détecté.

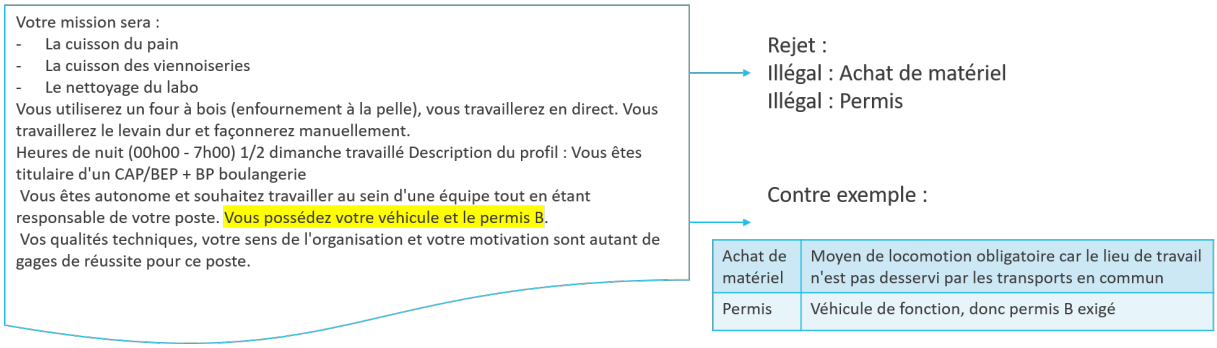


FIGURE 2.10 – Illustration d'explication par le contre-exemple

Toutes ces illustrations sont présentées aux personnes participant au test d'utilisabilité, présenté dans le chapitre suivant.

2.4 Conclusion

Dans ce chapitre, nous avons présenté le contexte applicatif de nos travaux, et les pré-requis à la réalisation de ceux-ci. Le cas d'usage illustrant nos travaux est développé en section 2.1. Nous avons généré des explications en section 2.2, et préparé diverses illustrations section 2.3.

La labellisation manuelle des données a mis en évidence des problématiques liées à la nature des données, avec la question de la prise en compte des mots vides de sens. La recherche d'une explication idéale qui corresponde aux préférences des utilisateurs est cruciale. Elle passe par la concertation des experts, mais également un travail de terrain coûteux de collecte d'une quantité significative d'explications, permettant d'obtenir un consensus.

Le temps limité de la thèse a amené à la labellisation d'une quantité restreinte de données. L'expérience ainsi acquise pointe vers la nécessité d'une définition collégiale des explications de référence. Avec une telle définition, il devient intéressant de collecter une quantité importante d'explications.

Nous avons généré des explications, collecté des préférences et déterminé des explications faisant office de référence. Dans le prochain chapitre nous allons déterminer comment comparer des méthodes d'explicabilité, sur la forme et le fond.

Résumé

- ✓ Nous avons labellisé un ensemble de données de test
- ✓ La définition d'une explication de référence de qualité n'est pas une évidence
- ✓ Nous avons généré des explications par mots importants
- ✓ Nous avons proposé différentes visualisation d'explications locales

Chapitre 3

Comparaison d'explications locales

Dans ce chapitre

Nous traitons ici l'évaluation des explications. Trois protocoles sont proposés, afin d'évaluer le format et la méthode de génération des explications. L'application des différents protocoles met en avant leurs intérêts et limites. Les protocoles d'évaluation des méthodes de génération présentent une forte dépendance à la disponibilité d'une explication de référence de qualité.

Dans le chapitre précédent, nous avons conçu différentes illustrations d'explications, et appliqué plusieurs méthodes de générations d'explications pour un même format. Nous allons maintenant mettre en place des protocoles de comparaison de ces formats et méthodes, avec et sans utilisateurs. L'application des protocoles montre leurs intérêts et limites respectives, notamment liées à la qualité des explications de référence.

La collecte des retours utilisateurs est présenté en section 3.1. Nous recueillons les avis d'experts sur différentes illustrations, et nous développons des interfaces en conséquence. Enfin, nous présentons la collecte des préférences des experts du domaine. Dans la section 3.2 les méthodes sont comparées en se basant sur des métriques objectives, permettant de valider le respect de critères donnés. Cette expérimentation se fait sans utilisateurs. Elle peut être menée rapidement et nécessite peu de prérequis. Dans la section 3.3, les utilisateurs cible sont intégrés à l'expérimentation, permettant de mesurer leurs préférences. Cette expérimentation nous permet de questionner la qualité des explications de références et la pertinence de la mesure utilisée en section 3.2.

3.1 Collecte des retours des experts

Dans un premier temps nous présentons les collectes des retours des utilisateurs experts du cas d'usage LEGO. En section 3.1.1 nous présentons les analyses à chaud et discussions avec le panel d'experts autour des illustrations du chapitre 2. Nous détaillons les interfaces créées grâce à ces retours en section 3.1.2 en précisant quelles solutions sont écartées, retenues, et quelles adaptations sont prises en compte. En section 3.1.3 nous présentons la collecte de préférences utilisateurs réalisée à partir de l'interface développée en section 3.1.2.

3.1.1 Test d'utilisabilité

Le point avec les experts débute par plusieurs rappels permettant de s'assurer que les éléments de vocabulaire, objectifs et éléments fonctionnels sont bien partagés par tous. Nous rappelons quel est l'acte métier étudié, à l'aide d'un exemple qui nous servira tout au long de la présentation. Cela nous permet de dé-corréler l'analyse de l'interface de l'analyse métier. La discussion est basée sur les illustrations présentées en chapitre 2.

Les personnes identifiées pour ce test sont toutes expertes dans l'analyse de la qualité des offres d'emploi. Elles sont quatre, de métiers complémentaires. Ce panel est ainsi constitué d'un chargé de relations entreprises, un manager de terrain, et un conseiller et une conseillère à dominante entreprise.

Une fois les illustrations présentées, les experts donnent leurs avis sur chacune des propositions, discutent entre eux et mettent en avant les éléments qui leur manquent. L'objectif de cette phase de discussion est de faire le tri sur les propositions et leurs variantes, ajuster les maquettes au besoin des personnes interrogées, et palier aux éventuels oublis et manques. Nous passons en revue chaque proposition et les nouvelles idées sont traitées à part.

Explication par phrase La mise en avant de phrases a été très appréciée par le panel d'utilisateurs. Iels apprécient le contexte global que fournit la phrase entière, notamment lorsque le descriptif de l'offre est long. Parmi les variantes, le surlignage est bien perçu tandis que le soulignage n'est pas apprécié. La bulle d'informations au survol de la phrase est fortement rejetée.

Explication par mot La mise en avant de mots a été bien reçue par les utilisateurs. Le système précédent fournissant des mots ou expressions également, cette solution ressemble fortement à ce qu'iels connaissent déjà, réduisant ainsi le cout de changement. Comme pour les phrases, la variante de surlignage des mots est plébiscitée, et les bulles lors du survol sont fortement rejetées. Les utilisateurs préfèrent le surlignage dans le texte plutôt que le bandeau à part. Si ce bandeau correspond à l'interface actuelle de DUNE (cf. figure 2.1), le surlignage direct réduit les allers-retours visuels.

Exemple L'explication par l'exemple n'a pas convaincu les utilisateurs. Iels ne voient pas l'intérêt de montrer une phrase proche et toujours rejetée, car cette phrase ne donne pas d'information intéressante. Les éléments communs pourraient être détectés par le surlignage de mots et le contexte global pourrait être mis en avant en surlignant la phrase rejetée. L'exemple ajoute de la lecture sans avantage comparé aux deux maquettes énoncées. Il n'apporte rien par rapport à leur expertise.

Contre-exemple Les explications par le contre-exemple en revanche ont été accueillies avec enthousiasme. Les experts ont trouvé un réel intérêt fonctionnel à cette maquette, car le contre-exemple peut être considéré comme une proposition de correction, si la phrase proposée est assez proche sémantiquement de la phrase en rejet. Du point de vue de l'acte métier, ce serait un gain de temps d'avoir seulement à valider ou ajuster la proposition de correction.

Proposition De nouvelles idées ont émergé en se basant sur les explications par l'exemple. Les experts proposent de fournir des explications sous forme de règles générales, en se référant au Guide d'Aide à la Rédaction des Offres (GARO). Ce document référence de 128 pages résume les bonnes pratiques et redirige si besoin vers des documentations plus complètes. Le GARO est long, et un récapitulatif affiché à chaque rejet pourrait éviter dans de nombreux cas de devoir s'y référer. Ce type d'explication serait particulièrement bienvenu dans le cadre de l'accompagnement d'employés novices sur la rédaction des offres.

Ces retours à chaud récupérés et consignés permettent la prise de décision sur les propositions, afin de les ajuster, abandonner, ou créer. Les personnes interrogées ont ainsi exprimé leur préférence pour les explications au niveau de la phrase, puis des mots et contre-exemples, puis des règles.

3.1.2 Solutions écartées et retenues

Dans les différentes variantes proposées, deux points font l'unanimité des personnes interrogées. Premièrement les bulles d'aides apparaissant en survolant des éléments ne sont pas appréciées. Deuxièmement, le surlignage des éléments mis en avant est apprécié, et détrône l'interface d'origine avec son bandeau d'informations.

Simplification du problème Cette discussion a acté la simplification principale des travaux menés : se limiter à la classification mono-label, multi-classe. C'est à dire, pour une phrase, ne considérer qu'un seul motif de rejet. Du point de vue fonctionnel, un utilisateur expert traitera probablement toutes les irrégularités d'un coup, où les éliminera une à une au fil de la mise à jour des motifs de rejet de la part du système. Cette simplification technique n'a donc pas d'impact négatif fonctionnel. De même, les personnes interrogées ont soulevé une inquiétude quant à la superposition des informations mises en avant, notamment par le soulignage, dans les rares cas concernés.

Règles générales Les utilisateurs proposent de remplacer les explications par l'exemple par des règles plus générales, apportant des précisions sur les bonnes pratiques consignées dans le GARO. En effet, ce type d'information est particulièrement adapté pour les personnes en formation, ou pour les cas rares ou ambigus. En donnant de précisions ou en redirigeant vers les bonnes pages du GARO, les rappels de règles permettent de gagner du temps et capitaliser sur des références déjà existantes.

La difficulté technique de l'implémentation d'un système d'explication n'est pas corrélée à son utilité perçue par les utilisateurs. Ainsi la préférence des personnes interrogées, basée sur les maquettes, est dans l'ordre de préférence décroissante : surlignage par phrase, puis à égalité mot surlignés et contre-exemples, et enfin l'affichage des règles. La difficulté quant à elle, a été, dans l'ordre décroissant : Génération de contre-exemples, surlignage des mots, affichage de règles et surlignage de phrases.

De manière globale, les retours des experts font écho à leur préoccupation principale : aller vite. Soit en ajoutant de l'information ou des précisions manquantes pour éviter d'aller chercher ces informations ailleurs, soit en faisant gagner du temps en proposant directement une correction. D'un point de vue utilisateur, surligner les mots est plus agréable que les phrases lorsque ces dernières sont longues.

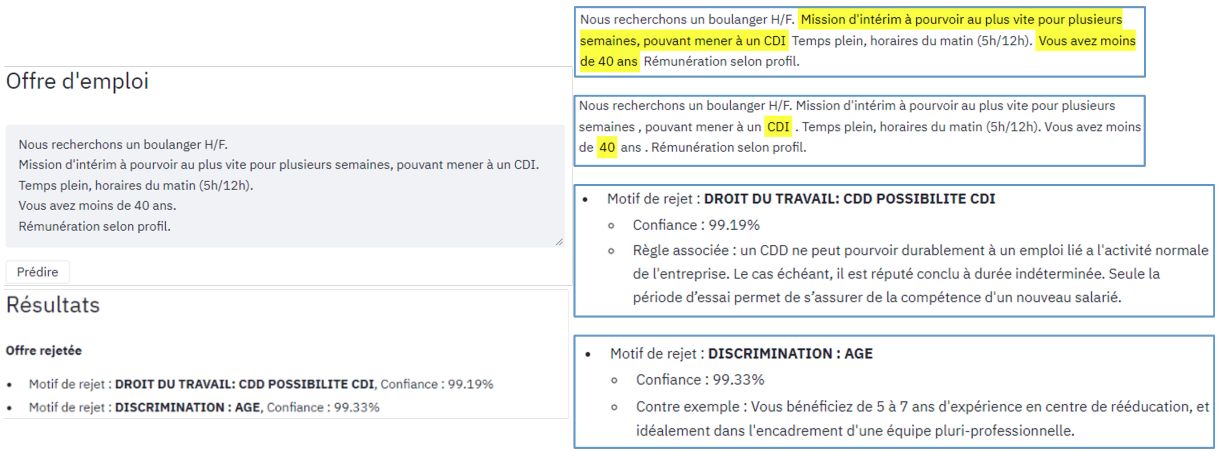


FIGURE 3.1 – Montage présentant un condensé des différentes interfaces développées.

Explication par phrase Cette explication consiste à surligner l'ensemble des phrases rejetées par le système, comme illustré dans la figure 3.2.

- Si le modèle d'IA renvoie un motif de rejet pour une phrase, alors le texte est réaffiché dans l'encart "Résultats", et la phrase en question est surlignée.
- Si plusieurs phrases sont surlignées, le texte n'apparaît qu'une fois et contient toutes les phrases surlignées.
- Les motifs de rejets sont affichés dans l'ordre d'apparition dans l'offre.

Ainsi, dans la figure 3.2, la phrase "Mission d'intérim à pourvoir [...] pouvant mener à un CDI." est associée au premier motif de rejet *Droit du travail : CDD possibilité CDI*.

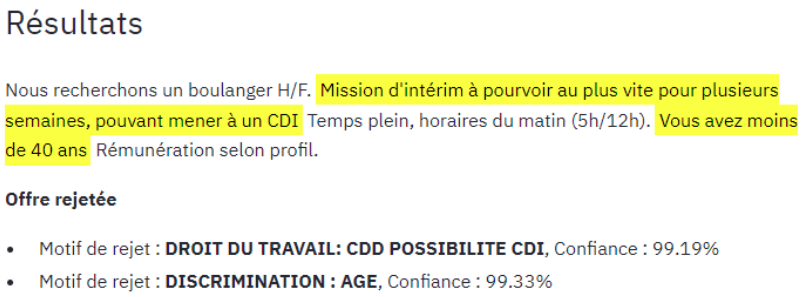
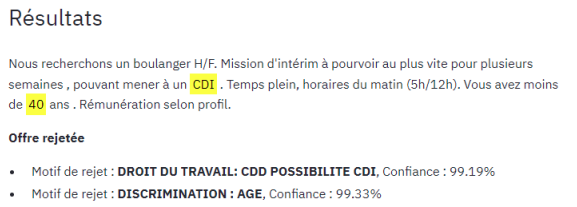


FIGURE 3.2 – Démonstrateur présentant l'interface pour l'explication par phrase(s) surlignée(s).

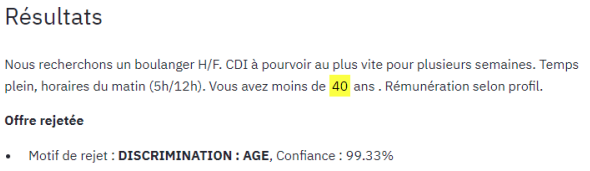
Ce type d'explication a l'avantage d'être peu couteux à mettre en place. Il est rapide, car ne nécessite pas de calcul supplémentaire après inférence. Il met en avant le contexte du rejet, laissant le soin à l'utilisateur d'analyser la ou les phrases en rejet. Toutefois, en

surlignant toute la phase, la visualisation est peu précise. C'est un problème si une phrase est longue, ou si un texte est mal formaté et ne comporte pas de ponctuation permettant de séparer les phrases. De même, le comportement interne du modèle n'est pas explicité.

Explication mot à mot Sur le même principe que l'explication par phrase, l'explication par mot surligne le ou les mots menant à la décision de rejet par le système. Comme illustré dans les figures 3.3, pour chaque phrase rejetée par le système, les mots sont surlignés dans le texte. Les motifs de rejets sont toujours affichés dans l'ordre d'apparition. Les mots à surligner sont déterminés par une méthode d'explication basée sur les importances de variables, ici avec la méthode d'attention. Plus de détails sur la sélection de ces mots sont donnés en section 2.2. Sur la figure 3.3a, le motif de rejet *Droit du travail : CDD possibilité CDI* est ainsi lié à la présence du mot "CDI" dans le texte. À noter que la seule présence du mot n'est pas une explication suffisante, le contexte étant également pris en compte. Par exemple dans la figure 3.3b, la seconde phrase n'est plus rejetée mais comporte toujours le mot "CDI".



(a) Explication pour deux phrases en rejet



(b) Explication pour une seule phrase rejetée

FIGURE 3.3 – Démonstrateur présentant l'explication mot à mot

L'explication mot à mot est précise, et reflète le comportement du modèle d'IA étudié, plus ou moins fidèlement selon la méthode d'explication employée. Néanmoins, ce type d'explication nécessite un calcul supplémentaire, afin de déterminer les mots responsables du rejet. Il complexifie l'outil, et ajoute un délai pour afficher l'explication.

Règle Afficher une règle métier constitue une explication courte, extraite des documents de référence et associée au motif de rejet détecté. Cette interface réalisée à la demande des utilisateurs est présentée en figure 3.4. Lors de la détection d'une phrase rejetée, le motif de rejet est affiché dans l'encart de résultats. Une règle associée est affichée juste en dessous. La figure 3.4 montre l'affichage de deux motifs de rejets et les règles associées. Une règle est toujours la même pour un motif donné. Elle est définie manuellement pour

chacun des 27 motifs de rejet, en résumant brièvement les informations présentes dans le GARO.

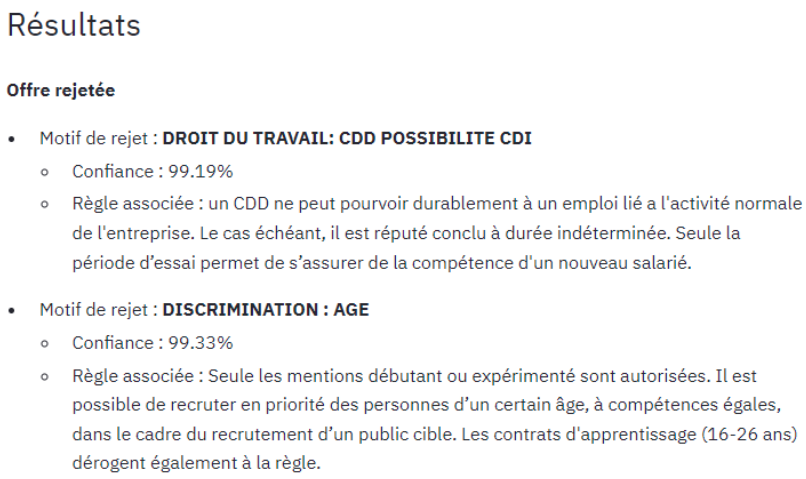


FIGURE 3.4 – Démonstrateur pour l'interface d'explication par la règle métier.

Cette interface a l'avantage d'être facile à mettre en place d'un point de vue technique. Son affichage est rapide, car elle ne nécessite pas de calcul. Elle est plus généralisable qu'un exemple. Elle est basée sur l'expertise métier et les documents de référence, ce qui la rend particulièrement adapté aux utilisateurs non experts, en leur fournissant un premier niveau d'information. En contrepartie, cette interface fournit des explications peu précises, parfois insuffisantes dans le cadre de cas délicats. Dans sa version implémentée, elle nécessite un travail manuel afin de résumer le document de référence.

Contre-exemple On peut également associer une phrase rejetée à une phrase proche de celle-ci, mais acceptée par le système. La figure 3.5 présente cette interface. La création des contre-exemples est une preuve de concept. C'est une version développée dans un temps court, peu efficiente, mais qui donne des pistes de réflexion. Les contre-exemples sont récupérés dans une base restreinte de 100 phrases légales, appelés ci-après *candidats*. La phrase illégale ainsi que tous les candidats sont rapprochés en les vectorisant via un plongement de mots, puis en rapprochant ces vecteurs avec une distance sémantique : la distance cosinus entre les vecteurs représentant les phrases. Le contre-exemple est la phrase légale ayant la distance cosinus minimale avec la phrase rejetée.

Ce type d'explication a l'avantage d'être contrastif, une explication bien perçue par les humains. Le contre-exemple est extrait d'un ensemble de phrases réelles, ce qui réduit le

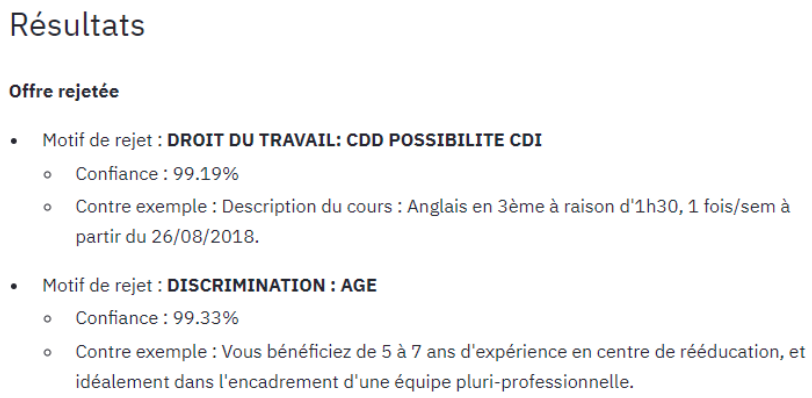


FIGURE 3.5 – Démonstrateur pour l'interface d'explication par le contre-exemple.

risque d'explications hors distributions, ressemblant à un élément réaliste, mais trompant le modèle d'intelligence artificielle. Enfin, dans ce cas d'usage, fournir un contre-exemple revient à proposer une correction automatique de l'offre, pour peu que la phrase de contre-exemple soit assez proche de celle d'origine. L'inconvénient de cette approche est qu'elle nécessite de bons candidats au contre-exemple. Il est nécessaire de trouver un moyen de filtrer ces bons candidats, car la comparaison par paires de phrase est couteuse en calculs. De même, l'explication avec des candidats qui sont peu intéressants car trop spécifique, ou trop éloignés du sens sémantique d'origine, sont peu intéressants pour les utilisateurs et pourraient générer plus de frustration qu'en enlever.

3.1.3 Préférences des utilisateurs

Nous utilisons l'interface d'explications par surlignage des mots précédemment présentée afin de recueillir les préférences des experts et expertes du domaine. Plusieurs méthodes de collecte sont possibles :

— La notation les explications, par exemple sur 5 points. Cela permet de donner une valeur absolue aux méthodes évaluées, mais implique la création d'un barème mental propre à chaque utilisateur.
— Le tri des explications. L'utilisateur a ainsi une vision d'ensemble des explications. Cette méthode permet d'ordonner des explications courtes.
— La comparaison par paire d'explications. Cette méthode est préférable pour ordonner des explications longues.

Nous nous intéressons ici uniquement à l'ordre de préférence et n'avons donc pas

besoin de cette valeur absolue. Nous présentons des phrases pouvant être longues, nous privilégions la comparaison par paires d'explication. C'est un outil efficace car il est plus facile pour un humain de dire quelle explication iel préfère, plutôt que donner une note.

Interface présentée Pour recueillir leurs préférences, les utilisateurs se voient présenter deux fois la même phrase dans l'interface d'explication par surlignage des mots importants. La classification du modèle d'IA et une explication par la règle sont également présentées. Ces éléments sont visibles sur la Figure 3.6 dans les deux zones sur fond blanc.

Les utilisateurs sont invités à choisir l'explication répondant à la question : "Quelle est l'explication la plus utile pour comprendre l'alerte ?", l'alerte étant la raison du rejet donné par le modèle. Comme le montre la Figure 3.6, la question est affichée en haut de l'écran pendant toute l'expérience. La réponse peut être une erreur du modèle. Dans ce cas, les explications fournies peuvent sembler hors sujet aux utilisateurs, comme discuté dans la section 2.1.2. L'explication peut être vide, ce qui signifie qu'il n'y a pas de mot mis en évidence. Si les utilisateurs ne perçoivent aucune préférence entre deux explications, on leur demande d'en choisir une selon leur sentiment subjectif. Comme plusieurs utilisateurs évaluent les mêmes phrases, ces cas apparaîtront dans les données comme des choix difficiles, où aucune explication ne ressort.

FIGURE 3.6 – L'interface de l'expérience. Les utilisateurs doivent choisir entre deux explications présentées celle qu'iels préfèrent. La raison du rejet et la règle associée sont affichées. Les explications sont des mots surlignés.

Contraintes expérimentales Pour lutter contre le biais d'apprentissage, 14 utilisateurs ont été recrutés par le biais du réseau interne de *Pôle Emploi*. Ce sont des employés spécialisés dans l'accompagnement des recruteurs, iels possèdent donc une expertise du domaine. Iels sont particulièrement habitués à rédiger et à éditer des offres d'emploi et travaillent depuis de nombreuses années avec une ancienne version de notre solution d'IA basée sur des expressions régulières (regex) qui lèvent des alertes légales. Par conséquent, les utilisateurs de cette expérience sont déjà conscients des cas possibles menant à des faux

positifs et négatifs pour le rejet automatique d'offres d'emploi. En définitive, ces experts ne sont pas sujets à des erreurs dues à la découverte du cas d'usage.

Nous avons également pris en compte la concentration, la fatigue et la faible disponibilité des utilisateurs. Les précautions suivantes ont été prises.

— La collecte est divisée en 9 sessions courtes, de 30 minutes chacune.
— Les sources de notification telles que les téléphones portables ou les logiciels de communicationsont mises en sourdine.
— Chaque utilisateur voit les paires dans un ordre différent.
— La tâche donnée est simple, l'interface possède très peu d'éléments, comme le montre la Figure 3.6.
— La tâche de comparaison est conçue pour être réalisée de manière autonome.
— L'expérience ne nécessite pas de conditions spécifiques telles que le niveau de lumière ou la colorimétrie de l'écran.
— En cas de blocage, les utilisateurs peuvent poser des questions par téléphone ou via leur outil habituel de messagerie instantanée.

Cette expérimentation a permis de recueillir les préférences de 14 utilisateurs sur 109 phrases. Tous n'ont pas eu la possibilité d'effectuer la totalité des 9 sessions. 104 sessions ont été réalisé sur les 126, soit 83% des comparaisons à effectuer. 9 participants ont réalisé la totalité des sessions. Eut égard de la difficulté de mobiliser les experts et expertes, il était prévu que toutes les comparaisons ne soient pas effectuées.

Nous avons présenté le test d'utilisabilité, de son déroulement en section 3.1.1 aux conclusions tirées des entretiens en section 3.1.2. Les retours des utilisateurs nous ont aidés à définir des interfaces mieux adaptées à leur besoins. La figure 3.1 montre les interfaces finales conçues dans le démonstrateur présenté dans le chapitre précédent. La section suivante présente l'évaluation de méthodes d'explications sans utilisateurs.

3.2 Évaluation sans utilisateurs

L'évaluation présentée fournit de premiers résultats, lorsqu'il n'y a pas d'utilisateur expert du domaine disponible. Nous cherchons à savoir quelle méthode d'explication convient le mieux à chaque cas d'usage dans son contexte donné. Ces expérimentations sans utilisateurs permettent de valider des propriétés telles que la concision ou la représentativité, ainsi que décrites en introduction.

Nous divisons notre problème en deux questions.

1. Cette explication est-elle proche d'une explication idéale ? Nous répondons à cette question de façon quantitative, en mesurant la proximité entre explications générées et de référence.
2. Lorsqu'il n'y a pas de référence, comment évaluer les méthodes d'explication ? Nous optons pour une approche qualitative, en filtrant pour analyser uniquement les cas où les explications générées sont peu similaires.

Les réponses à ces deux questions permettent d'établir une première évaluation des méthodes appliquées. Nous appliquons ce protocole aux explications générées dans le chapitre précédent. Nous nous assurerons notamment que les explications sont concises, fidèles et adaptées aux receveurs Nous présentons dans la section 3.2.2 l'étude quantitative des explications, puis en section 3.2.3 l'analyse qualitative.

3.2.1 Spécificités du cas d'usage Yelp

Le cas d'usage Yelp est utilisé afin de généraliser notre protocole. Il est basé sur l'application du même nom permettant à des utilisateurs de noter et donner des avis sur des commerces dont iels ont été clients. La tâche est de retrouver le nombre d'étoiles, entre 1 et 5, associées à un avis utilisateur. Si ce n'est la langue qui est l'anglais, Yelp est, comme le cas d'usage LEGO, un problème de classification de texte, multi-label, multi-classe. L'ensemble d'entraînement contient 453600 avis.

Ce cas d'usage requiert une version spécifique du réseau d'attention. Les textes sont vectorisés avec un plongement de mots spécifique à l'anglais, générant des vecteurs de 100 dimensions, et basé sur Wikipédia dans sa version anglophone [1]. L'optimiseur est Adam, avec un taux d'apprentissage de $0,0005$. Ce réseau obtient une précision de $74,63\%$ sur son ensemble de test. En comparaison, les auteurs de [66] présentent dans leur article une précision de 64.21% sur leur propre jeu de test.

L'architecture précise du modèle à attention est présentée en C2, pour le cas d'usage LEGO. Ici, nous présentons les différences entre les deux architectures.

Deux explications sont extraites de ce modèle : les explications par les ancres et les explications d'attention avec un seuil $t = 0,15$. Le tableau 3.2 illustre ces explications sur une évaluation du jeu de données Yelp. L'étude utilisateur n'est pas réalisée dans ce cas d'usage, il n'y a donc qu'un seul seuil d'attention généré. Comme l'explication des ancres sur de grands textes entraîne souvent des problèmes de mémoire, les ancres ont été

1. `https://wikipedia2vec.github.io/wikipedia2vec/pretrained/`

TABLE 3.1 – Comparaison des architectures des réseaux de neurones pour YELP par rapport à LEGO.

ID	**Type de couche**	**YELP**	**LEGO**	**Commentaires**
1	Couche d'entrée	300	80	La taille est le nombre de mots dans les textes
2	Plongement de mots	100	300	Plongement de mots Word2vec
3	Bi-LSTM	u =150	u =50	La sortie est la matrice d'états cachés H
4	Couche dense 0	d_a =350	d_a =300	Activation $tanh$
5	Couche dense 1	r =1	r =1	La sortie est la matrice d'attention A
6	Représentation	sortie : $[2u, r]$	sortie : $[2u, r]$	Combinaison de l'attention et de la couche cachée, $M = A^T * H$
7	Couche dense 2	1000	$\emptyset$	Activation $ReLu$, pour Yelp uniquement
8	Couche dense 3	5	28	Couche de sortie

appliquées à un sous-ensemble de 1060 évaluations les plus courtes sur les 2653 évaluations de l'ensemble de test complet.

TABLE 3.2 – Exemples extraits de Yelp : deux critiques avec des notes et des explications pour les ancres et les méthodes d'attention. Le texte est au-dessus des autres informations.

Note	**Ancres**	**Attention 0.15**
"Wow ! Superb Maids did an amazing job cleaning my house. They stayed as long as it took to make sure everything was immaculate. I will be using them on a regular basis."		
5	[]	['superb', 'amazing', 'everything']
"For the record, this place is not gay friendly. Very homophobic and sad for 2019. Avoid at all costs"		
1	['not']	['record', 'not', 'homophobic', 'sad', 'avoid']

Une fois les explications générées pour les deux cas d'usage, il reste un dernier type d'explication à récupérer pour mener à bien la suite des expérimentations : les explications des utilisateurs.

3.2.2 Analyse quantitative

Dans un premier temps nous cherchons à répondre à la première question : "Cette explication est-elle proche d'une explication idéale ?". Pour cela, nous comparons les explications générées à l'explication de référence, considérée comme le meilleur résultat

possible. Ces mesures ont pour objectif de savoir quelles sont les explications les plus fidèles, adaptées au receveur et concises.

La concision des explications est mesurée par le nombre de mots des explications. Pour quantifier l'adaptation au receveur et la fidélité, les explications générées sont comparées à la référence (voir section 2.1.2). Une métrique simple de similarité entre deux ensembles a été reprise, l'IOU, et est utilisée parmi d'autres métriques connues (précision, score F1...). Tel qu'utilisé dans [14], l'IOU, l'exactitude et le score F1 sont également comparés, ainsi que le rappel et la précision utilisés dans le score F1. Le rappel est une mesure intéressante car elle n'est pas affectée par les vrais négatifs

La concision est peu différenciante pour Yelp, les longueurs moyennes des explications étant similaires, 2,34 et 2,13 mots pour les ancres et l'attention respectivement. Pour LEGO, les explications basées sur les ancres sont en moyenne plus courtes que celles basées sur l'attention. Les longueurs moyennes sont respectivement de 0,15 et 0,33 mots dans l'ensemble de test. La valeur moyenne est faible en raison des explications vides. Ces mesures comparent la référence aux explications générées pour le jeu de données LEGO - Bonnes Prédictions (BP). Pour obtenir des mesures équitables, les mots vides ne sont pas pris en compte. L'évaluation du modèle n'étant pas le but de cette expérimentation, le jeu de test utilisé est le jeu de données LEGO - BP de 147 phrases correctement prédites.

TABLE 3.3 – Comparaison des explications générées avec la référence, jeu de données LEGO BP. Les meilleurs résultats sont en gras. La méthode des ancres obtient des résultats légèrement meilleurs.

Mesure	**Ancres**	**Attention**
IOU	$0,94$	$0,93$
Taux de reconnaissance	$0,98$	$0,97$
Rappel	$0,97$	$0,98$
Précision	$0,96$	$0,94$
F1	$0,97$	$0,96$

Les résultats de toutes les métriques pour le cas d'usage LEGO sont affichés dans le tableau 3.3. Dans l'ensemble, comparées à la référence, les deux méthodes de génération d'explication obtiennent des résultats similaires, cf. Tableau 3.3. Les scores sont élevés, en partie à cause du nombre de phrases non rejetées dans le jeu de données LEGO - BP : 131. Ces instances n'ont pas d'explication, les différentes métriques sont dans ce cas égales à 1.

Les ancres et l'attention sont également comparées les unes aux autres. En l'absence

de référence, les métriques telles que la précision et le score F1 ne sont pas pertinentes. Une IOU élevée indique que les explications sont similaires dans les deux méthodes. L'IOU entre les ancres et l'attention est de $0,92$, ce qui indique qu'elles donnent des résultats similaires. Pour le cas d'usage LEGO, les explications des ancres et de l'attention sont toutes deux similaires et proches de l'explication idéale.

Comme il n'y a pas de documentation ni d'expert du domaine pour le cas d'usage Yelp, il n'y a pas de référence pour son jeu de données. Par conséquent, la comparaison n'est possible qu'entre les ancres et les explications d'attention. L'IOU indique si les explications générées sont similaires. La moyenne de l'IOU sur l'ensemble de test réussi est de $0,23$, ce qui montre de fortes différences entre les deux méthodes d'explication. Cela peut s'expliquer par les longs textes et le vaste vocabulaire attendu dans les explications. Par conséquent, pour évaluer les méthodes d'explication dans le cas d'usage de Yelp, une analyse qualitative est nécessaire.

En définitive, l'analyse quantitative donne une première analyse sur chaque cas d'usage. Pour LEGO, les ancres sont plus concises, et aussi fidèles et adaptées que l'attention. Pour Yelp, les deux méthodes sont proches en concision, mais diffèrent beaucoup en contenu. Le manque de données ne permet pas de juger de leur fidélité ni de leur adaptation aux personnes cibles.

3.2.3 Analyse qualitative

Cette première analyse nous amène à la deuxième question : "Lorsqu'il n'y a pas d'explication humaine, comment évaluer les méthodes d'explication ?" En effet il n'y a pas toujours de données de référence à disposition, c'est le cas pour Yelp. Dans ce cas, l'alternative est l'analyse qualitative par un humain, une tâche longue et coûteuse. Nous proposons une évaluation qualitative de la fidélité et l'adaptation des ancres et de l'attention de manière efficiente, en effectuant un filtre sur les exemples d'intérêt.

Ces éléments pertinents sont ceux pour lesquels les explications à comparer sont très différentes. Cela permettra de déterminer si une méthode d'explication est plus précise lorsqu'elle est différente. Dans le contexte d'une expérience sans experts du domaine, cela permet à un expert en données de procéder à la première évaluation des différentes méthodes d'explication.

Ce filtrage sera donc utilisé dans l'analyse qualitative suivante pour les deux cas d'usage. Les phrases d'intérêt sont celles avec une forte différence entre explications générées par les différentes méthodes. Le filtrage de ces phrases se fait donc sur l'IOU entre

méthodes, en conservant les phrases pour lesquelles l'IOU est la plus faible.

TABLE 3.4 – Textes de LEGO avec différentes explications (IOU inférieure à $0,5$). Le texte est au-dessus des autres informations.

Motif de rejet	**Vérité terrain**	**Ancre**	**Attention 0.15**
"Contrat a duree indeterminee - Dfd Notre agence de Saint-Medard-en-Jalles recherche une Assistante Administrative pour completer son equipe."			
Genre	['assistante, administrative']	['recherche', 'Assistante', 'Jalles']	['assistante', 'administrative']
"Nous recherchons actuellement un Teleconseiller FRANCAIS / NEERLANDAIS (H/F) pour le compte de notre client, a Marcq-en-Baroeul."			
Nationalité	['francais, neerlandais']	['un', 'neerlandais', 'recherchons', 'francais']	['neerlandais']

Pour le cas d'usage LEGO, le tableau 3.4 donne des exemples où la valeur de l'IOU est inférieure à 0,5. Cette analyse qualitative indique que les explications d'attention sont une meilleure méthode d'explication pour ce cas d'usage.

TABLE 3.5 – Textes de YELP avec explications. Le texte est au-dessus des autres informations.

Étoiles	**Ancres**	**Attention 0.15**
Wow ! Superb Maids did an amazing job cleaning my house. They stayed as long as it took to make sure everything was immaculate. I will be using them on a regular basis.		
5	[]	['superb', 'amazing', 'everything']
For the record, this place is not gay friendly. Very homophobic and sad for 2019. Avoid at all costs		
1	['not']	['record', 'not', 'homophobic', 'sad', 'avoid']
Had the best experience buying my dress at brilliant bridal in jan 2018. Can't wait to wear my beautiful gown in oct 2018		
5	['brilliant']	['best', 'buying', 'can']

Le tableau 3.5 illustre des exemples d'évaluations Yelp et les explications associées pour les évaluations extrêmes (5 et 1 étoiles) lorsque les explications sont différentes. Il indique un manque de mots significatifs dans les ancres. Comme les longueurs moyennes sont similaires et que l'attention semble plus précise lorsque les explications sont très différentes, cette analyse qualitative indique que les explications basées sur l'attention sont un choix plus sûr dans ce cas d'usage particulier.

Pour le cas d'usage Yelp, les deux explications mettent en évidence les mêmes parties lorsque les évaluateurs mentionnent leurs propres évaluations. Cela conduit même à des

prédictions erronées, comme le montre le tableau 3.6. Puisque le texte mentionne 2 étoiles, l'évaluation prédite est de 2 étoiles. La note réelle était de 3 étoiles, mais les explications font toutes deux ressortir le "2" du texte.

TABLE 3.6 – Influence de la notation montrée par explication dans le cas d'usage Yelp.

Description	**Note**	**Note prédite**	**Ancres**	**Attention 0.15**
"[...] And this is the reason I gave them a mere 2 stars[...]"	3	2	['2', 'stars']	['2']

L'évaluation sans explication de référence tend à la même conclusion pour les deux cas d'usage. L'IOU permet de sélectionner des phrases intéressantes à analyser, diminuant l'effort d'analyse. L'explication par l'attention s'est avérée être la méthode la plus fidèle.

En résumé, l'expérience sans experts du domaine donne des indications sur différentes propriétés. Premièrement, les explications par les ancres sont plus concises pour LEGO, mais de taille similaire aux explications par attention pour Yelp. Enfin, les explications par attention sont plus fidèles pour Yelp, et légèrement plus fidèles et adaptées aux receveurs pour LEGO. Selon la propriété, fidélité ou concision, à privilégier pour LEGO, l'une ou l'autre méthode sera à choisir. Pour Yelp, l'analyse qualitative met en avant les explications à attention.

Cette première expérience a été conçue pour donner une première évaluation de plusieurs méthodes d'explication lorsque les utilisateurs et utilisatrices experts sont hors de portée. La section suivante présente une expérience plus importante, avec des utilisateurs impliqués.

3.3 Étude psychométrique avec experts du domaine

Cette étude vise à évaluer les méthodes d'explication lorsque des utilisateurs et utilisatrices experts du domaine sont disponibles. Cette étude est illustrée avec le cas d'usage LEGO. Yelp ne sera pas traité dans cette partie car nous ne disposons pas de sujets experts pour appliquer le protocole décrit.

Cette analyse répond à deux questions : "Quelle explication est préférée par les experts du domaine ?" et "Comment faire en sorte que la métrique quantitative corresponde aux préférences des utilisateurs ?". Comme le but est de travailler avec les préférences, le jeu de données de test est différent de celui de la section 3.2. Le jeu de données LEGO - DE de

106 phrases sera utilisé. Ce filtre évite que les utilisateurs se retrouvent avec des explications identiques à comparer. Les préférences humaines sont comparées aux performances mesurées par similarité entre explications générées et explications de référence.

La section fournit une vue d'ensemble du protocole d'étude psychométrique, ainsi que nombre de détails permettant son implémentation adaptée. Les résultats obtenus par l'application de ce protocole ne donnent pas de méthode vainqueure claire au global. Une analyse approfondie des résultats montre que les méthodes préférées évoluent selon la complexité des explications de référence. La notion de concision est ainsi à prendre en compte dans l'évaluation des performances.

Le protocole expérimental est détaillé en section 3.3.1. Les calculs nécessaires pour classer les méthodes par performance et préférence sont respectivement présentés en sections 3.3.2 et 3.3.3. Enfin, les résultats sont présentés et analysés en section 3.3.4.

3.3.1 Protocole expérimental

Cette section présente le protocole de ce deuxième volet d'évaluation de méthodes d'explication. Ce processus permet d'évaluer les méthodes d'explication qui prennent en compte les préférences des utilisateurs. L'objectif est double. Premièrement, recueillir les préférences humaines pour les méthodes d'explication. Deuxièmement, valider une métrique quantitative pour estimer les préférences des utilisateurs. En utilisant la figure 3.7 comme support, le processus complet est décrit en suivant les flèches du diagramme.

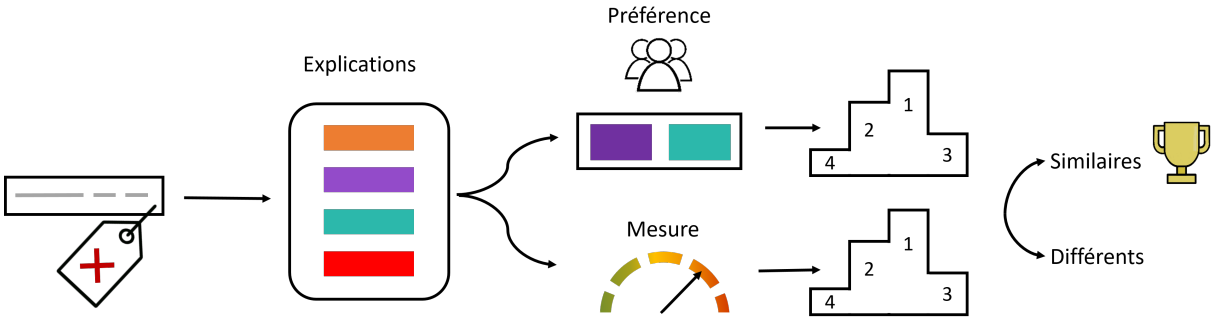


FIGURE 3.7 – Processus global de l'expérience, avec étude psychométrique des utilisateurs et analyse quantitative.

Tout d'abord, des données étiquetées sont fournies au modèle et aux méthodes d'explication à évaluer. Comme la comparaison par paires est réalisée plus tard, nous recommandons de ne considérer que quelques méthodes d'explication évaluées à la fois.

Pour obtenir la métrique quantitative, les explications générées sont comparées à la référence. La première version est une métrique simple, ici l'IOU comme vu dans la section 3.2.2. Encore une fois, la référence est considérée comme la meilleure option par conception et obtient le score maximal de 1. Les autres explications obtiennent des scores compris entre 0 et 1, 1 étant le meilleur. En classant les méthodes d'explication par leur score moyen, on obtient le classement quantitatif.

La comparaison par paires du chapitre précédent donne une matrice de préférences. L'élément i, j de cette matrice indique le nombre de fois où i a été préféré à j. Pour obtenir un classement à partir de cette matrice, l'algorithme Bradley-Terry-Luce (BTL) [16, 70] est utilisé. Il permet de transformer une matrice de comparaison par paires en un classement ordonné de tous les éléments comparés. Nous obtenons ainsi une échelle psychométrique ordonnant les quatre méthodes d'explication : la référence et les explications générées.

Une fois les podiums définis, nous pouvons les comparer. Si le classement de la métrique quantitative est similaire aux préférences humaines, nous pouvons estimer sans risque que la métrique reflète les préférences des utilisateurs. La métrique peut être conservée et réutilisée avec d'autres méthodes d'explication si nécessaire. Si le classement de la métrique quantitative n'est pas similaire aux préférences des utilisateurs, alors notre métrique quantitative doit être améliorée. Une analyse des données de cette expérience donnera des pistes sur ce qu'il faut améliorer.

Ce protocole est appliqué au cas d'usage LEGO. Toutefois, il doit être adapté pour chaque cas d'usage, avec si besoin un format de données différent, un nombre variable de méthodes d'explication, et une métrique quantitative personnalisée.

La génération des explications est décrite dans le chapitre précédent. Comme les ancres doivent être générées avec un modèle, le modèle d'attention est utilisé pour générer l'étiquette. Cela garantit que les différences de modèle n'auront pas d'impact sur l'expérience. Dans ce travail, nous comparons les ancres et deux explications de l'attention avec des seuils de valeurs $t = 0,15$ et $t = 0,5$. La deuxième explication avec un seuil $t = 0,5$ est générée spécifiquement pour cette expérimentation. Comme seul le seuil est plus élevé, les explications avec *attention 0,5* sont un sous-ensemble des explications avec *attention 0,15*, avec les mêmes mots ou plus concises. Elles permettent d'étudier le niveau d'information attendu par les utilisateurs. Pour chaque phrase, une explication de référence par attention humaine est également établie, tous les détails sont donnés dans la section 2.1.2 du chapitre précédent. Une entrée génère quatre éléments à comparer. Maintenant que l'expérience globale a été présentée, nous procédons à la mesure quantitative de qualité

des explications.

3.3.2 Mesure quantitative

Dans l'expérimentation précédente (cf. section 3.2) sur le jeu de données LEGO - BP, nous avons considéré la performance des explications en tant que similarité entre explication générées et de référence. Cette similarité est mesurée avec l'IOU. Dans la section courante, nous effectuons cette même mesure, cette fois sur le jeu de données LEGO - DE, puisque la collecte des préférences utilisateurs contraint à ne pas avoir 2 fois la même explication,

La comparaison de la métrique quantitative avec les préférences utilisateurs implique de recalculer l'IOU dans ce jeu de données pour définir le podium de la métrique quantitative. En raison de cette variation dans la conception du jeu de données, une baisse conséquente de l'IOU est attendue par rapport à l'IOU calculé sur le jeu de données LEGO BP dans la section 3.2 : les phrases conformes du jeu de données LEGO - BP ne sont plus présentes dans le jeu de données LEGO - DE, et tiraient les résultats vers le haut. Le jeu de données LEGO - DE ne contient lui aucune phrase conforme.

L'IOU est calculé avec le même protocole qu'en section 3.2, les mots d'arrêt étant ignorés. Comme le montre la première ligne du tableau 3.7, les scores baissent de manière significative par rapport aux chiffres présentés dans le tableau 3.3 ce qui était attendu.

Une analyse plus poussée montre que, dans notre cas, cette baisse est principalement due au fait de prendre en compte les cas où le modèle de prédiction est en échec, ces cas étant filtrés dans le jeu de données LEGO - BP. Ainsi, l'IOU entre les ancres et les explications de référence pour le jeu de données DE est en moyenne de $0,27$. Pour les cas où le modèle est en succès elle est en moyenne de $0,52$, pour les cas où il est en échec, l'IOU est de $0,02$.

TABLE 3.7 – Comparaison objective des ancres et des explications d'attention. Les explications sont comparées à la référence avec l'IOU, sur le jeu de données LEGO - DE.

Type d'explication	**Ancres**	**Attention 0,15**	**Attention 0,5**
Total (106)	0,27	0,23	0,23
Vide (53)	0,02	0,13	0,21
Simple (12)	0,53	0,58	0,46
Complexe (41)	0,51	0,25	0,20

La première ligne du tableau 3.7 montre une faible différence entre les méthodes. Une

analyse plus poussée dans ce cas d'usage implique de différencier les phrases en fonction de leur explication de référence, cette analyse est présentée dans les lignes suivantes du tableau 3.7. Les phrases conformes n'ont pas de référence. Ces phrases constituent la catégorie d'explication *vide*. Certaines phrases ont des explications attendues *simples*, composées d'un ou deux mots. Enfin, nous regrouperons les explications attendues de trois mots ou plus dans la catégorie *complexe*. Ce regroupement nous donne les scores moyens de similarité pour la Figure 3.8.

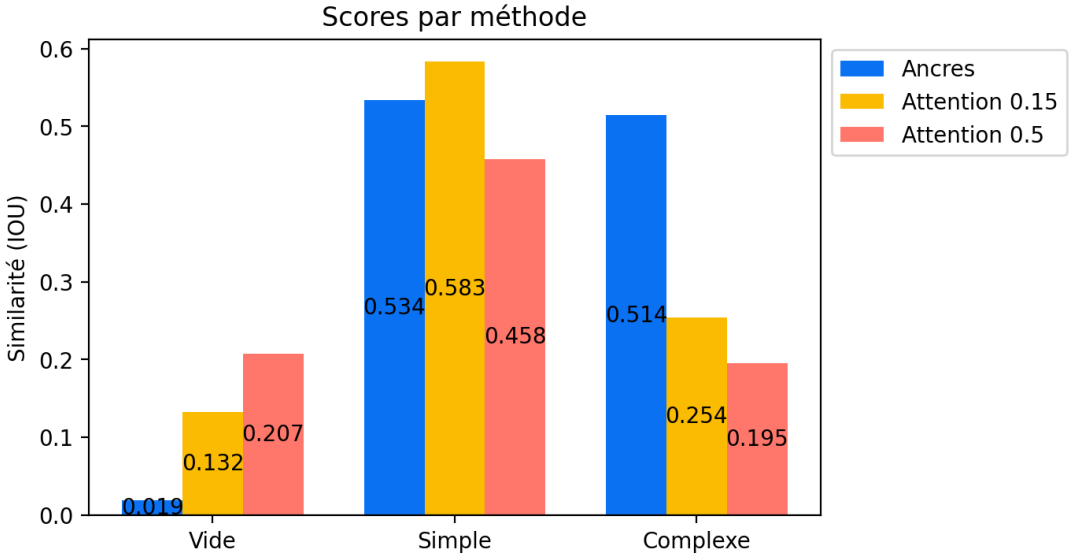


FIGURE 3.8 – Scores moyens de similarité aux explications de référence, obtenus avec l'IOU pour chaque méthode d'explication, sur le jeu de données LEGO DE.

Le premier groupe de la figure 3.8 montre que les trois explications générées ont une faible IOU, en particulier la méthode des ancres. Cela s'explique par le fait que l'IOU avec une explication de référence vide ne peut être que de 0 ou 1. Une IOU de 0 est obtenue lorsqu'une méthode donne n'importe quel mot comme explication, donc l'intersection de cette explication et de l'absence d'explication est de longueur 0. Lorsqu'une méthode ne donne aucune explication, nous comparons deux ensembles vides. Comme ils sont égaux, l'IOU est égale à 1. Comme les méthodes des ancres et d'attention ne donnent rarement aucune explication, leur IOU est très faible par rapport à l'explication humaine, vide. Le deuxième groupe met en évidence les bonnes performances des méthodes des ancres et d'attention avec $t = 0.15$ pour imiter l'explication humaine donnée. Le dernier groupe, pour les cas complexes, indique une divergence significative des deux méthodes d'attention par rapport à l'explication humaine. Cependant, la méthode des ancres conserve une IOU

stable entre le 2ème et le 3ème groupe.

TABLE 3.8 – Podium basé sur la mesure de similarité, sur le jeu de données LEGO DE, par catégorie d'explication. l'explication humaine sert de référence pour le calcul de l'IOU, elle n'apparaît donc pas dans ce tableau.

	Premier	**Second**	**Troisième**
Vide	Attention 0.5	Attention 0.15	Ancre
Simple	Attention 0.15	Ancre	Attention 0.5
Complexe	Ancre	Attention 0.15	Attention 0.5

En définitive, les scores obtenus peuvent être traduits dans le podium du Tableau 3.8. Pour rappel, les explications de référence servent au calcul de l'IOU, elles sont considérées comme les meilleures explications pour chaque cas. Nous avons donc un classement avec l'explication humaine, et les trois autres méthodes dans l'ordre comme indiqué dans le tableau 3.8. Ces podiums peuvent maintenant être comparés aux podiums des préférences des utilisateurs. Ces préférences utilisateurs sont abordées dans la section suivante.

3.3.3 Préférences des experts

Dans cette section, nous classons les types d'explications étudiés selon les préférences des experts du domaine. Nous nous appuyons sur les comparaisons par paires dont la collecte est détaillée dans le chapitre précédent en section 3.1.3. Ces comparaisons permettent le calcul du classement global des méthodes.

Ce classement des préférences des experts sera par la suite comparé aux mesures quantitatives. Toutefois en l'état, les comparaisons par paires ne donnent pas de comparaison globale des méthodes d'explications. Nous allons donc dans cette section transformer ces comparaisons par paires en classement global.

$A = (a_{ij})_{m \times m}$ est la matrice de préférences issue de la collecte de comparaisons par paires. Chaque coefficient a_{ij} est le nombre de fois où le stimulus (l'explication) S_i a été préféré au stimulus S_j. La diagonale de la matrice est nulle. Le nombre de comparaisons est $n_{ij} = a_{ij} + a_{ji}$ et la probabilité que S_i soit préféré à S_j est $P_{ij} = a_{ij}/n_{ij}$.

Nous obtenons un score de crédit v_i à partir de la matrice de préférences grâce au modèle BTL [16, 70]. Ce modèle se base sur les comparaisons par paires pour estimer les scores v_i selon l'équation 3.1. Ils sont calculés itérativement pour tous les éléments i, puis normalisés en les divisant par la somme des v_i, jusqu'à ce que les scores convergent à un

état stable.

$$v_i = \frac{A_i}{\sum_{j \neq i} \frac{a_{ij}+a_{ji}}{v_i+v_j}} \tag{3.1}$$

Ici, le score de crédit v_i représente la probabilité qu'un utilisateur choisisse une méthode plutôt que les autres. Pour un stimulus donné i, A_i est la somme des a_{ij} pour tout j. On obtient alors une échelle psychométrique telle que celle de la figure 3.9. Un score plus élevé indique la préférence des utilisateurs. Comme le score v_i est calculé par rapport aux autres éléments, il n'est pas possible de comparer les scores des éléments i d'une comparaison, avec les éventuels scores v_k obtenus par l'application du BTL à un ensemble différent d'éléments.

Nous illustrons le calcul des scores avec le cas d'usage LEGO. Les scores BTL peuvent être évalués par phrase. Comme le montre la figure 3.9, pour cette phrase spécifique, l'échelle psychométrique indique l'ancre comme méthode préférée, et l'explication humaine comme moins appréciée par les utilisateurs.

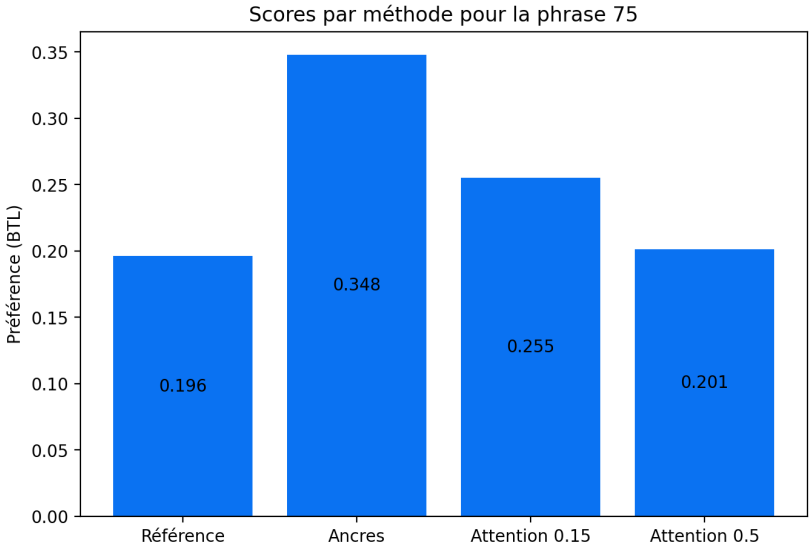


FIGURE 3.9 – Exemple des scores obtenus avec le modèle BTL pour chaque méthode d'explication sur une phrase du jeu de données LEGO - DE. La valeur correspond à la probabilité qu'un utilisateur préfère une méthode plutôt que les autres. L'explication humaine correspond à la colonne "Référence".

Nous observons ensuite les scores BTL sur l'ensemble du jeu de données LEGO DE, et sur deux sous-ensembles : les vrais positifs et les faux positifs, car nous nous attendons à ce que les résultats diffèrent. Les résultats sont comparés dans la figure 3.10. Cette analyse ne montre aucune différence significative entre les méthodes, autant sur la base complète

que sur les deux sous-ensembles. Un tel résultat souligne que, sur le plan global, aucune méthode ne l'emporte en termes de préférence utilisateur.

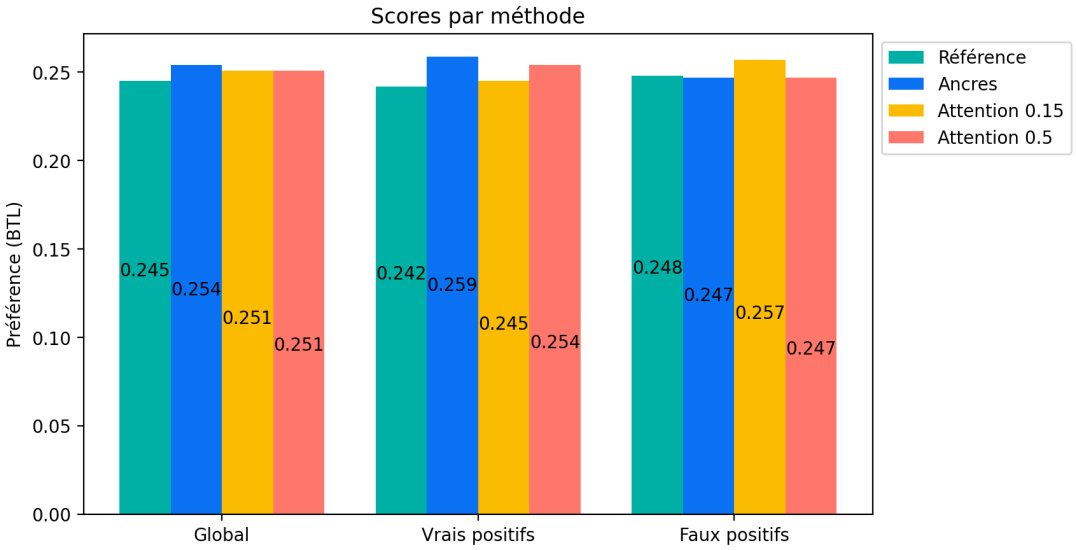


FIGURE 3.10 – Scores obtenus avec le modèle BTL pour chaque méthode d'explication, sur le jeu de données LEGO DE. Les vrais positifs et les faux positifs sont indiqués. Aucune différence significative n'apparaît entre les méthodes.

Comme nous l'avons fait dans la section 3.3.2, nous allons diviser notre ensemble de données par phrases avec des explications de référence *vide*, *simple* et *complexe*. L'algorithme BTL nous donne l'échelle psychométrique associée dans la Figure 3.11.

Encore une fois, cette échelle psychométrique peut être transformée en podiums, affichés dans le tableau 3.9. Faisons la première analyse pour tous les types de phrases. Pour les phrases avec une explication de référence vide, les utilisateurs n'ont pas une forte préférence pour une méthode ou une autre, comme le montre la Figure 3.11. Le podium dans le tableau 3.9 indique que la légère préférence va aux méthodes d'attention. Pour les cas simples, les méthodes d'attention interne sont préférées, et l'explication humaine est la moins appréciée. Pour les cas plus complexes, la méthode d'explication des ancres a été préférée, et les méthodes d'attention interne sont en difficulté. Dans notre cas d'usage, en d'autres termes, les utilisateurs préfèrent les méthodes d'attention dans les cas simples. Lorsque les choses deviennent plus complexes, leur préférence va à la méthode d'explication des ancres.

Nous avons calculé les préférences de chaque méthode d'explication et établi des podiums. L'analyse des résultats montre que différentes catégories de phrases existent et

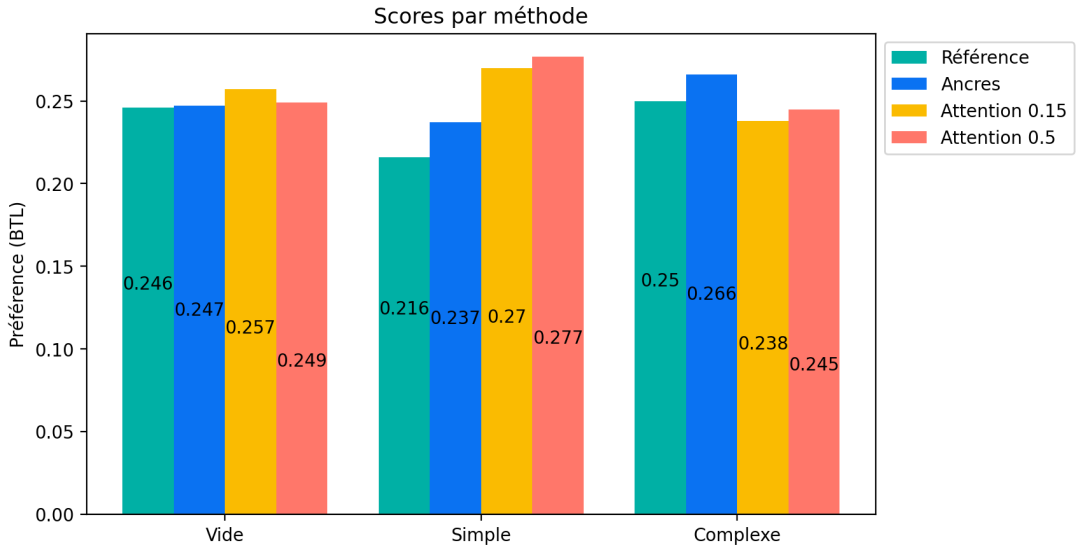


FIGURE 3.11 – Scores obtenus avec le modèle BTL pour chaque méthode d'explication, sur le jeu de données LEGO DE, par complexité. Ce regroupement montre des différences dans les préférences des utilisateurs.

TABLE 3.9 – Podium basé sur une étude psychométrique des utilisateurs, sur le jeu de données LEGO DE, par catégorie d'explication.

	Premier	**Second**	**Troisième**	
Vide	Attention 0,15	(Attention 0,5 , Ancre , Référence)		
Simple	Attention 0,5	Attention 0,15	Ancre	Référence
Complexe	Ancre	Référence	Attention 0,5	Attention 0,15

amènent des différences dans les préférences utilisateurs. Les podiums ainsi obtenus par catégorie sont comparés aux podiums de la métrique qualitative dans la section suivante.

3.3.4 Résultats

Nous pouvons maintenant comparer les résultats de l'analyse quantitative de la section 3.2.2 et les préférences des utilisateurs. Comme nous l'avons fait précédemment, nous regroupons les phrases par la complexité des explications de référence selon trois groupes : explications *vides*, *simples*, et *complexes*. Nous souhaitons désormais savoir si les méthodes d'explication qui performent le mieux sont également celles préférées par les humains.

Les podiums du Tableau 3.8 et du Tableau 3.9 présentés dans les sections précédentes diffèrent fortement. En effet, la métrique de similarité est basée sur les explications humaines en tant que référence, qui n'est pas la méthode préférée des utilisateurs. Le diagramme à barres empilées tel qu'indiqué dans la Figure 3.12 permet de comparer la proportion relative des méthodes dans chaque cas. Les valeurs de similarité par IOU sont normalisées pour la comparaison, afin que la somme de tous les scores soit égale à un. Les valeurs réelles sont présentées dans le tableau 3.7.

Pour les phrases de catégorie *vides*, la figure 3.12 montre que l'IOU et BTL reflètent des explications différentes. Pour rappel, le jeu de données DE est composé de phrases étiquetées comme non conformes par notre modèle. Les explications de référence vides sont associées à des phrases conformes, qui ont été prédites à tort comme non conformes par le modèle. Dans ce cas, les explications humaines telles que nous les avons définies peuvent ne pas être la bonne référence pour calculer les IOU. Comme expliqué dans la section 2.1.2, les experts du domaine pourraient préférer une explication attendue, qui devrait être définie pour un modèle spécifique.

Le cas des phrases associées à des explications humaines *simples* est présenté dans la figure 3.13. Elle met en évidence de bons résultats quantitatifs pour l'*attention 0,15* et les ancres, l'*attention 0,5* étant la méthode qui correspond le moins à l'explication humaine. Les utilisateurs n'ont pas préféré l'explication humaine aux autres méthodes et ont préféré l'*attention 0,5*. Prendre l'explication humaine comme référence n'a donc pas de sens, et nous pouvons en conclure que l'IOU basée sur l'explication humaine n'est pas efficace ici non plus. Cependant, les utilisateurs semblent apprécier les explications courtes, puisqu'iels ont préféré *attention 0,5* à *attention 0,15*. Notre métrique quantitative devrait refléter ce point, en diminuant lorsque la longueur d'une explication augmente.

Les podiums pour les phrases avec des explications de référence plus complexes sont

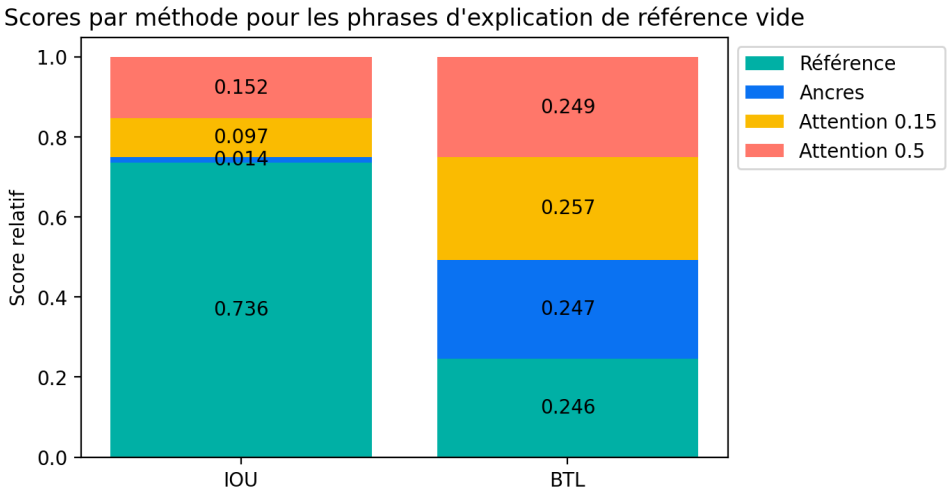


FIGURE 3.12 – Scores relatifs de similarité (IOU) et de préférence (BTL) normalisés pour chaque méthode d'explication, sur le jeu de données LEGO DE, filtré sur les phrases avec des explications vides.

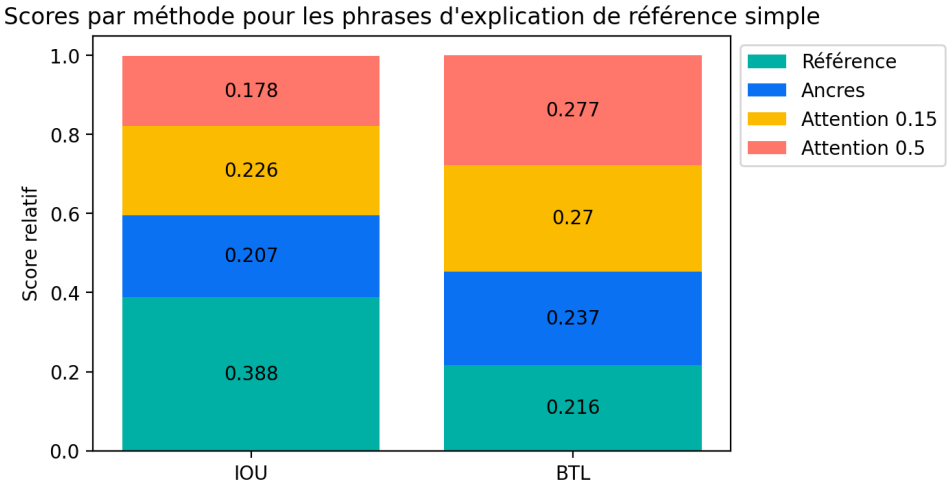


FIGURE 3.13 – Scores relatifs de similarité (IOU) et de préférence (BTL) normalisés pour chaque méthode d'explication, sur le jeu de données LEGO - DE, filtré sur les phrases avec des explications simples. Les explications de référence sont d'IOU égale à 1 avant normalisation.

comparés dans la figure 3.14. L'IOU montre une bonne correspondance pour les explications des ancres. Cela se reflète dans l'étude des utilisateurs, puisque la méthode préférée est celle des ancres, suivie de la référence. En considérant l'IOU, l'*attention 0.5* est moins performante comparée à *attention 0.15*. Cependant la préférence des utilisateurs est inverse. Comme l'*attention 0,5* est un sous-ensemble de *attention 0,15*, cela indique une préférence pour les explications plus courtes. Tout comme pour les cas simples, notre métrique refléterait mieux les préférences des utilisateurs en tenant compte de la longueur des explications.

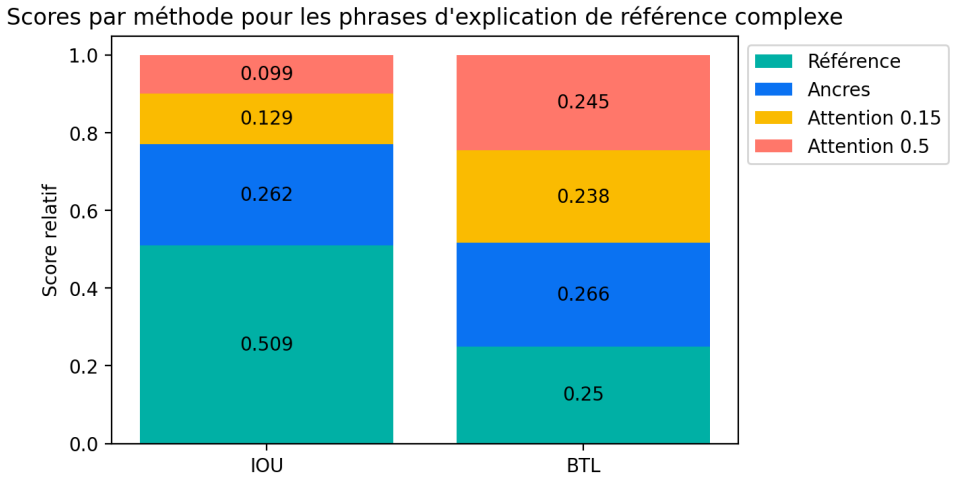


FIGURE 3.14 – Scores relatifs de similarité (IOU) et de préférence (BTL) normalisés par méthode d'explication, sur le jeu de données LEGO DE, filtré sur les phrases avec des explications complexes.

Pour toutes ces catégories, nos podiums sont différents. Selon notre protocole, cela signifie que notre mesure quantitative ne reflète pas la préférence humaine. Pour ajuster cette mesure, l'analyse des résultats donne deux pistes.

— Les explications attendues sont plus pertinentes que les explications idéales (Cf. section 2.1.2) pour mesurer la préférence des utilisateurs.

— Prendre en compte la préférence des utilisateurs pour les explications courtes.

Nous répondons à notre seconde question : "comment faire pour que la métrique quantitative corresponde aux préférences des utilisateurs ?" par une nouvelle mesure de performance.

$$performance = \frac{e_{generee} \cap e}{e_{attendue} \cup e} \frac{\alpha}{len(e)} \tag{3.2}$$

3.4 Conclusion

Nous avons collecté en section 3.1 les retours des utilisateurs dans la conception des systèmes d'explications. Nous avons ensuite présenté les comparaisons de méthodes de génération d'explications, dans deux contextes : sans utilisateurs en section 3.2 et avec en section 3.3. Les protocoles ont été conçus pour être généralisés à tout cas d'usage de classification multi-classe. Ils sont applicables à toute méthode d'explication par variable d'importance.

Le test d'utilisabilité a donné l'idée d'une interface non envisagée avant : l'explication par la règle. Il a démontré son intérêt tout en nécessitant peu de temps de réalisation.

La comparaison sans utilisateurs donne des résultats différents selon le cas d'usage. Pour le second protocole, les résultats montrent que la méthode préférée dépend de la taille de l'explication de référence. Les explications de référence ne sont pas représentatives des préférences des utilisateurs. Les utilisateurs préfèrent des explications courtes.

Cette mise en pratique met en lumière la dépendance de ces méthodes d'évaluation à la disponibilité et la qualité d'explications de référence. Sans ces explications les évaluations sont restreintes et nécessitent une analyse manuelle qui, bien que rendue la plus efficiente possible, reste lourde. Les faux positifs sont une source de différence entre les explications attendues et les explications idéales, et cibler ceux-ci peut réduire le coût de passer d'explications idéales à attendues.

Nous avons évalué le format et les méthodes de génération d'explications locales. Dans le chapitre suivant, nous nous attelons à générer des explications globales sur le comportement d'un modèle.

Résumé

- ✓ Nous avons collecté les besoins des experts du domaine
- ✓ Les systèmes d'explications les plus complexes ne mènent pas aux interfaces les plus appréciées des utilisateurs
- ✓ L'évaluation des explications sans avis humain permet une première mesure de d'adéquation
- ✓ L'évaluation subjective permet de tenir compte les préférences des experts dans la mesure de performance
- ✓ Le protocole d'évaluation a mené aux publications [53, 54, 55]

Chapitre 4

Caractérisation d'un modèle d'Intelligence Artificielle

Dans ce chapitre

Ce chapitre propose une méthode de caractérisation globale d'un modèle. L'objectif est de construire une représentation mentale du modèle. Un extrait pertinent de paires d'exemples de contre-exemples proches. La stratégie puis l'implémentation de la méthode sont présentées. L'application met en avant les avantages et limites de la proposition. Cette méthode de caractérisation est prometteuse, mais reste à comparer à d'autres méthodes de la littérature.

Nous avons présenté dans les chapitres précédents des explications locales. Dans ce chapitre, nous caractérisons globalement un modèle d'IA. Ce chapitre répond à des besoins d'auditabilité et des contraintes légales. Notre objectif est de permettre aux utilisateurs de créer un modèle mental fidèle au modèle caractérisé. La notion de *modèle mental* est introduite en section 1.2. Nous proposons une méthodologie générale de caractérisation, qui s'appuie sur les limites de décision par classe d'un modèle d'IA

Nous présentons notre protocole et l'appliquons à un cas d'utilisation réel dans le domaine du traitement du langage naturel. L'application montre son potentiel sur l'une des classes d'un modèle de classification. L'analyse de l'explication permet d'appréhender des éléments clés de la frontière de décision, et d'émettre des hypothèses d'amélioration du jeu de données d'entraînement. Nous notons également des marges d'amélioration. Ainsi l'ajout d'une méthode d'explication locale est nécessaire pour l'analyse des textes longs.

Nous présentons en section 4.1 la méthode de caractérisation, et son implémentation en section 4.2. Enfin, nous appliquons cette caractérisation en section 4.3.

4.1 Stratégie

Nous nous intéressons au contexte de la classification multi-label, multi-classes. Dans le chapitre 1, nous présentons Lime, méthode d'explication locale, et son explication globale associée : SP-Lime [90]. Cette explication globale s'appuie sur des exemples dont l'explication Lime a été réalisée au préalable. Elle est donc coûteuse. Toutefois, c'est cette philosophie qui conduit la suite des travaux présentés dans ce chapitre.

Comment permettre à l'utilisateur d'avoir un modèle mental fidèle au comportement réel du modèle original (MO) ? Pour résoudre cette problématique, nous proposons d'aider à l'appréhension des frontières de décision du MO. Cette aide se traduit par l'analyse d'un ensemble limité d'exemples et de contre-exemples pour une classe donnée, comme illustré avec le tableau 4.1. Un exemple, dans la première colonne du tableau, correspond ici à la classe étudiée et un contre-exemple, dans la seconde colonne, est un élément d'une autre classe. L'ensemble des couples d'exemple et contre-exemple permet d'appréhender différents aspects d'une classe, notamment ses frontières avec d'autres classes. Les exemples situés au cœur d'une classe nous semblent moins pertinents.

TABLE 4.1 – Délimitation de la classe *"Discrimination : Contrat étudiant"* sous la forme d'exemples et contre-exemples associés

Exemple type	**Contre-exemple associé**
Profil recherché : Etudiant(e), salarié(e), retraité(e).	Etudiant(e), retraité(e), travaillant à temps partiel ou en recherche d'emploi, vous êtes avant tout passionné(e) par les enfants ?
Etudiants acceptés.	35 H/Semaine minimum Etudiants acceptés.

Un utilisateur ne peut recevoir un nombre trop important d'éléments à traiter à la fois. Ces exemples doivent donc être habilement sélectionnés. Illustrée en figure 4.1, notre stratégie consiste à prendre un ensemble d'exemples candidats, puis de filtrer et trier les exemples qui en sont issus. Le filtre vise à conserver uniquement les exemples qui ont une pertinence technique pour représenter le comportement du MO. Il est possible que le filtre ne suffise pas pour montrer un ensemble restreint d'une dizaine d'éléments à un utilisateur ou une utilisatrice. Un tri est alors effectué avec pour but de proposer un résumé digeste et représentatif pour l'utilisateur. Ce tri peut être laissé à la main de l'utilisateur.

La figure 4.1 présente l'enchainement de définition d'exemples candidats, filtre, et tri de ces exemples. Les détails sont volontairement mis de côté, nous proposons une méthodologie générale dont les implémentations des différents éléments sont à adapter.

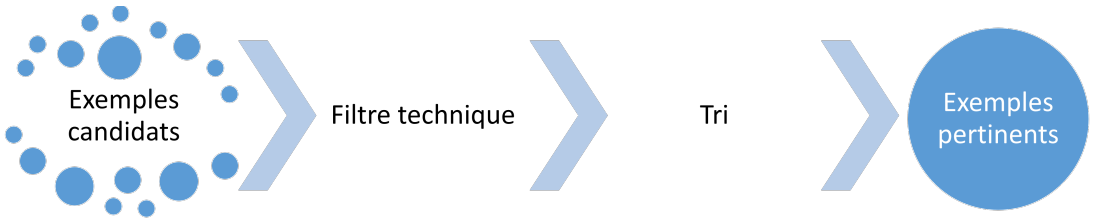


FIGURE 4.1 – La stratégie de sélection des exemples à présenter à l'utilisateur.

Maintenant que nous avons défini ce que nous souhaitons présenter aux utilisateurs, et la stratégie générale, nous pouvons définir l'implémentation de cette stratégie.

4.2 Implémentation

Dans cette section, nous présentons cette implémentation ainsi que nos choix techniques. Tous les éléments présentés ici sont des choix d'implémentation qui ne sont pas fixés. Il est possible de modifier les trois étages présentés ici, en conservant la structure présentée en section précédente. Nous proposons cette première solution fonctionnelle, appliquée sur un cas d'usage en section 4.3. Celle-ci reste à évaluer, et adapter à d'autres situations et contraintes.

Nous présenterons en premier lieu la sélection d'un ensemble d'exemples candidats en section 4.2.1. En section 4.2.2 nous détaillerons le filtre technique effectué, et en section 4.2.3 le tri appliqué.

4.2.1 Choix de la source d'exemples candidats

Les exemples présentés aux utilisateurs dans la littérature peuvent être issus des données réelles, comme dans [90]. Les candidats sont alors les données d'entraînement, de développement et de test. Ces exemples peuvent également être générés. Dans [18], les auteurs créent des explications sous la forme de la différence entre des exemples et contre-exemples générés. Les techniques de génération actuelles permettent de créer des données complexes telles que des textes [113] et images [88].

Générer les exemples et contre-exemples nécessite un apprentissage supplémentaire du modèle générateur, et crée des données non connues par le MO. L'entraînement d'un modèle générateur ajoute une technique d'apprentissage profond. Cette méthode a pour risque de reporter le problème d'explication à ce nouveau modèle générateur. Afin de ré-

duire les efforts des utilisateurs pour reconstruire leur modèle mental, nous nous appuyons sur des exemples candidats concrets. Nous préférons nous appuyer sur les données issues de la base d'apprentissage, permettant ainsi de ne pas s'appuyer sur une couche supplémentaire d'apprentissage profond. Nous nous assurons aussi d'avoir uniquement des données réelles, nécessitant en contrepartie d'avoir accès au jeu de données d'entraînement.

Toutes les données d'entraînement n'ont pas le même impact sur le comportement du modèle. Comme défini en section 4.1, nous souhaitons conserver, parmi les exemples candidats, uniquement ceux qui ont une pertinence du point de vue du MO.

4.2.2 Filtre

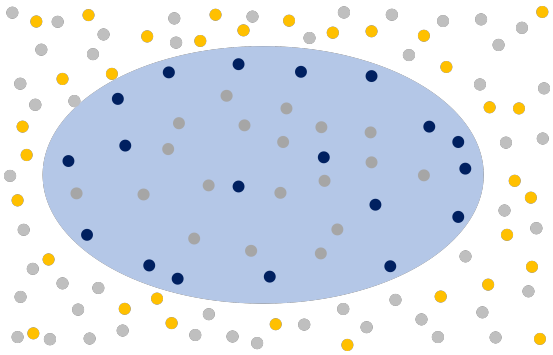
Ces exemples pertinents sont ceux qui représentent le fonctionnement du MO. Nous cherchons notamment les données présentes aux frontières de décision, qui pourraient être récupérées en filtrant les données d'apprentissage qui en sont proches.

Pour déterminer ces points, différentes approches existent. Nous pouvons par exemple nous appuyer sur le fonctionnement d'un réseau de neurones. Les points dont le score de prédiction (sortie du MO) est proche de 0,5 pour une classe donnée correspondent à des points qui ne sont pas typiques de cette classe. Cependant, il est plus difficile de dire si ces points sont situés près d'une frontière ou s'ils ont un score faible car ils sont hors de la distribution des données ou aberrants.

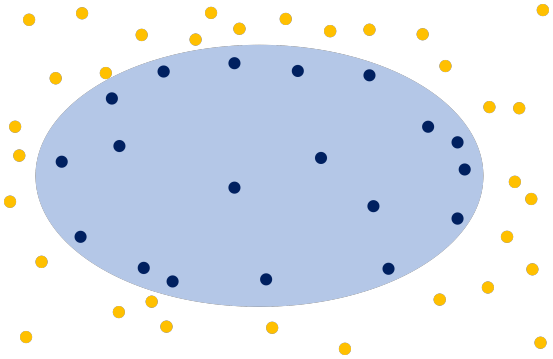
Il est également possible de prendre des points antagonistes (de classes différentes) proches d'une manière centrée sur les données. En définissant une fonction de distance entre les points, nous pouvons sélectionner le couple de points les plus proches qui appartiennent à deux décisions de modèle différentes comme antagonistes. Mais ces points peuvent ne pas être les plus représentatifs de la frontière de décision. Dans un contexte de classification One *vs.* Rest, plusieurs couples d'instances de données peuvent être nécessaires pour bien définir la frontière de décision. Dans un mode One *vs.* One, si deux classes sont éloignées dans l'espace de représentation, le couple d'instances de données résultant n'aura aucun intérêt. Certaines frontières complexes nécessitent également la définition de plus d'un couple d'exemples.

Nous proposons un filtre basé sur les SVM pour extraire les exemples factuels et contrefactuels pertinents. Le filtre s'appuie sur la notion de vecteurs supports du SVM, soit les points de données importantes pour l'apprentissage du SVM. Il est composé de 3 étapes : suppression des doublons, sélection des vecteurs supports et appairage.

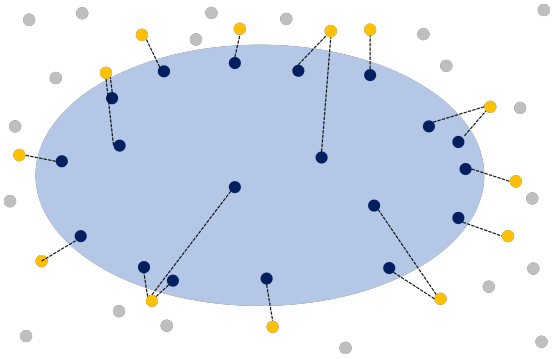
Les deux dernières étapes sont illustrées dans la figure 4.2. La suppression des doublons



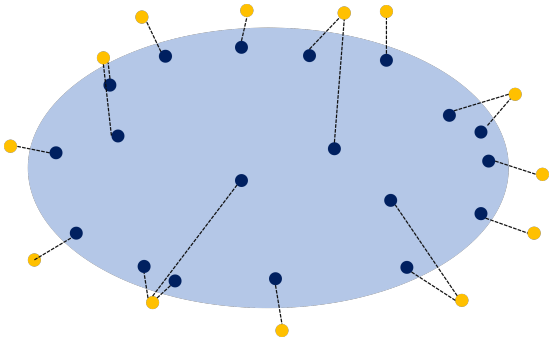
(a) Les données candidates. Les points bleus et jaunes correspondent aux vecteurs supports protagonistes et antagonistes. Les autres points de données sont en gris



(b) Les points qui ne sont pas des vecteurs supports sont supprimés.



(c) Chaque vecteur support protagoniste est apparié au vecteur support antagoniste le plus proche. Les vecteurs supports antagonistes non appariés sont grisés.



(d) Seuls les vecteurs supports appariés sont conservés.

FIGURE 4.2 – Illustration des principes des différentes étapes de filtre. L'ellipse bleue correspond aux éléments d'une classe observée. Les points bleus appartiennent à la classe considérée, ce sont des éléments protagonistes. Les points jaunes appartiennent aux autres classes, ils sont antagonistes. Les points gris sont ceux supprimés à chaque étape.

consiste à ne conserver qu'un seul point au cas où plusieurs seraient au même endroit, et n'est donc pas visible sur cette illustration. Dans le contexte One *vs.* Rest, les points de la classe étudiée sont situés à l'intérieur de l'ellipse bleue. Les points en dehors de cette ellipse peuvent être de toutes les autres classes. Les sous figures 4.2a et 4.2b montrent le filtre sur les vecteurs supports. Les points grisés en figure 4.2a sont les points candidats qui ne sont pas des vecteurs supports. La figure 4.2b présente le même espace avec ces points de données en gris supprimés. La figure 4.2c montre l'appairage entre chaque vecteur support protagoniste et le vecteur support antagoniste le plus proche. Un vecteur support antagoniste peut ainsi être apparié à plusieurs vecteurs supports protagonistes. Cette étape permet de conserver *au maximum* $2 * P$ vecteurs supports, avec P le nombre de vecteurs supports protagonistes. Pour appliquer ce filtre, il faut dans un premier temps entraîner un SVM.

Entraînement du SVM Avec le SVM entraîné, nous nous concentrons sur les vecteurs supports. Les vecteurs supports sont des points de données intéressants utilisés par le SVM pour apprendre sa fonction de décision.

La figure 4.3 compare les architectures du MO, sur la première ligne et du SVM sur la seconde. Pour un MO donné, l'architecture des premières couches est conservée, et devient la vectorisation en seconde ligne. La classification est effectuée par un SVM à noyau gaussien. Pour que le SVM copie le MO, il doit être entraîné sur les décisions de classification du MO. L'entrée du MO correspond aux données réelles (X, Y), X étant les données vectorisées et Y étant la classe. Il produit en sortie une décision $\bar{Y}$. L'entraînement du SVM prend la même entrée X, mais la cible est $(\bar{Y})$: il est entraîné dans les résultats de classification du MO. Le jeu de données d'entraînement est le même ou à défaut un sous-ensemble.

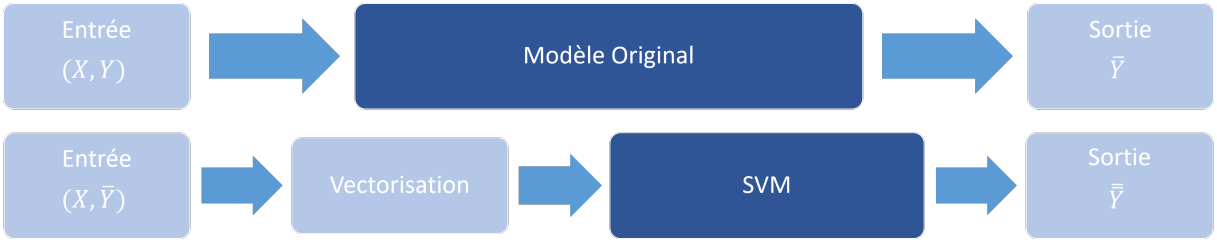


FIGURE 4.3 – Comparaison entre le modèle boîte noire et les architectures SVM. L'entrée du SVM est basée sur la sortie du modèle original.

Les vecteurs supports du SVM ainsi déterminés sont appariés par classe, un vecteur

positif avec son vecteur négatif le plus proche. Cette liaison donne des couples pertinents d'exemples factuels et contre-exemples pour une classe donnée.

Les vecteurs support Le SVM est composé de $n-1$ classifieurs One *vs.* Rest. Chacun de ces classifieurs est une fonction de décision, basée sur des vecteurs supports SV. Son signe de sortie correspond à une classe (positif) ou au reste (négatif). La fonction de décision pour une instance x est :

$$\sum_{i \in SV} y_i \alpha_i K(x_i, x) + b \tag{4.1}$$

Avec K la fonction noyau gaussienne :

$$K(x_i, x) = e^{-\gamma ||x_i - x||} \tag{4.2}$$

Les coefficients duaux des vecteurs supports $y_i \alpha_i$ sont définis avec leur signe $y \in \{-1, 1\}$ et leur valeur $\alpha \geq 0$. Ils peuvent être positifs (protagonistes) : pour une classe, le label du vecteur support est le même. Ils peuvent aussi être négatifs : leur étiquette diffère de la classe donnée.

Un classifieur peut, selon le cas d'utilisation, avoir de nombreux vecteurs supports, en particulier avec des coefficients y négatifs. En associant à chaque vecteur positif le vecteur négatif le plus proche, nous éliminons les vecteurs négatifs non associés. Dans le cadre de l'analyse de textes, nous utilisons la distance cosinus.

Pour fournir une vision claire à l'utilisateur, il est nécessaire de trier les points les plus pertinents parmi les vecteurs supports, qui sont déjà les points les plus pertinents du jeu de données d'apprentissage.

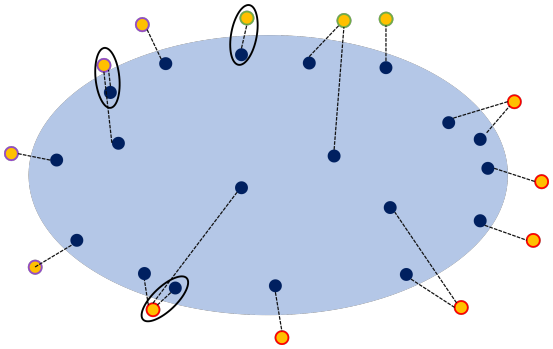
4.2.3 Tri

Le tri proposé permet de fournir un résumé digeste pour un humain. L'objectif est donc de ressortir un ensemble de moins d'une dizaine de couples d'exemples. Il est possible de laisser la main à l'utilisateur ou utilisatrice, afin de lui laisser choisir le niveau de zoom qu'iel souhaite.

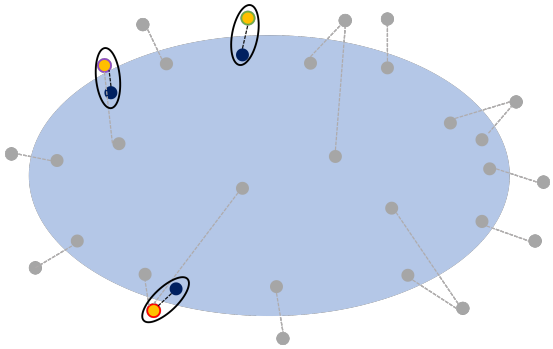
Encore une fois, de nombreuses techniques sont à notre disposition. Il est possible de se baser sur le SVM et trier les couples de vecteurs par somme de leurs valeurs alpha. Il est également possible de prendre un point de la classe étudiée (exemple) au hasard, et effec-

tuer une approximation polygonale en passant par une heuristique gloutonne de sélection du point le plus éloigné du point précédent, ou du barycentre des points sélectionnés.

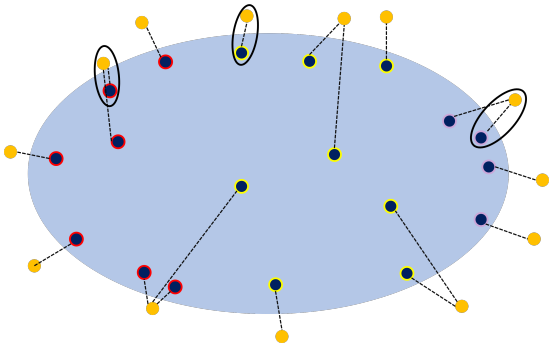
Nous proposons de regrouper les exemples négatifs, en un nombre réduit de groupes. Nous discuterons dans la suite d'une méthode permettant de calculer de tels groupes. Pour chaque groupe constitué, un seul élément est conservé. Ce mécanisme permet d'assurer une bonne répartition des exemples et contre-exemples sélectionnés.



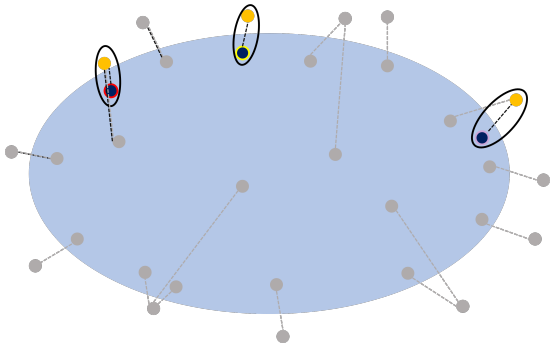
(a) Regroupement des vecteurs supports antagonistes en 3 groupes par une méthode quelconque : violet, vert et rouge. Pour chaque groupe, la paire de distance minimale est entourée.



(b) Seules les paires entourées sont conservées, les autres points sont mis de côté.



(c) Regroupement des vecteurs supports protagonistes en 3 groupes par une méthode quelconque : violet, jaune et rouge. Pour chaque groupe, la paire de distance minimale est entourée.



(d) Seules les paires entourées sont conservées, les autres points sont mis de côté.

FIGURE 4.4 – Illustration des différentes étapes de tri.

La figure 4.4 présente le filtre avec la sélection de $n = 3$ paires d'exemples et contre-exemples. Les figures 4.4a et 4.4b présentent le premier mode du filtre en effectuant les regroupements sur les vecteurs supports antagonistes. Les figures 4.4c et 4.4d présentent

le second mode du filtre en effectuant les regroupements sur les vecteurs supports protagonistes. Pour chaque mode, le principe est identique. Les vecteurs supports (protagonistes ou antagonistes) sont regroupés en n groupes. Pour chaque groupe, le couple d'éléments de distance minimale est présenté, c'est à dire les instances les plus proches de la frontière. Tous les autres éléments sont mis de côté.

Il est également possible de concaténer les résultats des deux modes. La sélection des paires de distance intra paire minimale augmente la probabilité que les paires sélectionnées soient identiques entre les deux modes de regroupements. Si n paires sont demandées, le nombre de paires concaténées des deux modes sera alors compris entre n et $2n$.

Le regroupement est effectué par une Classification Ascendante Hiérarchique (CAH) basée sur le critère de Ward. Une heuristique classique afin de déterminer le nombre de groupes à conserver consiste à observer le dendrogramme présentant les regroupements et la distance interclasses. La meilleure découpe de ce dendrogramme est classiquement celle apportant le plus d'informations, c'est à dire celle où le saut est le plus important. Nous considérons la CAH comme un zoom, et laissons l'utilisateur couper où iel le souhaite. Les couples d'exemples ainsi sélectionnés sont présentés.

Cette méthode permet de présenter un ensemble restreint et représentatif des couples d'exemples et contre-exemples d'intérêt. Il revient alors à l'utilisateur d'effectuer l'analyse des éléments pour la classe étudiée. Ce travail est à faire pour toutes les classes afin d'obtenir une vision globale du modèle.

Maintenant que le protocole est défini, il est appliqué à un cas d'utilisation réel, ce qui nous permet de voir ses avantages, ses limites et ses possibilités d'amélioration.

4.3 Application

Nous appliquons cette stratégie de caractérisation au cas d'usage LEGO présenté en chapitre 3. Pour rappel, c'est un problème de classification multi-classes, mono-label, par phrase. Il y a 28 classes, l'une d'entre elles comprenant les phrases légales. Cette classe a un intérêt fonctionnel tout particulier : elle correspond à des phrases "correctes", là où tous les autres motifs sont des phrases "incorrectes". Deux des classes possèdent 5 éléments ou moins dans les données d'entrainement et sont donc mises de côté dans les parties suivantes. Ces classes sont *Discrimination : race, ethnie* et *Discrimination : convictions religieuses*.

Le modèle étudié pour cette application est le modèle à attention présenté en chapitre

TABLE 4.2 – Exemples factuels pertinents et leurs exemples contrefactuels pour la classe "Discrimination : contrat étudiant". La décision du MO associée au contrefactuel et la distance entre l'exemple factuel et le contrefactuel sont également affichées.

Exemple factuel	**Exemple contrefactuel associé (CFE)**	**Distance cosinus**
Profil recherché : Etudiant(e), salarié(e), retraité(e).	Etudiant(e), retraité(e), travaillant à temps partiel ou en recherche d'emploi, vous êtes avant tout passionné(e) par les enfants ? Devenez un(e) nounou Kangourou !	0, 066
Etudiants acceptés.	35 H/Semaine minimum Etudiants acceptés.	0, 050

3. Dans ce chapitre, le modèle à attention est considéré comme une boîte noire. Les données candidates de la section 4.2.1 sont les données d'entraînement présentées en chapitre 3. L'implémentation du filtre présentée précédemment est appliquée en section 4.3.1. Celle du tri est appliquée en section 4.3.2.

4.3.1 Filtre

Le SVM entraîné est un SVC de sci-kit learn, avec un noyau gaussien. L'ensemble d'apprentissage est le même que celui utilisé pour le MO, mais la classe à prédire est la décision du MO, comme spécifié dans 4.2.2. Il obtient une précision moyenne de 0, 997, signifiant qu'il parvient à imiter le MO.

Le SVM possède 14840 vecteurs support, correspondant à 3, 09% de l'ensemble de données d'apprentissage. Les vecteurs supports correspondent à 12077 vecteurs uniques, soit 2, 52% de l'ensemble de données d'apprentissage. Certaines phrases de l'ensemble d'apprentissage sont des doublons, d'où la légère variation. Dans notre analyse, nous avons élagué les doublons en sélectionnant le premier de tous les vecteurs supports correspondant à une phrase donnée.

Pour chaque classe, des vecteurs supports sont sélectionnés, divisés en vecteurs supports positifs et négatifs. Une distance cosinus est calculée entre les deux groupes. Comme nous cherchons à retrouver la proximité sémantique des phrases, nous utilisons la distance cosinus. Un vecteur support positif (resp. négatif) donne un exemple factuel (resp. contrefactuel) pertinent. Chaque exemple factuel pertinent de la classe étudiée est apparié avec l'exemple contrefactuel le plus proche.

La table 4.2 présente deux couples d'exemples factuels et contrefactuels pour la classe *"Discrimination : contrat étudiant"*. Il est illégal en France de spécifier que le candidat

TABLE 4.3 – Diminution des paires pour la classe *"Discrimination : Contrat étudiant"*

	Légal	**Contrat étudiant**	**Paires**
Données brutes	385754	2477	955 512 658
Suppression des doublons			
Vecteurs uniques	259 853	841	218 536 373
Sélection des vecteurs supports			
vecteurs supports	4 688	151	707 888
Appairage par similarité cosinus			
Paires de vecteurs supports	43	151	151

d'une offre d'emploi doit être un étudiant.

Pour cette classe, nous obtenons 151 couples de vecteurs supports positifs et négatifs. Le tableau 4.3 montre les réductions consécutives du nombre de points en partant du nombre de données brutes). La dernière colonne présente le nombre de paires possibles. Avant le filtre, le nombre de paires d'exemples et contre-exemples dans les données d'entraînement pour la classe *Discrimination : Contrat étudiant* est proche du milliard. Pour cette classe spécifiquement, la suppression des doublons divise le nombre de paires par 4 environ. La sélection des vecteurs supports divise par 309 ce nombre, et l'appairage final le divise par 4688. Cette réduction est notable sur toutes les classes étudiées, et est plus prononcées sur les classes fortement représentées, comme le montre le tableau 4.4.

Le tableau 4.4 présente le nombre de vecteurs supports protagonistes uniques, ainsi que le nombre de paires finales pour chaque classe du cas d'usage. Pour l'ensemble des classes correspondant à des motifs de rejet, l'étape de filtre permet d'obtenir entre 8 et 1154 paires d'exemples et contre-exemples. La classe de phrases légales en comporte 4688, ce qui est justifié par sa nature plus générale.

Afin de présenter un nombre digeste d'exemples et contre-exemples, il faut effectuer un tri supplémentaire, présenté dans la section suivante.

4.3.2 Tri

La CAH permet d'obtenir une représentation hiérarchique sous la forme d'un dendrogramme. La figure 4.5 présente le dendrogramme obtenu grâce à la CAH des 151 vecteurs supports protagonistes de la classe "*Discrimination : contrat étudiant*". Le dendrogramme est affiché pour $n = 10$ groupes, puisqu'il n'est pas souhaitable de montrer plus d'éléments aux utilisateurs et utilisatrices.

TABLE 4.4 – Nombre de vecteurs supports protagonistes uniques et de paires d'exemples et contre-exemples obtenus grâce au filtre, par classe.

Classe	**VS uniques**	**Paires**
Discrimination : Activité syndicale ou mutualiste	8	8
Discrimination : Age	2251	579
Discrimination : Apparence physique	590	311
Discrimination : Contrat aidé	15	14
Discrimination : Contrat étudiant	841	151
Discrimination : État de santé	5815	1006
Discrimination : Grossesse	43	30
Discrimination : Handicap	674	296
Discrimination : Mœurs	102	62
Discrimination : Nationalité	168	123
Discrimination : Opinions politiques	27	22
Discrimination : Origine	349	193
Discrimination : Résidence	1845	369
Discrimination : Genre - Au moins un accord au féminin	1247	415
Discrimination : Genre - Intitulé exclusivement féminin	2606	478
Discrimination : Genre - Mention genre exclusif	129	84
Discrimination : Situsation familiale	29	26
Discrimination : Taille	173	101
Droit du travail : Achat de matériel	6018	892
Droit du travail : CDD Possibilité CDI	12461	1154
Libertés : Casier Judiciaire	351	117
Libertés : Tenue vestimentaire	22	18
Légal	259853	4688
Offre non conforme : Stage	115	78
Terme innaproprié	2974	650
Texte spécifique : Gratuité	377	212

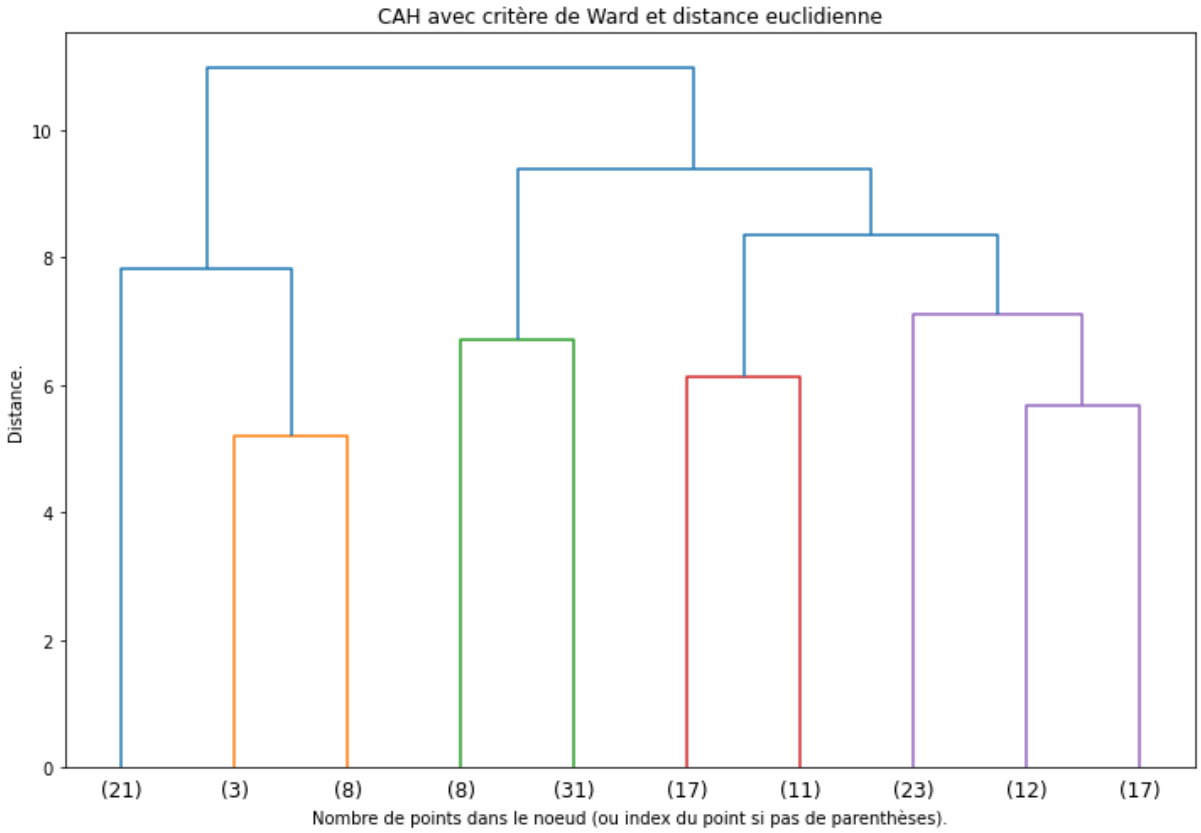


FIGURE 4.5 – Dendrogramme de la Classification Ascendante Hiérarchique des 151 vecteurs supports protagonistes de la classe "*Discrimination : contrat étudiant*", pour 10 groupes au plus bas du dendrogramme.

Sur les 151 points de données, la figure 4.5 montre que le regroupement en 10 groupes donne des ensembles plutôt équilibrés, à l'exception d'un groupe de 3 éléments et d'un groupe de 31 éléments. En nous basant sur la figure 4.5, nous considérons que le découpage en 5 groupes est un niveau de zoom intéressant, avec des groupes à la fois équilibrés et un saut de taille correcte ; notamment plus élevé que celui pour 4 groupes.

En considérant $n = 5$ groupes, et en appliquant le tri sur les paires via les vecteurs supports antagonistes, on obtient le tableau 4.5. Les 5 paires sont affichées, montrant des textes de taille variable. Les travaux du chapitre 3 3 ont été employés pour mettre en avant les mots de fort poids d'attention, c'est à dire un poids supérieur à un seuil $t = 0,15$. Les

TABLE 4.5 – Exemples factuels pertinents pour la classe "Discrimination : contrat étudiant" et leurs exemples contrefactuels de la classe "Légal", tri effectué sur les vecteurs supports antagonistes. Les mots soulignés sont ceux remontés par les explications par attention.

Paire	**Type**	**Texte**
1	Exemple	Votre profil : Vous detenez des experiences verifiables avec les enfants, aupres de structures ou de particuliers Votre mission, si vous l'acceptez : Vous vous occupez des enfants au domicile des parents Vous assurez le bien-etre des enfants jusqu'au retour des parents (maternage, change, soins, gouter, sortie d'ecole/creche) Vous organisez des activites d'eveil et jeux en fonction de l'age des enfants Il vous faudra etre disponible : Lundi Mardi Jeudi et Vendredi de 17h30 a 19h et le mercredi de 9h a 12h Avantage : Vous pouvez cumuler plusieurs missions Un job proche de votre domicile Des horaires compatibles avec votre emploi du temps vous permettant ainsi de cumuler deux emplois ou un job etudiant avec vos etudes.

	Contre - exemple	Votre mission, si vous l&rsquo ;acceptez : Vous vous occupez des enfants au domicile des parents Vous assurez le bien-etre des enfants jusqu&rsquo ;au retour des parents (maternage, change, soins, gouter, sortie d&rsquo ;ecole/creche) Vous organisez des activites d&rsquo ;eveil et jeux en fonction de l&rsquo ;age des enfants Il vous faudra etre disponible : En moyenne 2 fois par semaine de 6h15 a 13h30 ou de 16h a 20h30 (un planning mensuel vous sera fourni par la famille) Plusieurs disponibles en complement, a proximite Agence : Metz Categorie : Garde d'enfants de moins de 3 ans Ville :Delme Type de poste :Temps partiel Periode :Annee scolaire 2019-2020Competences :Debutant(e) accepte(e)
2	Exemple	Recherche baby-sitter à domicile à CHATOU pour 7,8 heures de travail par semaine pour baby-sitter 1 enfant, 7 ans.>br>Tâches confiées : garde d'enfants/baby-sitting, goûter, aide à la toilette, suivi des devoirs, préparation et prise des repas, accompagnement dans les déplacements.>br>Rémunération : 10,50 € brut/heure.>br>Horaire du baby-sitting : Du 05/09/18 au 17/10/18 puis du 07/11/18 au 19/12/18 puis du 09/01/19 au 20/02/19 puis du 13/03/19 au 17/04/19 puis du 08/05/19 au 29/05/19 puis du 05/06/19 au 03/07/19 : Le mercredi de 08h30 à 18h00.
	Contre - exemple	Recherche baby-sitter à domicile à MARSEILLE pour 0,9 heures de travail par semaine pour baby-sitter 2 enfants, 8 ans, 11 ans.>br>Tâches confiées : garde d'enfants/baby-sitting, sortie d'école.>br>Rémunération : 9,88 € brut/heure.>br>Horaire du baby-sitting : Du 04/09/18 au 04/10/18 puis du 27/11/18 au 29/11/18 puis du 22/01/19 au 24/01/19 puis du 19/03/19 au 21/03/19 puis du 14/05/19 au 13/06/19 :>br>1 semaine sur 4 sera travaillée.
3	Exemple	35 H/Semaine Etudiants acceptés.
	Contre - exemple	35 H/Semaine minimum Etudiants acceptés.
4	Exemple	Etudiant(e), à la retraite ou en activité à temps partiel, nous serons enchantés de vous compter parmi notre équipe, à bientôt Dans le cadre de cette mission, le véhicule est exigé pour le transport des enfants.

	Contre - exemple	Etudiant(e) de niveau Bac+4, vous préparez un diplôme en sécurité des systèmes informatiques / gestion de projet.
5	Exemple	Lieu de travail : Montpellier et alentours (Castelnau, Jacou, Clapiers, Montferrier, Teyran, Saint Clément de Rivière, Saint Gély du Fesc, Le Crès.)Profil recherché :v Débutant acceptév Ponctualité, Sérieux et Esprit d'initiativev Autonomie et Capacité d'adaptationv Sens de la relation clientèle, Sociabilitév Permis B indispensable Si vous êtes Etudiant, merci de vous assurer de la régularité de vos disponibilités sur une période de longue durée (au moins 8 mois).
	Contre - exemple	TERRE &amp ; MER INTÉRIM recherche, pour un de ses clients, 8 OPÉRATEURS NETTOYAGE INDUSTRIEL (H/F) sur Port de commerce à BRESTVos missions : - Nettoyage / Dégazage de cuves de Bateaux- Manutention- Une réunion / formation sécurité sera assurée par le client Durée de la mission : 1 semaine TERRE &amp ; MER INTÉRIM recrute le profil suivant :- Carte d'identité obligatoire- Débutant / Etudiant accepté,Autonome, motivé(e), organisé(e), courage

Le tableau 4.6 présente le même tri effectué sur les vecteurs supports protagonistes. une partie des paires ont déjà été remontées par le tableau 4.5 à savoir les paires 1, 3 et 4 et ne sont donc pas réaffichées. Comme dans le tableau précédent, les mots de poids d'attention élevé sont mis en gras.

Le premier exemple, soit la paire 1 dans le tableau 4.5, présente une offre longue, rejetée principalement par la présence du terme "école". Le contre-exemple présente une séquence de mots fortement similaire, mais avec un problème d'encodage du texte, présentant la séquence de texte *"&rsquo ;"* juste avant le mot école. Une analyse plus approfondie de cette paire permet de constater un fort poids d'attention pour le mot école dans l'exemple : $0,98$. Dans le contre-exemple, le mot école a un poids d'attention de $0,33$. Toutefois, en supprimant la séquence *"&rsquo ;"*, le poids d'attention du mot école augmente fortement et passe à $0,98$. Cet exemple met en avant la problématique d'une captation de textes de qualité variable, avec la présence de séquence de caractères qui bruitent les phrases. Ici, *"&rsquo ;"* correspond à une apostrophe, en langage XML ou HTML.

La seconde paire d'exemples du tableau 4.5 montre un cas typique de faux positif, avec un mot peu connu : la ville de *"Chatou"*. En contre-exemple, une offre similaire pour la ville de Marseille, plus grande donc à priori plus connue, ne pose pas de problème.

TABLE 4.6 – Exemples factuels pertinents pour la classe "Discrimination : contrat étudiant" et leurs exemples contrefactuels de la classe "Légal", tri effectué sur les vecteurs supports protagonistes. Les mots en gras sont les mots remontés par les explications par attention. Les paires 1, 3 et 4 tu tableau 4.5 sont également remontées, et non affichées ici dans un souci de clarté.

Paire	**Type**	**Texte**
6	Exemple	Vous gererez en autonomie le trajet creche/domicile, realisez des **activites** diverses (activite manuelles, lectures &hellip ;), suivi de **devoirs** pour le plus grand, temps calme jusqu&rsquo ;au retour des parents Experience exigee d&rsquo ;1 an minimum en garde d&rsquo ;enfants chez des particuliers (sortie d&rsquo ;ecole, aides aux devoirs, babysitting) ou en structure collective (centre de loisirs, creche&hellip ;) Formation appreciee dans le secteur (CAP, BEP carrieres sanitaires et sociales, Bac Pro ASSP, BAFA&hellip ;) Horaire d&rsquo ;intervention : Lundi/Mar/Jeu./Ven : 16h30 &ndash ; 18h30 et Mercredi 13h30-18h30 ou 11h30 &ndash ; 16h Remuneration : 9,88 a 10,10 &euro ;/ H (variable selon experience et formation) en CDI a pourvoir des la rentree 2018 Vous etes etudiant(e)s, s ou salarie(e)s a temps partiel ? Devenez Babychou-sitter(trice) en envoyant votre CV !
	Contre - exemple	Vous gererez en autonomie le trajet ecole/domicile, realisez des activites diverses (jeux de societe, lectures, sorties au parc&hellip ;), et parfois les repas et le bain Experience exigee d&rsquo ;1 an minimum en garde d&rsquo ;enfants chez des particuliers (sortie d&rsquo ;ecole, aides aux devoirs, babysitting) ou en structure collective (centre de loisirs, creche&hellip ;) Formation obligatoire dans le secteur (CAP, BEP carrieres sanitaires et sociales, Bac Pro ASSP, BAFA&hellip ;) Horaire d&rsquo ;intervention : Semaine 1 Lundi/Mardi/Jeudi/vendredi 16h30-19h + Mercredi 9h -19h + Semaine 2 Jeudi/vendredi 16h30 &ndash ; 19h Remuneration : de 10.10 a 10.88 &euro ;/ H (variable selon experience et formation) en CDI a pourvoir des la rentree 2018 Vous etes etudiant(e)s, s ou salarie(e)s a temps partiel ? Devenez Babychou-sitter(trice) en envoyant votre CV !
7	Exemple	Recherche : Opérateur de production - contrat **Etudiant** (H/F)PARTNAIRE recrute pour son client MAITRE COQ qui est une filiale du groupe agroalimentaire LDC, groupe en pleine croissance, solide et pérenne, connu pour ses marques Loué, Le Gaulois, Maître Coq, Marie et Traditions d'Asie.
	Contre - exemple	&lt ;br&gt ;&lt ;br&gt ;&lt ;br&gt ;Dans le cadre de sa politique diversité, Manpower étudie, à compétences égales, toutes candidatures dont celles de personnes en situation de handicap Profil :Etudiant opérateur approvisionneur (H/F)Poste en INTERIMEntreprise :Activité du client : Agroalimentaire

Les paires 3, 4, 5 et 7 montrent que le modèle déclenche un rejet autour du mot "*étudiant*" au singulier et pluriel. Cependant, le contexte joue énormément. Les différences sont parfois minimes, comme pour la paire 3 du tableau 4.5 où l'ajout du mot "minimum" rend la phrase acceptable.

La paire 6 du tableau 4.6 montre deux offres d'un même recruteur. Elle met en avant les mots "activités" et "devoirs" de la phrase rejetée ; non présents dans le contre-exemple associé. Une hypothèse, non vérifiée, est-ce que ces mots rapportent au champ de la garde d'enfant, a l'instar du mot école vu dans la paire 1. Hors, les offres de garde d'enfants comportent régulièrement des mentions de contrats étudiants, ce qui peut amener à un biais dans l'apprentissage de cette classe. Cet exemple a largement bénéficié de l'application des explications locales. En effet sans celles-ci, nous aurions cherché à analyser la partie du texte portant sur la mention "étudiant", alors qu'elle est identique pour l'exemple et le contre-exemple.

La caractérisation proposée a permis de mettre en avant la définition des frontières de la classe "*Discrimination : contrat étudiant*". Certains termes ont été mis en avant en tant que déclencheurs de rejet. Une analyse d'un expert du domaine est encore nécessaire pour définir les expressions acceptées ou non ; notamment dans notre exemple avec le mot "étudiant" et ses variantes ("étudiants acceptés", "contrat étudiant" etc.). D'autres termes amènent à penser qu'il existe un fort biais menant au rejet d'offres comportant des mots associés à la garde d'enfants. D'autres éléments ajoutent du bruit à la phrase analysée. Ce sont des noms propres peu connus, ou des problèmes de qualité des données. Ces éléments apportent des pistes d'amélioration du prétraitement des textes. Une piste consiste à remplacer tous les noms de localisations par un mot commun, tel que cela est fait en reconnaissance d'entités nommées. Les balises HTML et autres artefacts glissés dans le texte peuvent également être recherchés et supprimés via des expressions régulières.

4.4 Conclusion

Dans cette contribution, nous proposons un protocole de caractérisation par l'exemple de modèles complexes, de manière ad-hoc. Avec ce protocole, nous souhaitons fournir aux utilisateurs des éléments concrets pour les aider à créer leurs propres modèles mentaux.

Ce protocole consiste à extraire des éléments clés des données d'apprentissage, et à les trier pour présenter un ensemble digeste d'éléments. Ces éléments permettent à l'utilisateur ou utilisatrice effectuant la caractérisation du modèle de mieux appréhender

ses frontières de décision.

La méthode de caractérisation proposée ne requiert pas d'architecture spécifique (méthode ad-hoc). La stratégie est modulaire, permettant ainsi de modifier son implémentation à divers niveaux, la rendant flexible et adaptable. L'implémentation proposée s'appuie sur les données réelles extraites du jeu d'entraînement du modèle. Dans la mesure du possible, elle fait appel à des algorithmes de complexité limitée.

Toutefois, cette implémentation nécessite d'avoir accès aux données d'entraînement, ce qui n'est pas toujours possible.

L'application de ce protocole a également mis en évidence ses limites. Ainsi, nous avons profité du travail sur un modèle transparent pour récupérer les mots déclencheurs de rejet de manière fine. Sans l'ajout d'explications locales, l'analyse n'aurait pas été aussi poussée. L'analyse manque d'une analyse qualitative fine, réalisée avec des experts du domaine fonctionnel. Il aurait également été intéressant de pouvoir appliquer une correction du jeu d'entraînement, issu d'une telle analyse, et mesurer un éventuel gain (ou perte) de performance. Enfin, l'analyse est effectuée classe par classe. Nous présentons ici l'application sur une seule classe, qui comporte déjà un nombre important d'éléments. Il serait intéressant de mener une étude sur la capacité de génération d'un modèle mental complet par des personnes utilisant cet outil.

Notons que ce chapitre présente une première application, qui n'a pas été comparée aux méthodes de la littérature. Mettre en place une comparaison robuste serait souhaitable avant de chercher à la peaufiner. Dans l'état de l'art, la génération de contre-exemples nécessite l'entrainement de réseaux de neurones génératifs [3, 18]. La comparaison devrait ainsi prendre en compte le temps de calcul et les coûts d'entraînement des différentes méthodes.

Un autre axe d'approfondissement de ces travaux serait de rapprocher la recherche d'éléments pertinents avec les méthodes d'apprentissage actif. Les préoccupations de sobriété des algorithmes, communément rassemblées sous le nom de *green it*, font partie intégrante d'une volonté de développer des algorithmes et outils plus vertueux. À ce titre, nous nous sommes efforcés de privilégier des pistes peu coûteuses mais suffisamment intéressantes.

Résumé

- ✓ Une stratégie de caractérisation d'un modèle a été présentée
- ✓ Nous présentons une preuve de valeur d'une application de cette méthode à un cas d'usage
- ✓ L'application met en avant des éléments de compréhension du comportement du modèle
- ✓ L'application met en avant des limites de la méthode, qui perd en intérêt sans explications locales
- ✓ Un comparatif qualitatif robuste avec d'autres méthodes de la littérature est nécessaire

Chapitre 5

Intégration à l'environnement industriel

Dans ce chapitre

Ce chapitre présente les liens entre explicabilité et problématiques de Pôle emploi. Nous décrivons le cadre logiciel *Gabarit* et les développements de la thèse réalisés dans ce dernier. Les apports de la démarche d'explicabilité aux engagements éthiques sont présentés. Enfin, nous proposons un guide des méthodes d'explicabilité dédiées à l'industrie.

Ce chapitre présente l'intégration des travaux de cette thèse dans l'environnement industriel de Pôle emploi. Les liens entre ces travaux et l'entreprise sont multiples. D'un point de vue technique les contributions d'explicabilité présentées sont implémentées dans l'infrastructure de développement de Pôle emploi : *Gabarit*. D'un point de vue des enjeux éthiques autour de l'IA chez Pôle emploi, ce chapitre participe à la mise en place et à la réponse aux exigences de la charte éthique IA.

Nous présentons l'apport cette thèse à l'outil, en section 5.1. La section 5.2, montre la contribution de la thèse pour la mise en œuvre de la charte éthique Pôle emploi. Enfin, un guide est proposé en section 5.3 afin de choisir d'une méthode d'explicabilité dans un contexte industriel.

5.1 Gabarit

Dans cette section, nous présentons *Gabarit*[1], un outil open source créé par Pôle Emploi. Sous la forme de module python, il permet de générer des projets d'apprentissage

1. `https://github.com/OSS-Pole-Emploi/gabarit`

automatique prêts à l'emploi. Il permet ainsi d'uniformiser les pratiques, favoriser le partage, accélérer la mise en production des modèles d'IA. Dans ce chapitre, nous présentons l'historique de l'outil, sa philosophie, et comment s'y intègrent mes travaux de thèse.

5.1.1 Historique

Les premiers travaux de création d'outillage communs autour de la donnée démarrent en 2018. Ils sont liés au besoin du département *Agence Data Services* d'unifier les processus spécifiques de nettoyage de données textuelles. Cet outil, nommé *Words'n'fun*, présente un ensemble de fonctions python pour le prétraitement de textes [2]. Ce besoin d'uniformisation des bonnes pratiques s'étend au développement de modèles d'IA en NLP. La volonté est de faciliter leur mise en production et le transfert du projet d'une personne à l'autre. Le *Template NLP* est créé en 2019, en tant que projet git à télécharger. Il est axé sur l'entraînement, la sérialisation et la mise à disposition de modèles d'IA. L'outil est un générateur de projets, il embarque et intègre divers outils open source pour historiser les données avec *Data Version Control* (DVC) [3] et les prétraiter avec *Words'n'fun*), effectuer l'apprentissage automatique (*Scikit-Learn* [4], *Tensorflow* [5] et *Torch* [6]), aider au suivi des modèles (*MLflow* [7], *Artifactory* [8]) ainsi qu'un démonstrateur *Streamlit* [9].

Pour faciliter sa prise en main par les scientifiques des données de l'entreprise, une formation au travers de tutoriels interactifs est mise en place. Les travaux de cette thèse sont réalisés en utilisant cet outil, contribuant ainsi directement à l'ajout de fonctionnalités d'explicabilité.

En 2020 et 2021 sont intégrés au *Template NLP* les *Template vision* et *Template numérique*, gérant respectivement les données tabulaires et les images. L'outil évolue selon les besoins et les contraintes de mise en production. Dès le démarrage du projet, son ouverture à la communauté open source est considérée. En 2022, le service juridique donne son aval pour que le code soit rendu disponible sous licence copyleft ; ce qui donne lieu à des travaux de refonte. Un des points principaux est la traduction en anglais, car le code interne Pôle emploi est historiquement écrit en anglais avec des commentaires français, ce

2. `https://github.com/OSS-Pole-Emploi/words_n_fun`
3. `https://dvc.org/`
4. `https://scikit-learn.org/`
5. `https://www.tensorflow.org/`
6. `https://pytorch.org/`
7. `https://mlflow.org/`
8. `https://jfrog.com/fr/artifactory/`
9. `https://streamlit.io/`

qui limite le partage à la communauté. Les tests sont améliorés pour consolider le projet, et l'installation est facilitée en partageant le projet sous forme de module python. Le tout est rendu public en 2022 sous le nom *Gabarit*.

L'outil *Gabarit* est composé de 2 éléments : le générateur de projets et le projet généré à utiliser. Le générateur est présenté dans la section suivante, et le projet généré est présenté à la suite.

5.1.2 Le générateur de projets

Le générateur de projets permet de créer un projet d'IA en une ligne de commandes. Il gère notamment les formats de données lus et sérialisés (encodage, séparateur de colonne des fichiers csv...), et les éventuels outils à configurer tel que DVC.

Une fois le module installé (ou mis à jour), il est directement possible de créer un nouveau projet. La ligne de commande permet de générer un projet d'analyse de textes (*generate_nlp_project*), d'images (*generate_vision_project*) ou de données tabulaires (*generate_num_project*). D'autres précisions optionnelles sont disponibles pour la gestion du projet : intégration de DVC, configuration spécifique (encodage...), etc. Plus de détails sont donnés dans la documentation du projet.

Le générateur repose sur une arborescence de fichiers "templates" correspondant chacun à une typologie de données (texte, image et tabulaire). Chaque *template* contient le code source presque prêt à l'emploi : il ne reste qu'à remplacer le nom du projet, à la manière d'un texte à compléter, ce qui est effectué automatiquement lors de la génération du projet. Le générateur peut ainsi créer le projet souhaité avec le nom et l'emplacement donnés, ainsi que les spécifications optionnelles. Tous les fichiers sont générés pour être prêts à l'emploi.

5.1.3 Architecture du projet généré

Une fois le projet généré et installé, tout le code du projet peut être modifié par le scientifique des données qui y a un accès direct, à la différence d'outils plus graphiques tels que Dataiku. Il reste possible d'intégrer par la suite tout module python souhaité. C'est également le moment idéal pour démarrer l'historisation sur *git*, si cela n'a pas déjà été mis en place.

Une fois le projet installé, il ne reste plus qu'à intégrer les données souhaitées. Des fonctions sont à disposition pour prétraiter les données. Des architectures de modèles d'ap-

prentissage automatique sont à disposition des utilisateurs, basées sur le trio *Scikit-learn*, *Tensorflow*, *Pytorch*. Il est possible d'entraîner directement un de ces modèles, d'ajuster leur architecture, ou encore d'en créer un nouveau, le tout avec très peu de modifications de code à effectuer. Les architectures de modèle couvrent à la fois l'apprentissage automatique et profond.

La figure 5.1 illustre les six fonctionnalités liées directement au modèle et à son cycle de vie, à savoir : le définir, l'entraîner (méthode *fit*), l'enregistrer et recharger (*save* et *load_model*), l'utiliser pour prédire un résultat (*predict*) et expliquer celui-ci (*explain*).

```
# Description du modèle et de ses paramètres
model = melsa.ModelEmbeddingLstmStructuredAttention(x_col, y_col, batch_size, epochs, max_sequence_length, max_words)

# Entrainement du modèle, x correspondant aux données d'entrées, et y à la valeur cible à prédire
model.fit(x_train, y_train, x_valid, y_valid)

# Sérialisation et chargement
model.save()
model, model_conf = model.load(selected_model)

# Inférences pour le jeu de données dataset, sur la colone d'entrée x_col
model.predict(dataset[x_col])

# Explications associées aux inférences
model.explain(dataset[x_col])
```

FIGURE 5.1 – Illustration des fonctionnalités de *Gabarit* pour un modèle : définition, entraînement, sérialisation, chargement, inférence et explication.

L'architecture de chaque modèle est disponible dans un script associé, ce qui permet notamment d'ajouter, supprimer, modifier des couches de réseaux.

Le projet intègre également des interfaces avec divers outils, notamment un démonstrateur via une page web, et la mise à disposition du modèle via *artifactory* afin d'être utilisé dans différents services et outils.

Cet environnement a servi de base pour les développements des travaux présentés ici. De même, les travaux ont nourri l'outil *Gabarit*. Nous présentons cet échange dans la section suivante.

5.1.4 Intégration des travaux d'explicabilité

Les travaux présentés ici sur l'explicabilité des algorithmes ont été développés dans *Gabarit*. Le modèle à attention présenté en chapitre 4 a été conçu au sein de l'outil, ainsi que les développements liés aux explications globales. De même, le démonstrateur fourni a servi de base aux illustrations des différentes méthodes d'explications.

Réciproquement, une partie des avancées de cette thèse ont servi à enrichir l'outil. Le principal avantage est l'intégration directe d'une méthode d'explicabilité pour tout modèle, via la méthode *model.explain*. L'explication est par défaut apportée par Lime [90], une méthode agnostique au modèle parmi les plus légères, et en moyenne plus rapide que les ancres utilisées dans les chapitres 3 et 4. Lime sert de premier point de référence pour les scientifiques des données. Notre état de l'art montre qu'il existe toutefois d'autres méthodes intéressantes.

L'architecture du modèle à attention utilisé y est mise à disposition en générant un projet d'analyse de textes, via la classe *ModelEmbeddingLstmStructuredAttention*. Les explications par attention sont récupérables, à titre expérimental et pour cette classe uniquement, via la méthode *model.explain_indexes*. L'intégration au fil de l'eau des contributions au cadre logiciel permet aux équipes de Pôle emploi ou autres utilisateurs de l'outil d'intégrer la notion d'explicabilité à tous les modèles d'IA.

Pour conclure brièvement sur cette section, *Gabarit*, a accéléré la mise à disposition des développements effectués dans le cadre de cette thèse. En l'employant, nous avons facilité l'accès aux travaux d'explicabilité, pour les équipes de Pôle emploi, mais également à toute personne intéressée.

L'outil étant conçu pour faciliter la mise en production d'outils d'IA, il possède toutefois des contraintes non adaptées à un environnement de recherche pure. C'est un outil lourd, faisant appel à de nombreuses dépendances. Celles-ci sont ajustables grâce à la modification directe du code python, mais il est contre-productif de s'écarter totalement du cadre prévu. Ainsi, les premiers développements de cette thèse se sont faits avant l'intégration de Pytorch, ce qui a contraint à réaliser les développements sur tensorflow.

Les travaux mentionnés dans cette section ont été techniques, mais ce n'est pas le seul apport pour l'entreprise. La section suivante montre leur intérêt dans la réflexion éthique réalisée à Pôle emploi.

5.2 Éthique

L'éthique de l'IA est un enjeu présent à Pole emploi depuis 2018. La figure 5.2 montre la chronologie et les grands points d'étape de la réflexion autour de ces enjeux. Les étapes auxquelles j'ai activement contribué, notamment en participant aux réunions d'avancement, sont encadrées en noir. L'éthique est d'abord abordée par le prisme de la maitrise des risques. Les premiers travaux mènent à la construction de la *charte pour une IA*

éthique [83]. C'est en parallèle et dans le même contexte que les travaux de la présente thèse sur l'explicabilité des algorithmes d'IA sont proposés, le sujet étant affiné au long de l'année 2019.

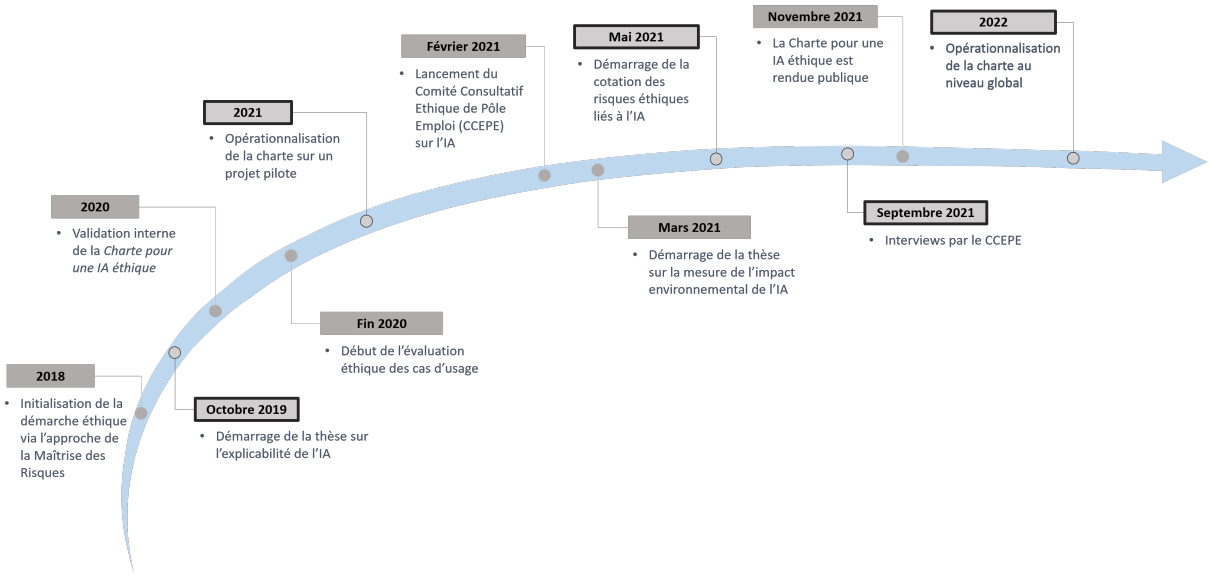


FIGURE 5.2 – Chrolonogie des projets et évènements liés à l'éthique à Pôle Emploi depuis 2018. Les projets encadrés en noir sont ceux auxquels j'ai activement participé.

Ainsi, avec les réflexions entamées en 2018, la *charte pour une IA éthique* fait son apparition dès 2020. Un premier point d'évaluation de l'existant est alors effectué dans la foulée. La figure 5.2 montre l'accélération des travaux en 2021. Un projet pilote met en place une première salve d'ateliers d'*opérationnalisation* de la charte.

En début d'année, la création du *Comité Consultatif Éthique de Pôle Emploi* (CCEPE) est actée. Ce comité externe est composé de dix experts d'horizons complémentaires : universitaires, représentants de syndicats et des usagers. Ce CCEPE a pour vocation d'accompagner et porter un regard externe critique sur les engagements et actions de l'établissement. Dans ce cadre, le CCEPE a dialogué activement avec les acteurs de Pôle emploi, me permettant de présenter l'avancée des travaux sur explicabilité.

Par la suite, les travaux de thèse d'Angela Ciocan sur la mesure de l'impact environnemental de l'IA démarrent. Les risques éthiques liés à l'IA sont évalués de façon plus globale pour tous les sujets existants et à venir. Les ateliers de cotation de ces risques mêlent des profils techniques et managériaux. Ces cotations sont une première étape permettant l'opérationnalisation de la charte.

Par la suite, nous présenterons la charte et ses grands principes en section 5.2.1. Sa

mise en œuvre est traitée, de la cotation des risques à la l'établissement d'outils pour les acteurs des projets IA, en section 5.2.2.

5.2.1 La charte éthique

La charte est publiée en interne dans un premier temps, puis publiquement en Novembre 2021. Elle est donc disponible sur le site de communication officiel de Pôle emploi [10]. Elle est également en annexe en section A.1. Cette charte répond au besoin de suivi interne, mais également à la volonté d'être auditables.

Les 7 axes de la charte :

1. Finalité et légitimité des algorithmes,
2. L'humain au centre ; l'intelligence artificielle au service de l'humain,
3. Equité et non-discrimination,
4. Liberté de choix,
5. Transparence,
6. Sécurité,
7. Impact environnemental.

Finalité et légitimité des algorithmes Cet axe implique d'utiliser les outils d'IA à bon escient, dans le but de fournir un service bénéfique ou de lutter contre les actes de malveillance.

L'humain au centre ; l'intelligence artificielle au service de l'humain Les outils sont créés pour accompagner l'humain dans ses tâches habituelles. Pôle emploi indique dans sa charte s'engager à fournir "une explication sur le fonctionnement d'un service ou une aide à la décision" ainsi que "des actions d'accompagnement et de sensibilisation [...] à l'intelligence artificielle" aux utilisateurs internes (*agents*) et externes (*usagers*) [83].

Equité et non-discrimination Il s'agit ici à la fois de reconnaitre que l'IA peut "reproduire, renforcer ou générer des biais discriminatoires", et veiller à limiter l'impact de ceux-ci [83]. Le contexte applicatif de l'établissement est à risque du fait de biais humains présents dans le monde du travail. Les discriminations à l'embauche ou les métiers dits "genrés" en sont des exemples.

10. `https://tinyurl.com/charteIA`

Liberté de choix Les décisions prises par un algorithme peuvent toujours être modifiées par un humain, et les utilisateurs ont accès à un interlocuteur pour demander un tel recours. Il est toujours possible d'ignorer les recommandations des algorithmes d'aide à la décision. En d'autres termes, l'humain a le dernier mot.

Transparence Cet axe répond aux exigences du RGPD et de la loi pour une république numérique [64]. Il comprend le recueil du consentement éclairé des utilisateurs pour collecter et traiter leurs données. La transparence concerne ici l'information aux utilisateurs lorsqu'iels ont à faire à un outil automatique, potentiellement basé sur l'IA. Cette exigence couvre notamment les agents conversationnels (ou "chatbots"). La charte indique très clairement que l'indication de l'utilisation d'un service d'IA s'accompagne de la "capacité d'expliquer de la façon la plus compréhensible possible son fonctionnement et ses résultats" [83].

Sécurité La sécurisation concerne le traitement de données personnelles et sensibles, les paiements et la lutte contre les attaques (fraudes, arnaques, vol de données). Dans le cadre de l'IA, il s'agit notamment de valider l'usage de données personnelles et sensibles, si nécessaire les anonymiser. La sécurisation des services passe également par la vérification de leur résistance aux attaques adversaires.

Impact environnemental Dans une démarche d'usage responsable des outils à disposition, Pôle emploi reste attentif à l'impact et au coût environnemental des solutions développées. La démarche étant déjà entamée pour des développements hors IA, il s'agit ici d'une adaptation des bonnes pratiques existantes au développement, spécifique, des modèles d'apprentissage automatiques et profonds.

Les travaux présentés dans ce manuscrit s'inscrivent directement dans deux des axes de la charte. Pour respecter les engagements des axes de *l'humain au centre*, et de la *Transparence*, il faut être en mesure d'expliquer le fonctionnement des algorithmes d'IA utilisés, dans leur fonctionnement global mais également lors de recours pour un résultat spécifique.

5.2.2 Mise en œuvre de la charte

La création de la charte est une première étape, mais son existence ne suffit pas à rendre les travaux entamés et révolus plus éthiques. Après une première étude sur un

projet pilote, la mise en œuvre globale de la charte s'est faite en deux temps.

Le projet pilote étudié porte sur l'analyse de mails afin d'assister les agents Pôle emploi. L'objectif est d'apposer une étiquette à chaque mail reçu par un conseiller : s'agit-t-il d'une demande de rendez-vous ? D'une réclamation ? Ces étiquettes permettent aux agents de s'organiser et construire leur planning plus facilement, en traitant ensemble toutes leurs demandes de rendez-vous, par exemple. Ce projet pilote à l'avantage de faire appel à de nombreuses notions d'éthique (sécurité des données, non-discrimination, assistance et non remplacement du travail des conseillers...). L'étude met en avant les caractéristiques spécifiques de ce projet avec ses acteurs, les données traitées etc. Les sept axes de la charte sont abordés un à un, et les points d'attention sont abordés avec les risques associés, les mesures déjà mises en place et les améliorations envisageables.

Les travaux de mise en œuvre de la Charte se sont par la suite généralisés à tous les cas d'usage IA de Pôle emploi. La première étape a été la cotation des risques associés à chaque axe de la charte, selon leur probabilité et leur impact. Un risque est, par exemple une *Défaillance ou absence dans la capacité à expliquer de la façon la plus compréhensible possible son fonctionnement et ses résultats*. La probabilité d'occurrence P de ce risque est quantifiée entre 1, très peu probable, et 4, fréquent. Les impacts sont également notés entre 1 pour un impact faible, et 4 pour un impact capital. Ces impacts sont évalués au travers de 6 catégories :

1. l' **image**, interne et médiatique, de l'établissement,
2. les **finances** englobant les risques de pertes financières ou dépenses imprévues,
3. **réglementaire**, allant du non-respect de la charte jusqu'au risque pénal,
4. la réalisation des **missions**, comprenant les retards, la non réalisation des missions confiées,
5. le **climat social** interne, comprenant les tensions pour les agents, allant jusqu'au risque de grève,
6. l'**humain**, prenant en compte les agents et les usagers (demandeurs d'emploi et recruteurs).

Ces impacts sont cotés collégialement grâce à un panel d'agents référents techniques, métiers, décisionnaires. La gravité G du risque est calculée en fonction des cotations selon l'équation :

$$G = P * \frac{I_{Image} + I_{Finance} + I_{Reglementaire} + I_{Mission} + I_{Social} + I_{Humain}}{6} \tag{5.1}$$

Le risque est maîtrisé lorsque $G \in [1, 4[$, et préoccupant lorsque $G \in [12, 16]$.

À titre d'illustration, sans les travaux de cette thèse, la gravité de ce risque est de $4 * (3 + 2 + 1 + 2 + 2 + 1)/6 = 7,2$. Avec la prise en compte de ces travaux, la gravité devient $1 * (3 + 2 + 1 + 2 + 2 + 1)/6 = 1,8$. Le risque, auparavant moyen, est alors maîtrisé. Cette illustration est tirée de la documentation interne de Pôle emploi [84].

La seconde étape de cotation consiste à déterminer les actions associées aux risques détectés, afin de diminuer l'impact de ceux-ci. Les actions peuvent diminuer les impacts ou la probabilité d'occurrence des risques. Ces actions sont globales, générales afin de s'adapter à tout projet. Reprenons le risque d'incapacité à expliquer un algorithme. Une action liée peut être *Utiliser lorsque c'est possible des algorithmes et des solutions d'explicabilité afin d'être transparent sur les décisions prises.* Ces actions globales sont complétées par des précisions opérationnelles : tâches à mener, acteurs impliqués, moyens nécessaires et moment privilégié pour réaliser l'action. Dans l'exemple mentionné portant sur l'explicabilité, les travaux présentés dans ce manuscrit ont servi de base aux réflexions, apportant une vision des solutions existantes, et une preuve de première application.

Les travaux sur l'éthique ainsi que la cartographie des risques présents ont bénéficié, pour deux des sept axes de la charte, des réflexions engagées ici. Nous avons montré que nous pouvons réduire les risques rencontrés, en passant d'une volonté d'explicabilité à une application concrète de méthodes d'explications sur nos données, présentés en chapitres 2 et 4. Ces avancées nécessitent toutefois une priorisation et un suivi adéquats par les personnes en charge des différents projets. Nous conclurons cette section en soulevant le questionnement suivant : le devoir d'expliquer un résultat ou le fonctionnement d'un service doit-il être exigé uniquement lorsque ce dernier est basé sur des algorithmes d'intelligence artificielle ?

De manière concrète, et issue des savoirs engrangés dans cette aventure de trois ans, nous proposons à cet effet un guide permettant de faire le tri parmi les nombreuses méthodes d'explicabilité, en fonction des objectifs à atteindre.

5.3 Comment choisir une méthode d'explicabilité ?

Cette section présente un guide permettant de déterminer les caractéristiques souhaitées pour une méthode d'explication. Les éléments de la section 1.1 sont repris et organisés afin d'aider à la détermination de caractéristiques idéales pour un projet donné, en fonction de ses contraintes. Ce guide est inspiré de travaux existants [110], simplifié et adapté

au contexte industriel.

Cette section fournit ainsi des outils qui aideront le lecteur ou la lectrice à déterminer ses besoins, et *in fine* à sélectionner un panel restreint de méthodes de la littérature qui conviendront le mieux à ces besoins.

Guide des caractéristiques recherchées Pour déterminer les systèmes d'explication préférables dans un projet, il faut au préalable déterminer les caractéristiques recherchées. Pour déterminer ces caractéristiques, il faut avoir en tête l'objectif, le public, et le contexte de réception des explications. Il est possible que pour un même modèle d'IA, plusieurs besoins d'explicabilité soient détectés, à des moments de vie différents. Dans ce cas, l'exercice proposé dans cette section est à réaliser pour chaque besoin.

Cette section propose un guide, permettant de définir les caractéristiques souhaitées pour un système d'explication. Chaque caractéristique vue en introduction est passée en revue, à savoir la portée, la stratégie, et le format d'explication.

Choix de la portée Les données principales permettant de choisir la portée de l'explication sont les utilisateurs et leurs objectifs. Tel que vu en introduction, une explication de portée globale permet de construire la confiance dans le modèle sur son comportement général. Elle est pertinente pour les phases de validation et d'appropriation du modèle. Une explication locale permet de se concentrer sur un résultat, et est plus proche des conditions réelles de l'utilisation du modèle.

La figure 5.3 montre les publics concernés, à savoir :

— les régulateurs, qui s'assurent de la conformité des outils au regard de la loi,
— les gestionnaires du projet, qui s'assurent de l'efficacité du service rendu,
— les scientifiques des données, qui produisent les modèles,
— les experts du domaine, qui utilisent les outils ou assistent les personnes concernées,
— les personnes concernées par l'usage de l'outil, si celui-ci impacte leur vie.

Ces publics sont rattachés à leurs objectifs dans le cycle de vie d'un projet. Selon les organisations, les publics et objectifs peuvent varier, la figure 5.3 sert de modélisation globale et doit être adaptée.

Ainsi, dans la figure 5.3, un expert du domaine avec pour objectif de créer de la confiance dans un outil d'IA peut recevoir des explications globales ou locales. Les explications globales seront plutôt adaptées en phase de test et appropriation de l'outil, en amont de la mise en production. Les explications locales seront plus adaptées lors d'une

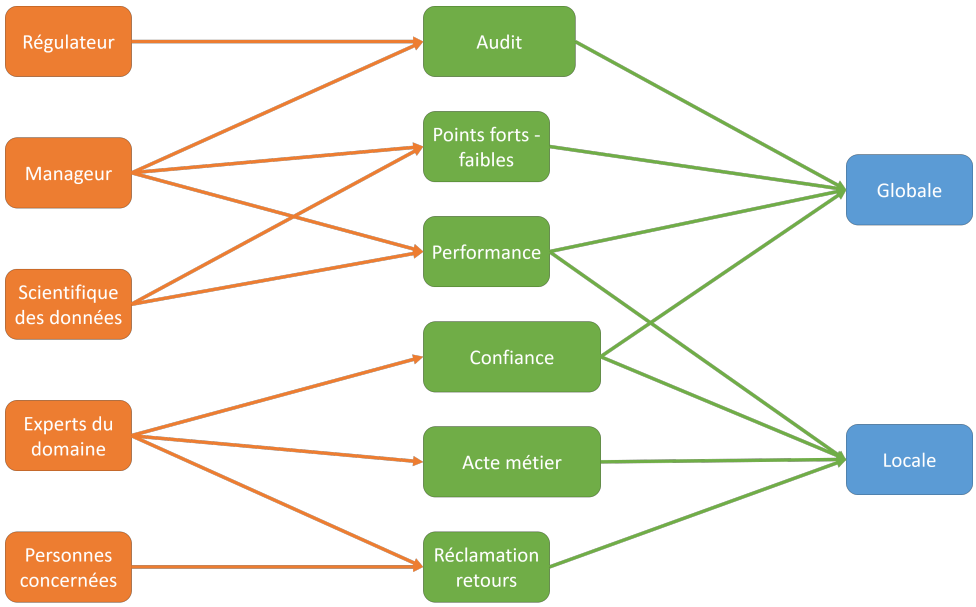


FIGURE 5.3 – Illustration du choix de la portée selon l'utilisateur et l'objectif de ce dernier

utilisation en conditions réelles d'utilisation.

Choix de la stratégie La stratégie à adopter est déterminée par plusieurs facteurs tels que l'accès au système d'intelligence artificielle, sa phase de conception, et son architecture si elle est déjà choisie. Les approches boites transparentes sont les plus contraignantes et les plus coûteuses à la conception, mais peuvent limiter les calculs de génération des explications à posteriori, et sont par nature fidèles au fonctionnement interne du modèle à expliquer. Les approches boites grises sont également basées sur le fonctionnement du modèle, mais nécessitent des calculs à postériori et sont basées sur les architectures des modèles à expliquer. Elles nécessitent d'y avoir accès afin de générer l'explication. Les approches boites noires sont les moins contraignantes, mais peuvent s'avérer coûteuses en calcul pour générer l'explication. Puisqu'il n'y a pas besoin d'avoir accès au modèle, elles sont également moins fidèles au fonctionnement interne du modèle que les autres stratégies.

La figure 5.4 illustre les stratégies à disposition. Au centre du cercle les stratégies boites noires (BN), sont applicables en toutes circonstances. Lorsque les conditions sont réunies, les stratégies boites grises (BG) s'ajoutent au panel des méthodes applicables, l'étendue des méthodes à disposition est alors élargie. Sur le même principe, les stratégies boites transparentes (BT) sont applicables sous certaines contraintes. Elles s'ajoutent alors aux

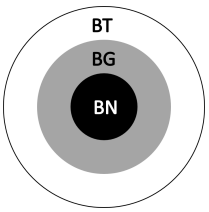


FIGURE 5.4 – Cercle des possibles des stratégies de la moins contraignante à la plus contraignante, du centre vers l'extérieur.

autres méthodes, élargissant encore le cercle des possibles, comme l'illustre la figure5.4.

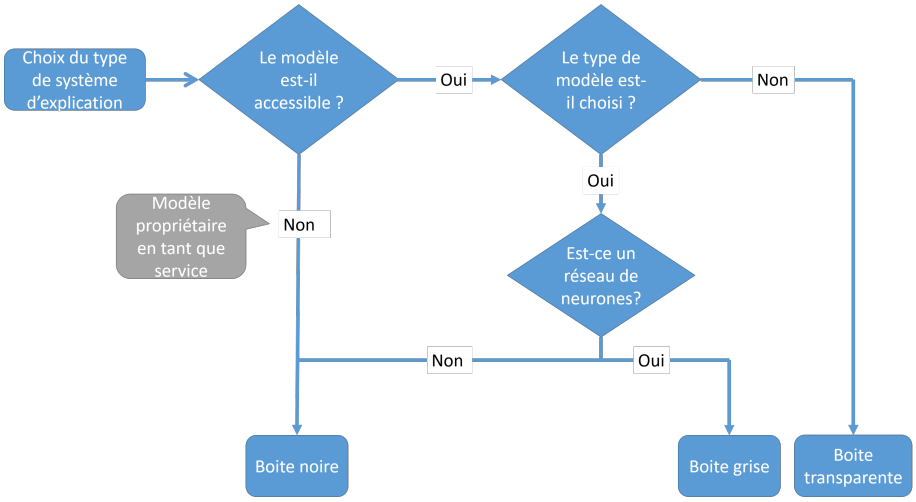


FIGURE 5.5 – Illustration du choix de la stratégie la plus transparente applicable selon les contraintes du projet.

La figure 5.5 présente les choix de stratégies possibles selon les contraintes du modèle. Le choix le moins restrictif est celui de la boite noire, applicable dans tous les cas, car il se base sur les entrées et sorties du modèle. Il est notamment adapté si le modèle est développé en externe ou si son fonctionnement interne est inaccessible pour une raison ou une autre. Si le modèle est accessible et déjà conçu ou si son architecture est déjà choisie, alors, en fonction de cette dernière, certaines méthodes en stratégie boite grise sont applicables. Sinon, il faut se reporter sur les méthodes boites noires. Si le modèle reste à concevoir, il est possible de concevoir son architecture avec une approche boite transparente. Les approches boite grise et boite noire restent également une option.

Choix du format Afin de choisir un type de format où un autre, il faut étudier les données à disposition et cibler les besoins des utilisateurs. La figure 5.6 schématise le choix

d'un format par rapport à un autre. Si le schéma permet de privilégier une solution ou une autre, le format des explications doit s'intégrer à l'interface de l'outil où elles apparaissent. La mise en avant de variables ou d'exemples s'appuie sur des éléments concrets. Ces méthodes conviennent à une utilisation en condition réelle. Mettre en avant une partie du contenu particulièrement utile lorsque l'exemple entier est trop long à analyser (image, texte composé de plusieurs phrases). Les règles sont plus facilement généralisables, et sont ainsi adaptées aux audits. Les explications générées contiennent des formats plus variés, et sont à envisager lorsque les informations souhaitées ne sont pas lisibles dans les données directement : génération d'une phrase d'explications, instances menant en erreur, extraction de motifs représentatifs etc. Si le schéma permet de privilégier une solution ou une autre, le format des explications doit s'intégrer à l'interface de l'outil où elles apparaissent. Le test d'utilisabilité tel que présenté en section 3.1 est idéalement à réaliser en complément.

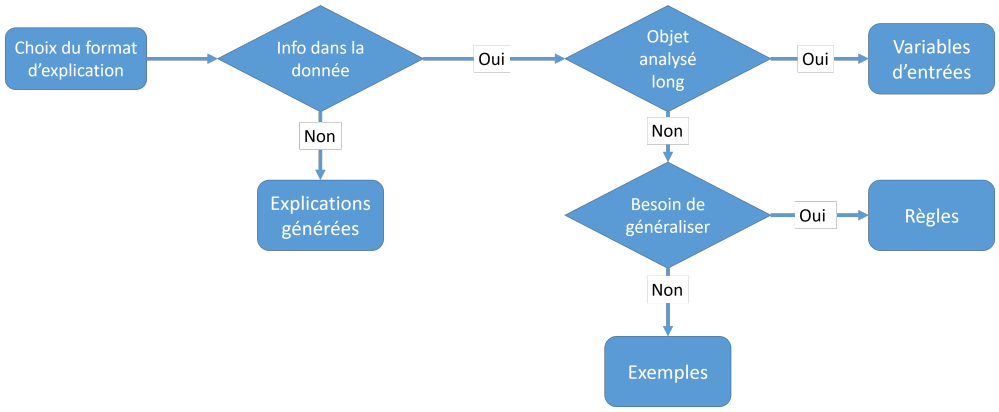


FIGURE 5.6 – Illustration du choix de format des explications selon les contraintes du projet.

Une fois un ensemble de caractéristiques choisi, il est possible de se référer aux nombreuses méthodes de la littérature pour trouver un ensemble de méthodes d'explication candidates à essayer.

5.4 Conclusion

Dans ce chapitre, nous avons montré les liens entre les travaux menés et les contraintes et objectifs industriels de Pôle emploi.

Nous avons présenté l'intégration technique des travaux de thèse aux contraintes de

mise en production, via l'outil *Gabarit*. Nous avons également mis en avant le lien entre explicabilité et respect de la charte éthique. Finalement, nous avons donné des indications permettant aux gestionnaires de projet de sélectionner des méthodes d'explicabilité en fonction de leurs objectifs.

Résumé

- ✓ Les fonctionnalités d'explicabilité sont intégrées et disponibles dans le cadre logiciel open source *Gabarit*
- ✓ L'outil *Gabarit* permet de diffuser les avancées sur l'explicabilité de l'IA
- ✓ L'explicabilité fait partie des engagements éthiques pris par l'entreprise
- ✓ Les travaux de cette thèse permettent de répondre en partie aux engagements pris

Conclusion

Apports de la thèse

Dans cette thèse, nous avons présenté des moyens de comprendre des modèles profonds dans le cadre de l'analyse de textes. L'analyse de l'état de l'art montre la forte diversité des types d'explications existantes. Les méthodes de génération d'explications, notamment locales, sont par conséquent également nombreuses. Cette diversité met en avant l'importance du contexte et du but d'une explication. Un jeu de données a été créé en collectant des données et en annotant manuellement des explications de référence associées.

Un test d'utilisabilité réalisé avec des utilisateurs experts a servi au développement d'interfaces d'explications locales dans un démonstrateur. Les interfaces permettent la visualisation d'explications adaptées au langage naturel. Le test d'utilisabilité a donné l'idée d'une interface non envisagée avant, basée sur les règles métier. Les retours des experts du domaine ont relevé que les systèmes d'explications les plus complexes ne mènent pas nécessairement aux interfaces les plus appréciées des utilisateurs.

Nous nous sommes ensuite concentrés sur une des interfaces, à savoir le surlignage de mots importants. Nous avons appliqué deux méthodes de génération d'explications locales de la littérature : les ancres [91] et les explications par attention [66]. Les explications locales permettent de comprendre un résultat spécifique d'un modèle d'intelligence artificielle.

Nous avons mis en place un protocole de comparaison d'explications locales avec et sans utilisateurs. Pour cela, nous avons utilisé les données annotées avec explications de référence, les explications générées, et le démonstrateur développé. Cette comparaison nous a permis d'effectuer un choix basé sur des mesures objectives. Pour le cas d'usage LEGO, la méthode des ancres a donné les meilleurs résultats, tandis que l'attention a été plus pertinente pour le cas d'usage Yelp. L'analyse des préférences utilisateurs a permis de constater que, malgré l'utilisation d'une documentation experte pour définir des explications de références, celles-ci ne correspond pas aux attentes des utilisateurs. Nos discussions avec les participants des expérimentations ont démontré que ces derniers n'étaient

pas en accord sur la définition de ces explications, et qu'un travail de concertation est nécessaire. Grâce à ces travaux, nous avons déterminé une mesure de performance cible : $performance = \frac{e_{generee} \cap e}{e_{attendue} \cup e} \frac{\alpha}{len(e)}$. Par ailleurs, nous avons été confrontés à des problématiques spécifiques au langage naturel : l'importance du contexte des mots, mais également présence de mots vides de sens, souhaitée par une partie des utilisateurs.

Une méthode modulaire adaptable de création d'explications globales a été conçue, implémentée et appliquée à nos données. Cette méthode est une aide à la création d'un modèle mental pour l'utilisateur. Elle permet une meilleure appréhension du comportement du modèle.

Nous avons enfin montré que les travaux sont intégrés au cadre logiciel de Pôle emploi et inscrits dans les engagements éthiques de l'établissement. Nous avons établi leur impact positif dans l'atteinte des objectifs opérationnels de la charte éthique.

Perspectives

La comparaison des explications par les utilisateurs a permis de constater une faible adhérence aux explications générées. Nous en déduisons qu'il serait pertinent d'avoir des utilisateurs à disposition sur un temps plus long, afin de définir ces explications idéales. Nous pourrions alors annoter avec eux, de façon collégiale, un jeu de données avec explications de référence plus conséquent.

Avec un tel jeu de données, il serait intéressant de relancer le protocole de comparaison d'explications, et ainsi affiner la mesure de performance des explications adaptées à nos utilisateurs. Par ailleurs, cette comparaison peut être complétée en mesurant les performances de réalisation de l'acte métier avec et sans explications. Cela aiderait à déterminer la quantité d'offres qui sont validées malgré une alerte. Cela peut aussi déterminer si l'ajout de l'explication permet d'améliorer la rapidité ou la qualité des corrections.

Maintenant que nous avons une stratégie d'évaluation des explications, nous pouvons envisager de faire apprendre à un modèle à générer ses explications par attention, avec une fonction de coût qui pénaliserai les "mauvaises" distributions d'attention. Une solution pour pallier à ce problème est d'apprendre des explications attendues sous forme de cartes d'attention, générées par des utilisateurs humains [11]. Cet apprentissage peut se faire de façon active [105].

Nous avons également créé des explications globales du comportement d'un modèle. Grâce à ces travaux, il est possible d'appliquer une correction du jeu d'entraînement du

modèle, issu des explications générées. Mesurer un éventuel gain ou perte de performance d'un modèle après une telle correction serait une preuve de valeur pertinente. Ces travaux nous amènent à proposer de mesurer la fidélité du modèle mental des utilisateurs, tel que cela a été réalisé dans la littérature. Encore une fois les expérimentations avec utilisateurs nécessitent des temps longs, mais sont enrichissantes. De même, il serait intéressant de chercher à appliquer les méthodes d'apprentissage actif dans le cadre du tri d'éléments pertinents.

Les explications globales présentées dans le chapitre 5 sont basées sur des contre-exemples. Une piste d'approfondissement de ces travaux serait la mesure et l'optimisation de la pertinence d'un contre-exemple en fonction d'un exemple précis. Obtenir des performances suffisantes permettrait de proposer des corrections de texte.

Nos travaux nous amènent à mieux comprendre le fonctionnement des modèles profonds dans un cadre naïf. Les attaques adversaires, permettant de fausser les résultats d'un modèle, forment une menace. Les explications peuvent être faussées au même titre que les résultats [32]. Nous proposons de se pencher sur la question complémentaire : est-ce que l'explicabilité peut aider à se prémunir de certaines attaques ? Le cas des "déclencheurs" mérite d'être étudié. Ces artefacts non perceptibles, issus de données d'entraînement piégées, détériorent les performances des modèles. Il serait intéressant de tenter de détecter non seulement ces déclencheurs via des explications locales, mais également les données d'apprentissage piégées ayant créé ce déclencheur grâce à l'analyse globale du modèle.

La dérive des données entraîne la dégradation des modèles d'IA en production au fil du temps. Avec les explications globales que nous proposons, basées sur les données d'entraînement, nous pouvons aller plus loin et imaginer détecter la dérive des données.

Nos travaux ont apporté des solutions applicables au cadre industriel, tout en ajoutant des contraintes : calculs supplémentaires pour les explications, nécessité de conserver les données d'entraînement d'un modèle, etc. L'impact opérationnel de ces contraintes pour les outils en production mérite d'être mesuré à court et moyen terme. De même, il serait pertinent d'étudier l'impact de ces mesures sur la capacité à suivre les recommandations de développements éco-responsables.

Annexe A

Annexes

A.1 Charte éthique

La charte éthique de Pôle emploi, dans sa version publique présente en octobre 2022 sur le site institutionnel de l'établissement `pole-emploi.org`.

RÉPUBLIQUE FRANÇAISE

Liberté
Égalité
Fraternité

pôle emploi

Charte de Pôle emploi pour une intelligence artificielle éthique

INSTITUTIONNEL

www.pole-emploi.org

PRÉAMBULE

L'"intelligence artificielle"* (IA) est un levier important pour Pôle emploi, afin d'accélérer le retour à l'emploi durable. Ses potentialités sont nombreuses, tant pour les agents* de Pôle emploi que pour ses usagers*, demandeurs d'emploi et recruteurs notamment. Les technologies intégrant de l'IA peuvent engendrer des questionnements éthiques* et soulever des défis sociétaux, en raison des spécificités et des craintes qu'elles suscitent.

Pour répondre à ces enjeux, Pôle emploi entend inscrire le développement et les usages des algorithmes* et solutions d'intelligence artificielle dans une démarche éthique pérenne. La présente charte en est le socle fondateur, résultant d'un travail collaboratif et consultatif. Elle présente des engagements qui, pris dans leur ensemble, permettent de garantir un cadre de confiance, respectueux des valeurs de Pôle emploi, et de minimiser les risques liés au déploiement de ces technologies. Les engagements sont regroupés selon les principaux enjeux éthiques soulevés par l'IA à Pôle emploi.

Bien qu'élaborée spécifiquement pour encadrer le développement et l'usage des algorithmes et de l'intelligence artificielle, cette charte s'inscrit dans le contexte global de Pôle emploi et dans le sillon de ses engagements en matière de responsabilité sociale et environnementale. Elle s'appuie notamment sur :

- **Le projet stratégique 2019-2022 de Pôle emploi et ses engagements en matière de Responsabilité Sociale et Environnementale ;**
- **Les travaux en cours sur l'éthique chez Pôle emploi ;**
- **La charte de Pôle emploi pour un numérique responsable (INR) ;**
- **Les engagements en matière de diversité ;**
- **Les engagements en matière de sécurité du système d'information (SI).**

La présente charte concerne tous les algorithmes, services et solutions d'intelligence artificielle dont la conception, la définition, la mise en oeuvre, l'usage, la finalité et les règles sont définies par Pôle emploi et relèvent de sa responsabilité sauf dispositions règlementaires ou législatives liés à des traitements spécifiques.

(*) Les termes comportant un astérisque font l'objet d'une définition dans le glossaire en fin de document.

Nos engagements pour garantir le développement et l'utilisation éthique des algorithmes et de l'intelligence artificielle

1. Finalité et légitimité des algorithmes

L'utilisation d'algorithmes et d'intelligence artificielle doit permettre à Pôle emploi de remplir ses missions, dans le respect de ses valeurs et obligations en tant que service public. La finalité et la légitimité des algorithmes doivent donc être encadrées, et ce d'autant plus lorsqu'un traitement informatique s'applique à des données personnelles.

- Pôle emploi s'engage à ne mettre en oeuvre que des algorithmes et solutions d'intelligence artificielle conçus pour :
  → servir les intérêts individuels légitimes des agents et usagers de Pôle emploi, dans la stricte limite des missions de l'organisme et de ses obligations en tant qu'employeur ;
  → ou lutter contre des pratiques malveillantes à l'encontre des agents et usagers de Pôle emploi, de l'organisme ou de ses financeurs.

2. L'humain au centre ; l'intelligence artificielle au service de l'humain

Il est indispensable que l'intelligence artificielle demeure au service de l'humain. Il s'agira ainsi pour Pôle emploi de garder les agents comme les usagers de Pôle emploi au centre, de veiller à l'inclusion de tous, de préserver la possibilité d'un dialogue humain entre les utilisateurs* et Pôle emploi, et d'apporter les connaissances nécessaires à la compréhension et à l'utilisation de ces technologies.

- Pôle emploi s'engage à utiliser les algorithmes et l'intelligence artificielle pour améliorer la délivrance de ses services, et notamment accompagner ses agents dans la réalisation de leurs tâches (apport d'informations et de connaissances, amélioration des conditions de travail, etc.).
- Pôle emploi s'engage à mettre à disposition des utilisateurs des moyens de solliciter l'intervention d'un agent Pôle emploi, en vue notamment d'obtenir une explication sur le fonctionnement d'un service ou une aide à la décision et de corriger les éventuels biais* algorithmiques.
- Pôle emploi s'engage à mettre en oeuvre des actions d'accompagnement et de sensibilisation des agents et des usagers à l'intelligence artificielle, ainsi qu'à réunir les conditions de confiance, dont la présente charte est garante.

(*) Les termes comportant un astérisque font l'objet d'une définition dans le glossaire en fin de document.

3. Équité et non-discrimination

Un des risques connus des algorithmes et des systèmes d'intelligence artificielle est de pouvoir reproduire, renforcer ou générer des biais discriminatoires. C'est pourquoi il est essentiel de mettre en oeuvre des mesures visant à garantir l'équité entre les individus, l'absence de discriminations et la fiabilité des solutions proposées.

- Pôle emploi s'engage à limiter les biais algorithmiques en vue d'un traitement équitable entre tous les utilisateurs.
- Pôle emploi s'engage à maintenir son niveau de qualité de service ou à le surpasser tout en limitant l'effort demandé aux utilisateurs pour l'entrainement des services d'intelligence artificielle [1].
- Pôle emploi s'engage à surveiller et à maintenir la qualité des données dans le temps pour garantir la stabilité de la performance des services proposés à ses utilisateurs.

4. Liberté de choix

Les algorithmes et solutions d'intelligence artificielle doivent être développés et utilisés dans le respect de l'autonomie humaine. Ils doivent pouvoir aider les individus à prendre des décisions et éclairer leurs choix, sans pour autant ni les y contraindre ni éluder leur responsabilité. À cet égard, il est essentiel pour Pôle emploi de maintenir la place de l'humain dans les processus et de limiter à des tâches routinières qui n'engagent pas la responsabilité humaine les décisions fondées exclusivement sur un traitement automatisé.

- Pôle emploi s'engage, lorsqu'une décision est prise par un algorithme, à mettre en oeuvre les conditions permettant aux utilisateurs d'exposer leurs éventuels recours auprès d'un interlocuteur humain et l'instruction de ce recours par un humain.
- Pôle emploi s'engage à respecter la liberté de choix des utilisateurs lors de la mise en oeuvre de services ayant recours à des algorithmes d'aide à la décision (possibilité de ne pas suivre les recommandations fournies, etc.).

5. Transparence

Dans un souci de transparence envers les usagers, Pôle emploi estime qu'il est indispensable de signaler l'utilisation d'un service basé sur un algorithme d'intelligence artificielle et d'être en capacité d'expliquer de la façon la plus compréhensible possible son fonctionnement et ses résultats.

- Pôle emploi s'engage à informer les utilisateurs des usages qui sont faits de leurs données et à recueillir leur consentement éclairé lorsque celui-ci s'avère nécessaire du point de vue de la loi.
- Pôle emploi s'engage à expliquer les algorithmes aux utilisateurs de la façon la plus compréhensible possible, en prenant en compte ces exigences dès la phase de conception des services et solutions d'intelligence artificielle ; les explications préciseront les grands principes de fonctionnement, les données utilisées et les facteurs déterminants dans les résultats en sortie d'algorithme.
- Pôle emploi s'engage à informer les utilisateurs lorsqu'ils sont en interaction avec un service automatique ou un système d'intelligence artificielle.
- Pôle emploi s'engage à intégrer, dans son rapport annuel, un bilan sur les travaux menés en lien avec les algorithmes et l'intelligence artificielle.

1 Un algorithme, par nature, a besoin des retours utilisateurs pour s'améliorer. Cependant, nous ne mettrons pas en production généralisée une solution demandant un effort trop important pour atteindre des résultats probants.
(*) Les termes comportant un astérisque font l'objet d'une définition dans le glossaire en fin de document.

6. Sécurité

La sécurité et la robustesse technique sont essentielles pour une utilisation en confiance des algorithmes et des technologies d'intelligence artificielle. Il s'agit aussi bien de garantir le respect de la vie privée des usagers tout au long du cycle de vie des solutions, que de s'assurer de la fiabilité et de la résilience aux attaques des services proposés.

- Pôle emploi s'engage à respecter les bonnes pratiques préconisées par l'ANSSI [2] et la CNIL [3] en matière de sécurisation des données : identification des données à risque, sécurisation des traitements, notification en cas de violation des données à caractère personnel.
- Pôle emploi s'engage à identifier les personnes susceptibles de recourir à l'un des services proposés et à garantir les rôles et les droits adaptés à leurs activités, leurs besoins et leurs obligations.
- Pôle emploi s'engage à attacher une attention particulière et une vigilance renforcée à la sécurité, en particulier à ne pas communiquer des données à caractère personnel aux utilisateurs externes à Pôle emploi si le besoin ne s'avère pas conforme à la loi.
- Pôle emploi s'engage à s'employer à éviter toute attaque ou tentative de détournement des services basés sur de l'intelligence artificielle.

7. Impact environnemental

Conscient des enjeux sociaux et environnementaux du numérique, Pôle emploi a adopté une démarche globale pour un numérique responsable et durable qui concourt à l'atteinte de ses objectifs RSE en matière de réduction des émissions carbone. Le développement et l'utilisation d'algorithmes et de systèmes d'intelligence artificielle doivent s'inscrire dans cette démarche, en se faisant de la façon la plus respectueuse de l'environnement possible.

- Pôle emploi s'engage à poursuivre sa démarche pour un numérique responsable et ainsi minimiser l'impact environnemental lié au développement et à l'utilisation d'algorithmes et de systèmes d'intelligence artificielle.

Évolution de la charte

Les technologies évoluant rapidement, il sera essentiel pour Pôle emploi de veiller à l'évolution de ses engagements éthiques dans le temps.
Pôle emploi s'engage à revisiter périodiquement la présente charte et, le cas échéant, à la faire évoluer afin de la mettre en cohérence avec les évolutions technologiques, sociétales et règlementaires.

Mise en oeuvre de la charte

La présente charte éthique de Pôle emploi pour les algorithmes et l'intelligence artificielle constitue un cadre de référence qui définit les engagements que Pôle emploi entend respecter dans ce domaine. La mise en oeuvre effective de ces principes requiert la mobilisation de moyens adaptés.
Pôle emploi s'engage à mettre en oeuvre la gouvernance, l'organisation, les dispositifs et outils nécessaires pour garantir le respect des engagements de la présente charte.

2 Agence nationale de la sécurité des systèmes d'information
3 Commission nationale de l'informatique et des libertés
(*)Les termes comportant un astérisque font l'objet d'une définition dans le glossaire en fin de document.

Glossaire

AGENTS

Les agents sont l'ensemble des personnels employés par Pôle emploi (au sens « agents de l'Etat »).

ALGORITHME

Un algorithme décrit une séquence d'étapes à suivre pour résoudre un problème, de façon suffisamment précise pour être transcrite en un programme pouvant être mis en oeuvre par un ordinateur.
Selon la nature du problème, le résultat peut être unique (effectuer une multiplication par exemple) ou comporter plusieurs réponses (proposer plusieurs pages web en réponse à une recherche par exemple).
Le mot algorithme vient du nom d'un mathématicien perse du IXe siècle, Al-Khwârizmî.
La science qui étudie les propriétés des algorithmes est l'algorithmique.

BIAIS ALGORITHMIQUES

Appliqué au contexte d'un algorithme, un biais peut advenir dans le cas où les données fournies, leur entrainement ou la construction de l'algorithme lui-même sont susceptibles de générer des erreurs, des approximations ou des discriminations.
Exemple : Un algorithme cherchant le candidat attendu pour un poste donné est entrainé avec un échantillon de CV reflétant les effectifs majoritairement masculins associés à ce type de poste. Il est possible, sans opération de débiaisement, que l'algorithme favorise en reflet les candidatures masculines.

EQUITÉ DE TRAITEMENT OU TRAITEMENT ÉQUITABLE

L'équité de traitement vise à prendre en compte la situation singulière de chacun afin de proposer les décisions les plus justes.
Exemple : Un demandeur d'emploi très éloigné de l'emploi se verra proposer un accompagnement renforcé et différentié en accord avec sa situation pour faciliter son retour à l'emploi.

ÉTHIQUE

L'éthique est la discipline philosophique de réflexion sur les fondements des règles de conduite et leur mise en oeuvre selon les contextes. Elle se distingue de la règlementation, qui établit des normes, et du droit qui édicte des lois et aspire à les faire respecter.

INTELLIGENCE ARTIFICIELLE

L'intelligence artificielle est un ensemble de théories et de techniques mises en oeuvre pour réaliser des logiciels capables de simuler des fonctions cognitives, utilisées à des fins de résolution de problèmes, d'assistance ou de substitution à des activités humaines.
Exemple : L'intelligence artificielle peut être utilisée pour « lire » des documents et en extraire les informations clé afin de pré-remplir les champs d'un formulaire qu'un humain devra ensuite contrôler et valider.

USAGERS

Les usagers sont l'ensemble des bénéficiaires des services de Pôle emploi : personnes à la recherche d'information, d'un emploi – qu'elles disposent ou non déjà d'un emploi, recruteurs mais aussi partenaires, etc.

UTILISATEURS

Ensemble des agents et usagers de Pôle emploi susceptibles d'être en présence d'une solution d'intelligence artificielle ou de l'utiliser.

RÉPUBLIQUE
FRANÇAISE
Liberté
Égalité
Fraternité

pôle emploi

Pôle emploi - Direction générale
Mise en page : © STUDIO MAQUETTE - Avril 2022

A.2 Analyse des jeux de données

A.2.1 Données de test du modèle

Ce jeu de données permet de tester la performance du modèle à attention entraîné. Ses données sont réparties de manière similaire au jeu d'entraînement. Le détail de la répartition est donné dans la figure A.1.

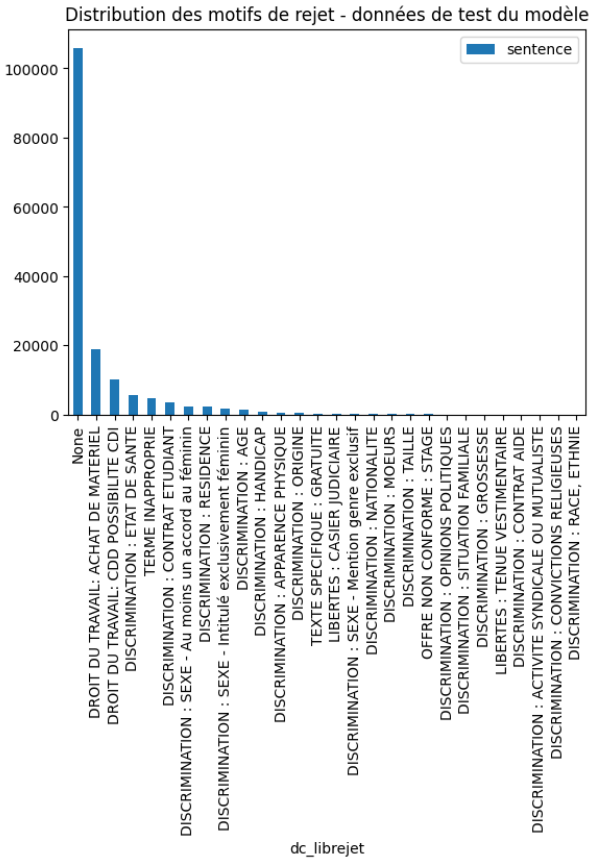


FIGURE A.1 – Distribution de l'ensemble des classes du jeu de données de test du modèle à attention pour le cas d'usage LEGO.

Sur ce jeu de données, le modèle possède une précision de 83.67%. La matrice de confusion présentée en figure A.2 montre le détail de ces performances par classe. Il en ressort que la plupart des phrases légales sont considérées comme telles. À l'inverse, les classes très peu représentées telles que *Discrimination : race, Ethnie* ou *Discrimination : convictions religieuses* ne sont pas correctement détectées. La matrice de confusion montre que le modèle effectue de nombreux faux négatifs : la colonne de classe prédite *None*

correspond aux phrases prédites légales. Le modèle tend à classer légales des phrases qui ne le sont pas. La définition des phrases légales est faite par la négation : ce sont les phrases qui ne comportent pas de motif de rejet. De même, la distribution présentée en figure A.1 montre que cette classe est sur-représentée. Ces éléments peuvent expliquer le fort taux de faux négatifs.

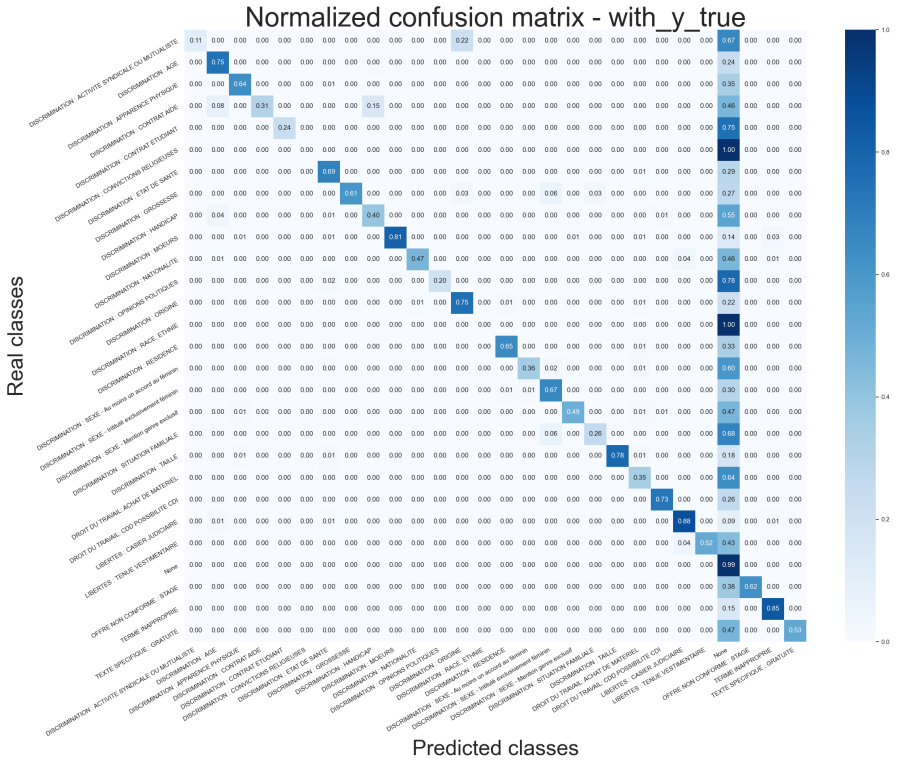


FIGURE A.2 – Distribution de l'ensemble des classes du jeu de données de test du modèle à attention pour le cas d'usage LEGO.

A.2.2 Données de test LEGO - BP

Ce jeu de données permet de comparer les explications générées dans le cas où le modèle effectue une classification correcte. Cela entraine une forte proportion de phrases légales, visible dans la figure A.3. Les phrases non légales sont au nombre de 16 dans ce jeu de données.

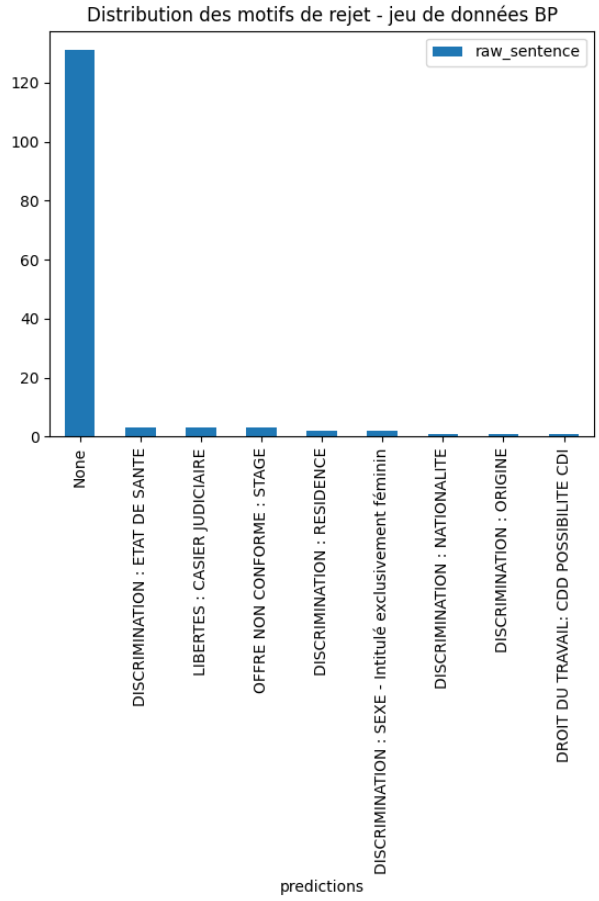


FIGURE A.3 – Distribution de l'ensemble des classes du jeu de données LEGO - BP.

A.2.3 Données de test LEGO - DE

Ce jeu de données permet de comparer les explications générées dans le cas où les explications ne sont pas toutes identiques entre les explications de référence et celles générées par les méthodes des ancres et de l'attention. Cela implique l'absence de phrases légales, pour lesquelles les explications sont vides donc identiques. La distribution des classes est illustrée en figure A.4. Ce jeu de données est spécifiquement adapté à la collecte de la préférence des utilisateurs.

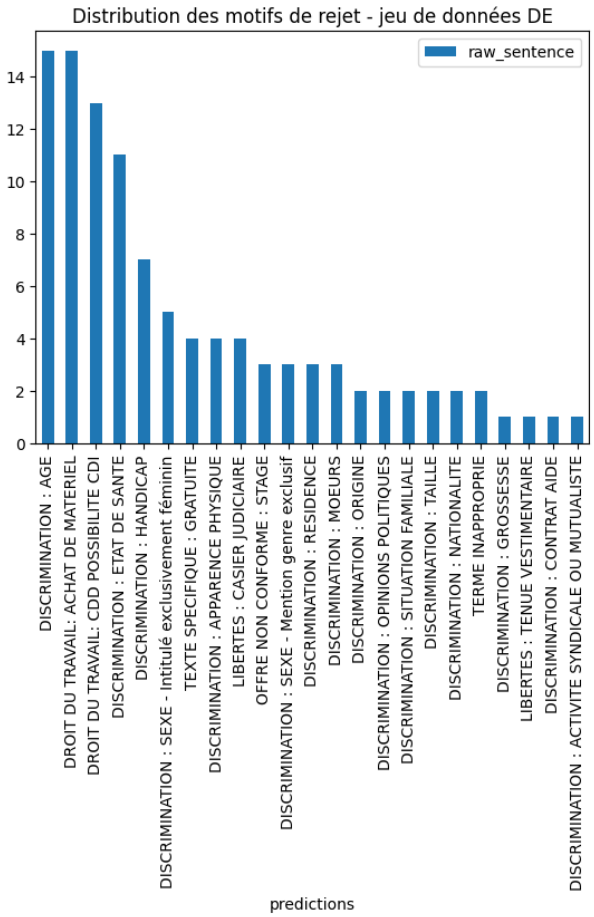


FIGURE A.4 – Distribution de l'ensemble des classes du jeu de données LEGO - DE.

Publications de l'autrice

[54] Gaëlle Jouis et al., « Anchors vs Attention : Comparing XAI on a Real-Life Use Case », in : *Pattern Recognition. ICPR International Workshops and Challenges*, Virtuel : Springer International Publishing, 2021, p. 219-227, ISBN : 978-3-030-68796-0.

[55] Gaëlle Jouis et al., « Ancres vs Attention : comparaison de méthodes d'explicabilité des réseaux profonds sur un cas d'usage réel », in : *21èmes Journées Francophones Extraction et Gestion des Connaissances (EGC 2021) Atelier "DL for NLP : Deep Learning pour le traitement automatique des langues"*, Montpellier, France, jan. 2021.

[56] Gaëlle Jouis et al., « Tour d'horizon autour de l'explicabilité des modèles profonds », in : *Rencontres des Jeunes Chercheur · ses en Intelligence Artificielle (RJCIA)*, Angers, France, 2020.

À paraître

[53] Gaëlle Jouis et al., « A Methodology to Compare XAI Explanations on Natural Language Processing », in : *Explainable Deep Learning AI Methods and Challenges*, sous la dir. de Dragutin Petkovic Jenny Benois-Pineau Romain Bourqui, Elsevier, 2023.

Bibliographie

[1] Samira Abnar et Willem Zuidema, « Quantifying Attention Flow in Transformers », in : *The 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, mai 2020.

[2] Julius Adebayo et al., « Sanity checks for saliency maps », in : *Advances in neural information processing systems* 31 (2018).

[3] Chirag Agarwal et Anh Nguyen, « Explaining image classifiers by removing input features using generative models », in : *Proceedings of the Asian Conference on Computer Vision*, 2020.

[6] Alejandro Barredo Arrieta et al., « Explainable Artificial Intelligence (XAI) : Concepts, taxonomies, opportunities and challenges toward responsible AI », in : *Information Fusion* 58 (2020), p. 82-115.

[7] Meghna P Ayyar, Jenny Benois-Pineau et Akka Zemmari, « Review of white box methods for explanations of convolutional neural networks in image classification tasks », in : *Journal of Electronic Imaging* 30.*5* (2021), p. 050901.

[8] Sebastian Bach et al., « On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation », in : *PloS one* 10.*7* (2015), e0130140.

[9] David Baehrens et al., « How to Explain Individual Classification Decisions », in : *J. Mach. Learn. Res.* 11 (2010), p. 1803-1831, issn : 1532-4435.

[10] Dzmitry Bahdanau, Kyunghyun Cho et Yoshua Bengio, « Neural Machine Translation by Jointly Learning to Align and Translate », in : *3rd International Conference on Learning Representations, ICLR, San Diego, CA, USA*, sous la dir. d'Yoshua Bengio et Yann LeCun, 2015.

[11] Yujia Bao et al., « Deriving Machine Attention from Human Rationales », in : *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium : Association for Computational Linguistics, oct. 2018, p. 1903-1913, doi : 10.18653/v1/D18-1216.

[12] CJ Barberan, Randall Balestriero et Richard G Baraniuk, « NeuroView : Explainable Deep Network Decision Making », in : *Diss., Rice University* (2022).

[13] Alina Jade Barnett et al., « A case-based interpretable deep learning model for classification of mass lesions in digital mammography », in : *Nature Machine Intelligence* 3.*12* (2021), p. 1061-1070.

[14] David Bau et al., « Network dissection : Quantifying interpretability of deep visual representations », in : *Proc of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2017, p. 6541-6549.

[15] Valérie Beaudouin et al., « Flexible and context-specific AI explainability : a multidisciplinary approach », in : *Available at SSRN 3559477* (2020).

[16] Ralph Allan Bradley et Milton E Terry, « Rank analysis of incomplete block designs : I. The method of paired comparisons », in : *Biometrika* 39.*3/4* (1952), p. 324-345.

[17] Chun-Hao Chang et al., « Explaining Image Classifiers by Counterfactual Generation », in : *7th International Conference on Learning Representations, ICLR, New Orleans, LA, USA*, OpenReview.net, 2019.

[18] Martin Charachon et al., « Combining similarity and adversarial learning to generate visual explanation : Application to medical image classification », in : *25th International Conference on Pattern Recognition (ICPR)*, IEEE, 2021, p. 7188-7195.

[19] Aditya Chattopadhay et al., « Grad-cam++ : Generalized gradient-based visual explanations for deep convolutional networks », in : *IEEE winter conference on applications of computer vision (WACV)*, IEEE, 2018, p. 839-847.

[20] Zhengping Che et al., « Distilling knowledge from deep networks with applications to healthcare domain », in : *arXiv preprint arXiv :1512.03542* (2015).

[21] Gabriele Ciravegna et al., « A Constraint-Based Approach to Learning and Explanation », in : t. 34, 04, Association for the Advancement of Artificial Intelligence (AAAI), avr. 2020, p. 3658-3665, doi : 10.1609/aaai.v34i04.5774.

[22] Noel CF Codella et al., « TED : Teaching AI to Explain its Decisions », in : *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, 2019, p. 123-129.

[23] European Commission, *2018 reform of EU data protection rules*, 25 mai 2018, (visité le 25/10/2022).

[24] Felipe Costa et al., « Automatic generation of natural language explanations », in : *Proceedings of the 23rd International Conference on Intelligent User Interfaces Companion*, 2018, p. 1-2.

[25] Hoa Khanh Dam, Truyen Tran et Aditya Ghose, « Explainable software analytics », in : *Proceedings of the 40th International Conference on Software Engineering : New Ideas and Emerging Results*, Gothenburg, Sweden, 2018, p. 53-56.

[26] Abhishek Das et al., « Human attention in visual question answering : Do humans and deep networks look at the same regions ? », in : *Computer Vision and Image Understanding* 163 (2017), p. 90-100.

[27] A. Datta, S. Sen et Y. Zick, « Algorithmic Transparency via Quantitative Input Influence : Theory and Experiments with Learning Systems », in : *IEEE Symposium on Security and Privacy (SP)*, mai 2016, p. 598-617, doi : 10.1109/SP.2016.42.

[28] Sonia Desmoulin-Canselier et Daniel Le Métayer, « Algorithmic Decision Systems in the Health and Justice Sectors : Certification and Explanations for Algorithms in European and French Law », in : *European Journal of Law and Technology* 9.*3* (2019).

[29] Jay DeYoung et al., « Eraser : A benchmark to evaluate rationalized nlp models », in : *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (juil. 2019), p. 4443-4458.

[30] Amit Dhurandhar et al., « Tip : Typifying the interpretability of procedures », in : *arXiv preprint arXiv :1706.02952* (2017).

[31] Jonathan Dodge et Margaret Burnett, « Position : We Can Measure XAI Explanations Better with Templates. », in : *ExSS-ATEC@ IUI*, 2020.

[32] Ann-Kathrin Dombrowski et al., « Explanations can be manipulated and geometry is to blame », in : *Advances in Neural Information Processing Systems*, sous la dir. de H. Wallach et al., t. 32, Vancouver, Canada : Curran Associates, Inc., 2019, url : https://proceedings.neurips.cc/paper/2019/file/bb836c01cdc9120a9c984c525e4b1a4a-Paper.pdf.

[33] Finale Doshi-Velez et Been Kim, « Towards a rigorous science of interpretable machine learning », in : *arXiv preprint arXiv :1702.08608* (2017).

[34] Finale Doshi-Velez et al., « Accountability of AI under the law : The role of explanation », in : *Berkman Klein Center Working Group on Explanation and the Law, Berkman KleinCenter for Internet & Society working paper* (2017).

[35] Huanming Fang et Hui Miao, *Introducing the Model Card Toolkit for Easier Model Transparency Reporting*, juil. 2020.

[36] Jun-Peng Fang et al., « Interpreting Model Predictions with Constrained Perturbation and Counterfactual Instances », in : *International Journal of Pattern Recognition and Artificial Intelligence* (2021), p. 2251001.

[37] Reza Ghaeini et al., « Interpreting Recurrent and Attention-Based Neural Models : a Case Study on Natural Language Inference », in : *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, août 2018, p. 4952-4957, doi : 10.18653/v1/D18-1537.

[38] Marzyeh Ghassemi, Luke Oakden-Rayner et Andrew L Beam, « The false hope of current approaches to explainable artificial intelligence in health care », in : *The Lancet Digital Health* 3.*11* (2021), e745-e750.

[39] Leilani H Gilpin et al., « Explaining explanations : An overview of interpretability of machine learning », in : *IEEE 5th International Conference on data science and advanced analytics (DSAA)*, IEEE, 2018, p. 80-89.

[40] Tristan Gomez, Thomas Fréour et Harold Mouchère, « Comparison of attention models and post-hoc explanation methods for embryo stage identification : a case study », in : *XAI Workshop (ICPR)*, Montréal, Canada, août 2022, url : https://hal.archives-ouvertes.fr/hal-03690574.

[41] Tristan Gomez, Thomas Fréour et Harold Mouchère, « Metrics for saliency map evaluation of deep learning explanation methods », in : *International Conference on Pattern Recognition and Artificial Intelligence*, 2022, doi : 10.48550/ARXIV.2201.13291.

[42] Tristan Gomez et al., « BR-NPA : A non-parametric high-resolution attention model to improve the interpretability of attention », in : *Pattern Recognition* 132 (déc. 2022), p. 108927, doi : 10.1016/j.patcog.2022.108927.

[43] Niku Gorji et Sasha Rubin, « Sufficient reasons for classifier decisions in the presence of constraints », in : *The Thirty-Sixth AAAI Conference on Artificial Intelligence* (2022).

[44] Riccardo Guidotti et al., « A survey of methods for explaining black box models », in : *ACM computing surveys (CSUR)* 51.*5* (2018), p. 1-42.

[45] Ramin Hasani et al., *Can a Compact Neuronal Circuit Policy be Re-purposed to Learn Simple Robotic Control ?*, 2018, arXiv : 1809.04423.

[46] Bernease Herman, « The promise and peril of human evaluation for model interpretability », in : *arXiv preprint arXiv :1711.07414* (2017).

[47] Geoffrey Hinton, Oriol Vinyals et Jeff Dean, « Distilling the knowledge in a neural network », in : *arXiv preprint arXiv :1503.02531* (2015).

[48] Robert R Hoffman et al., « Metrics for explainable AI : Challenges and prospects », in : *arXiv preprint arXiv :1812.04608* (2018).

[49] Fred Hohman et al., « Visual analytics in deep learning : An interrogative survey for the next frontiers », in : *IEEE transactions on visualization and computer graphics* 25.*8* (2018), p. 2674-2693.

[50] Rahul Iyer et al., « Transparency and explanation in deep reinforcement learning neural networks », in : *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, 2018, p. 144-150.

[51] Sarthak Jain et Byron C Wallace, « Attention is not explanation », in : *NAACL*, 2019.

[52] Marijn Janssen et al., « Will Algorithms Blind People ? The Effect of Explainable AI and Decision-Makers' Experience on AI-supported Decision-Making in Government », in : *Social Science Computer Review* (2020), doi : 10.1177/0894439320980118.

[57] Ikram Chraibi Kaadoud et al., « Automata-based Explainable Representation for a Complex System of Multivariate Times Series », in : (2022), doi : 10.5220/0011363400003335.

[58] Andrej Karpathy, Justin Johnson et Fei-Fei Li, « Visualizing and Understanding Recurrent Networks », in : *The International Conference on Learning Representations (ICLR)* (2016).

[59] Eoin M Kenny et Mark T Keane, « On Generating Plausible Counterfactual and Semi-Factual Explanations for Deep Learning », in : *Proceedings of the AAAI Conference on Artificial Intelligence 13* (2021), p. 11575-11585.

[60] Been Kim et al., « Tcav : Relative concept importance testing with linear concept activation vectors », in : (2018).

[62] Himabindu Lakkaraju et al., « Faithful and customizable explanations of black box models », in : *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, 2019, p. 131-138.

[63] Yann LeCun, Yoshua Bengio et Geoffrey Hinton, « Deep learning », in : *Nature* 521.*7553* (2015), p. 436-444.

[64] Legifrance, *LOI n 2016-1321 du 7 octobre 2016 pour une Republique numerique*, 2016.

[65] Brian Y Lim, Anind K Dey et Daniel Avrahami, « Why and why not explanations improve the intelligibility of context-aware intelligent systems », in : *Proceedings of the SIGCHI conference on human factors in computing systems*, 2009, p. 2119-2128.

[66] Zhouhan Lin et al., « A Structured Self-Attentive Sentence Embedding », in : *5th International Conference on Learning Representations, ICLR, Toulon, France*, 2017.

[68] Xuan Liu, Xiaoguang Wang et Stan Matwin, « Improving the interpretability of deep neural networks with knowledge distillation », in : *IEEE International Conference on Data Mining Workshops (ICDMW)*, IEEE, 2018, p. 905-912.

[69] Yang Liu et al., « Synthetic Benchmarks for Scientific Research in Explainable Machine Learning », in : *arXiv preprint arXiv :2106.12543* (2021).

[70] R Duncan Luce, *Individual choice behavior : A theoretical analysis*, Courier Corporation, 1959.

[71] Scott M Lundberg et Su-In Lee, « A Unified Approach to Interpreting Model Predictions », in : *Advances in Neural Information Processing Systems 30*, sous la dir. d'I. Guyon et al., Curran Associates, Inc., 2017, p. 4765-4774.

[72] Tim Miller, « Explanation in artificial intelligence : Insights from the social sciences », in : *Artificial Intelligence* 267 (2019), p. 1-38.

[73] Tim Miller, Piers Howe et Liz Sonenberg, « Explainable AI : Beware of inmates running the asylum or : How I learnt to stop worrying and love the social and behavioural sciences », in : *arXiv preprint arXiv :1712.00547* (2017).

[74] Margaret Mitchell et al., « Model cards for model reporting », in : *Proceedings of the conference on fairness, accountability, and transparency*, 2019, p. 220-229.

[75] Sina Mohseni, Jeremy E Block et Eric Ragan, « Quantitative Evaluation of Machine Learning Explanations : A Human-Grounded Benchmark », in : *26th International Conference on Intelligent User Interfaces*, 2021, p. 22-31.

[76] Sina Mohseni, Niloofar Zarei et Eric D Ragan, « A multidisciplinary survey and framework for design and evaluation of explainable AI systems », in : *ACM Transactions on Interactive Intelligent Systems (TiiS)* 11.*3-4* (2021), p. 1-45.

[77] Christoph Molnar, *Interpretable Machine Learning, A Guide for Making Black Box Models Explainable*, `https://christophm.github.io/interpretable-ml-book/`, 2019.

[79] Grégoire Montavon et al., « Explaining nonlinear classification decisions with deep taylor decomposition », in : *Pattern Recognition* 65 (2017), p. 211-222.

[80] Abraham Montoya Obeso et al., « Visual vs internal attention mechanisms in deep neural networks for image classification and object detection », in : *Pattern Recognition* 123 (2022), p. 108411.

[81] Chris Olah et Shan Carter, « Attention and Augmented Recurrent Neural Networks », in : *Distill* (2016), doi : `10.23915/distill.00001`.

[83] Pôle emploi, *Charte de Pôle emploi pour une intelligence artificielle éthique*, 2022.

[84] Pôle emploi, *IE_Ethique_Carto risques éthiques_Atelier 2_V2 post-atelier*, Documentation interne non accessible au public, 2021.

[85] Samuele Poppi et al., « Revisiting The Evaluation of Class Activation Mapping for Explainability : A Novel Metric and Experimental Analysis », in : *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, p. 2299-2304.

[86] Guangwu Qian et al., « Conceptor Learning for Class Activation Mapping », in : *arXiv preprint arXiv :2201.08636* (2022).

[87] Luyu Qiu et al., « Resisting out-of-distribution data problem in perturbation of xai », in : *arXiv preprint arXiv :2107.14000* (2021).

[88] Aditya Ramesh et al., *Hierarchical Text-Conditional Image Generation with CLIP Latents*, 2022, doi : 10.48550/ARXIV.2204.06125.

[89] Peyman Rasouli et Ingrid Chieh Yu, « CARE : coherent actionable recourse based on sound counterfactual explanations », in : *International Journal of Data Science and Analytics* (sept. 2022), doi : 10.1007/s41060-022-00365-6.

[90] Marco Tulio Ribeiro, Sameer Singh et Carlos Guestrin, « "Why Should I Trust You ?" : Explaining the Predictions of Any Classifier », in : *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, San Francisco, California, USA : ACM, 2016, p. 1135-1144, isbn : 978-1-4503-4232-2, doi : 10.1145/2939672.2939778, eprint : 1602.04938.

[91] Marco Túlio Ribeiro, Sameer Singh et Carlos Guestrin, « Anchors : High-Precision Model-Agnostic Explanations », in : *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA*, 2018, p. 1527-1535.

[92] Ronald Richman et Mario V Wuthrich, « LocalGLMnet : interpretable deep learning for tabular data », in : *Available at SSRN 3892015* (2021).

[93] Wojciech Samek, Thomas Wiegand et Klaus-Robert Müller, « Explainable Artificial Intelligence : Understanding, Visualizing and Interpreting Deep Learning Models », in : *ITU Journal : ICT Discoveries - Special Issue 1 - The Impact of Artificial Intelligence (AI) on Communication Networks and Services* 1 (oct. 2017), p. 1-10.

[94] Wojciech Samek et al., « Evaluating the visualization of what a deep neural network has learned », in : *IEEE transactions on neural networks and learning systems* 28.*11* (2016), p. 2660-2673.

[95] Alana de Santana Correia et Esther Luna Colombini, « Attention, please ! A survey of neural attention models in deep learning », in : *Artificial Intelligence Review* 55.*8* (mar. 2022), p. 6037-6124, doi : 10.1007/s10462-022-10148-x.

[97] Udo Schlegel et al., « Towards A Rigorous Evaluation Of XAI Methods On Time Series », in : *IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 2019, p. 4197-4201.

[99] Ramprasaath R. Selvaraju et al., « Grad-CAM : Visual Explanations From Deep Networks via Gradient-Based Localization », in : *The IEEE International Conference on Computer Vision (ICCV)*, oct. 2017.

[100] Avanti Shrikumar, Peyton Greenside et Anshul Kundaje, « Learning Important Features Through Propagating Activation Differences », in : *Proceedings of the 34th International Conference on Machine Learning*, sous la dir. de Doina Precup et Yee Whye Teh, t. 70, Proceedings of Machine Learning Research, PMLR, août 2017, p. 3145-3153.

[101] Karen Simonyan, Andrea Vedaldi et Andrew Zisserman, « Deep Inside Convolutional Networks : Visualising Image Classification Models and Saliency Maps », in : (2014).

[102] Jost Tobias Springenberg et al., « Striving for simplicity : The all convolutional net », in : *3rd International Conference on Learning Representations, ICLR*, sous la dir. d'Yoshua Bengio et Yann LeCun, San Diego, CA, USA, 7 mai 2015.

[103] Fabian Stieler, Fabian Rabe et Bernhard Bauer, « Towards domain-specific explainable AI : model interpretation of a skin image classifier using a human approach », in : *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, p. 1802-1809.

[104] Erik Strumbelj et Igor Kononenko, « An efficient explanation of individual classifications using game theory », in : *Journal of Machine Learning Research* 11.*Jan* (2010), p. 1-18.

[105] Stefano Teso et Kristian Kersting, « Explanatory Interactive Machine Learning », in : *Conference on Artificial Intelligence, Ethics and Society (AIES)*, AAAI, 2019.

[106] Shailja Thakur et Sebastian Fischmeister, « A generalizable saliency map-based interpretation of model outcome », in : *25th International Conference on Pattern Recognition (ICPR)*, IEEE, 2021, p. 4099-4106.

[107] K. Uehara et al., « Multi-Scale Explainable Feature Learning for Pathological Image Analysis Using Convolutional Neural Networks », in : *IEEE International Conference on Image Processing (ICIP)*, 2020, p. 1931-1935, doi : 10.1109/ICIP40778.2020.9190693.

[108] Kazuki UEHARA et al., « Explainable feature embedding using convolutional neural networks for pathological image analysis », in : *25th International Conference on Pattern Recognition (ICPR)*, IEEE, 2021, p. 4560-4565.

[109] Ashish VASWANI et al., « Attention is all you need », in : *Advances in neural information processing systems*, 2017, p. 5998-6008.

[110] Tom VERMEIRE et al., « How to Choose an Explainability Method ? Towards a Methodical Implementation of XAI in Practice », in : *arXiv preprint arXiv :2107.04427* (2021), p. 521-533, DOI : 10.1007/978-3-030-93736-2_39.

[112] Sandra WACHTER, Brent MITTELSTADT et Luciano FLORIDI, « Why a right to explanation of automated decision-making does not exist in the general data protection regulation », in : *International Data Privacy Law* 7.*2* (2017), p. 76-99.

[113] Ben WANG et Aran KOMATSUZAKI, *GPT-J-6B : A 6 Billion Parameter Autoregressive Language Model*, https://github.com/kingoflolz/mesh-transformer-jax, mai 2021.

[114] Yequan WANG et al., « Attention-based LSTM for aspect-level sentiment classification », in : *Proceedings of the conference on empirical methods in natural language processing*, 2016, p. 606-615.

[115] Joseph WEIZENBAUM, « ELIZA—a computer program for the study of natural language communication between man and machine », in : *Communications of the ACM* 9.*1* (1966), p. 36-45.

[116] Sarah WIEGREFFE et Yuval PINTER, « Attention is not not Explanation », in : *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China : Association for Computational Linguistics, nov. 2019, p. 11-20, DOI : 10.18653/v1/D19-1002.

[117] Kanit WONGSUPHASAWAT et al., « Visualizing Dataflow Graphs of Deep Learning Models in TensorFlow », in : *IEEE Transactions on Visualization and Computer Graphics*, 2017.

[118] Huijun WU et al., « Interpreting Shared Deep Learning Models via Explicable Boundary Trees », in : *CoRR* abs/1709.03730 (2017), arXiv : 1709.03730.

[119] Zhengyuan YANG et al., « Action recognition with spatio–temporal visual attention on skeleton image sequences », in : *IEEE Transactions on Circuits and Systems for Video Technology* 29.*8* (2018), p. 2405-2415.

[120] Zeyu YUN et al., « Transformer visualization via dictionary learning : contextualized embedding as a linear superposition of transformer factors », in : *arXiv preprint arXiv :2103.15949* (2021).

[121] Muhammad Bilal ZAFAR et al., « More Than Words : Towards Better Quality Interpretations of Text Classifiers », in : *arXiv preprint arXiv :2112.12444* (2021).

[122] Muhammad Bilal ZAFAR et al., « On the Lack of Robust Interpretability of Neural Text Classifiers », in : *The Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP)* (2021).

[123] Matthew D ZEILER et Rob FERGUS, « Visualizing and understanding convolutional networks », in : *European conference on computer vision*, Springer, 2014, p. 818-833.

[124] Bolei ZHOU et al., « Learning deep features for discriminative localization », in : *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, p. 2921-2929.

COLLEGE DOCTORAL PAYS DE LA LOIRE ED MATHEMATIQUES STIC

Nantes Université

Titre : Explicabilité des modèles profonds et méthodologie pour son évaluation : application aux données textuelles de Pôle emploi

Mot clés : Apprentissage profond, Explicabilité, Réseaux de Neurones, Intelligence Artificielle

Résumé : L'intelligence Artificielle fait partie de notre quotidien. Les modèles développés sont de plus en plus complexes. Les régulations telles que la Loi Pour une République Numérique orientent les développements logiciels vers plus d'éthique et d'explicabilité. Comprendre le fonctionnement des modèles profonds a un intérêt technique et humain. Les solutions proposées par la communauté sont nombreuses, et il n'y a pas de méthode miracle répondant à toutes les problématiques. Nous abordons la question suivante : comment intégrer l'explicabilité dans un projet d'IA basé sur des techniques d'apprentissage profond ?

Après un état de l'art présentant la richesse de la littérature du domaine, nous présentons le contexte et les prérequis de nos travaux. Ensuite nous présentons un protocole d'évaluation d'explications locales et une méthodologie modulaire de caractérisation globale du modèle. Enfin, nous montrons que nos travaux sont intégrés à leur environnement industriel.

Ces travaux résultent en l'obtention d'outils concrets permettant au lecteur d'appréhender la richesse des outils d'explicabilité à sa disposition.

Title: Explainability of deep models and methodology for its evaluation: application to textual data from Pôle emploi

Keywords: Deep learning, Explainability, Neural Networks, Artificial Intelligence

Abstract: Artificial intelligence is part of our daily life. The models developed are more and more complex. Regulations such as the French Law for a Digital Republic (Loi Pour une République Numérique) are directing software development towards more ethics and explainability. Understanding the functioning of deep models is of technical and human interest. The solutions proposed by the community are numerous, and there is no miracle method that answers all the problems. We address the following question: how to integrate explainability in an AI project based on deep learning techniques?

After a state of the art presenting the richness of the literature in the field, we present the context and prerequisites for our work. Then we present a protocol for evaluating local explanations and a modular methodology for global model characterization. Finally, we show that our work is integrated into its industrial environment.

This work results in concrete tools allowing the reader to apprehend the richness of the explicability tools at their disposal.