HAL open science

Un (petit) pas vers l'explicabilité de l'intelligence artificielle

Romain Giot

To cite this version:

Romain Giot. Un (petit) pas vers l'explicabilité de l'intelligence artificielle. Informatique [cs]. Université de Bordeaux, 2024. tel-04463271

HAL Id: tel-04463271

https://theses.hal.science/tel-04463271

Submitted on 16 Feb 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Copyright

Thèse présentée

pour obtenir le diplôme d'

Habilitation à Diriger des Recherches

École Doctorale Mathématiques et Informatique

informatique

Par **Romain Giot**

Un (petit) pas vers l'explicabilité de l'intelligence artificielle

Soutenue le 8 février 2024

Membres du jury :

M. David Auber	Professeur des universités	Université de Bordeaux	Garant
M. Baptiste Caramiaux	Chargé de Recherche	CNRS ISIR	Examinateur
M. Pierre Dragicevic	Chargé de Recherche	INRIA Bordeaux	Examinateur
Mme. Céline Hudelot	Professeure des universités	CentraleSupélec	Rapportrice
M. Christophe Hurter	Professeur des universités	École Nationale de l'Aviation Civile	Examinateur Président
M. Guy Mélançon	Professeur des universités	Université de Bordeaux	Invité
M. Alexandru-Cristian Telea	Full professor	Université d'Utrecht	Rapporteur
M. Gilles Venturini	Professeur des universités	Université de Tours	Rapporteur

Habilitation à diriger des recherches

Un (petit) pas vers l'explicabilité de l'intelligence artificielle

Romain Giot

Université de Bordeaux

Pour mes enfants : Rose, Léo et Lisa

Remerciements

L'écriture de ce manuscrit d'habilitation à diriger des recherches est la conséquence d'une passion pour la résolution de défis, autant à titre personnel que professionnel, qui date de plus de deux décennies. J'ai naturellement plusieurs personnes à remercier de m'avoir permis d'en arriver là.

Olivier F., Yves, Amaury, pour m'avoir enseigné, lorsque j'étais jeune lycéen, l'assembleur Z80, les spécificités matérielles de l'Amstrad CPC, les défis du démomaking, et l'envie du dépassement des contraintes techniques. Olivier G., Édouard, Philippe, David S., David T., Franck, Julien, et tant d'autres pour être encore capables d'innover sur cette plateforme désuète, même 40 ans après son invention. Depuis plus de 20 ans, j'essaie de vous surpasser, espérons que j'y arrive avant les 20 prochaines années.

Christophe Rosenberger et Bernadette Dorizzi pour m'avoir formé, donné goût à la recherche, et à la rigueur scientifique, une autre forme de dépassement.

Romain Bourqui pour m'avoir aidé à la fois à m'intégrer dans l'équipe MaBIOVIS et à monter en compétence rapidement en visualisation d'information. Sans lui, j'aurai progressivement délaissé la recherche en échouant à faire mon changement de thématique. David Auber pour m'avoir également aidé à monter rapidement en compétences en visualisation d'information, et d'avoir accepté d'être mon garant. Espérons qu'il ne regrette pas maintenant ;) L'équipe BKB du LaBRI pour être capable de travailler ensemble malgré l'hétérogénéité des domaines de recherche de ses membres.

Nicholas Journet pour m'avoir très rapidement aidé à m'intégrer au département informatique de l'IUT de Bordeaux. L'équipe du département informatique de l'IUT de Bordeaux pour faire en sorte d'avoir une formation qui tourne malgré le manque criant de moyens humains (cela dit, est-ce bien raisonnable ?).

Guy Mélançon, David Auber, Nicholas Journet, André CPLF de Carvalho, Jenny Benois-Pineau pour m'avoir accompagné dans la direction ou l'encadrement des étudiants qui sont à l'origine de beaucoup de résultats présentés, ou pas, dans ce manuscrit. Adrien, Alexey, Frédéric, Korlan, Loann, Luc-Etienne, Paulo H., Rémi pour leurs contributions significatives à ces résultats.

L'intégralité des membres du jury de cette HDR pour l'intérêt pour mon travail, et leurs longs échanges durant sa soutenance.

Et pour finir, mes parents et mes frères pour avoir su être présents au bon moment.

Table des matières

Acronymes . . . 1

I. Introduction 3

Chapitre 1. Introduction générale . . . 5

II. Analyse de données et apprentissage automatique 9

Chapitre 2. Introduction à l'analyse de données et l'apprentissage automatique . . . 13

Chapitre 3. Authentification biométrique . . . 15
3.1. Introduction à l'authentification biométrique . . . 15
3.1.1. Généralités sur la biométrie . . . 15
3.1.2. Modalités biométriques . . . 16
3.1.3. Évaluation de systèmes d'authentification biométrique . . . 17
3.1.4. Stratégies d'amélioration des performances des systèmes d'authentification biométriques . . . 17
3.1.5. La dynamique de frappe au clavier : une biométrie comportementale . . . 18
3.2. Mise à jour de la référence biométrique : une nécessité pour les biométries comportementales . . . 19
3.2.1. Contexte . . . 19
3.2.2. Méthode améliorée de mise à jour . . . 19
3.3. Diminution de la quantité de captures pour l'enrôlement en dynamique de frappe au clavier . . . 20
3.3.1. Contexte . . . 20
3.3.2. Réseau siamois pour la dynamique de frappe au clavier . . . 21

Chapitre 4. Attribution de caractéristiques pour les réseaux de neurones profonds . . . 23
4.1. Introduction à l'explicabilité pour l'explication des réseaux de neurones profonds . . . 23
4.1.1. Attribution de caractéristiques . . . 23
4.1.2. Détection de caractéristiques apprises . . . 24
4.1.3. Explication par exemple factuel . . . 25
4.1.4. Explication par contre-exemple . . . 25
4.1.5. Extraction du comportement du modèle . . . 25
4.2. Attribution de caractéristiques pour réseau de neurones sans prérequis de classe . . . 25
4.2.1. Explication de caractéristiques avec FEM . . . 26
4.2.2. Amélioration de la résolution de l'explication avec ML-FEM . . . 27

Chapitre 5. Discussion sur l'analyse de données et l'intelligence artificielle . . . 31
5.1. Authentification biométrique . . . 31
5.1.1. Mise à jour de la référence biométrique . . . 31
5.1.2. Adaptation aux réseaux de neurones profonds . . . 31
5.1.3. De l'acceptabilité de l'authentification biométrique . . . 31
5.2. Attribution de caractéristiques pour les réseaux de neurones profonds . . . 32
5.2.1. De la faible résolution des explications . . . 32

5.2.2. Évaluation des méthodes d'explication . . . 32
5.3. Travaux non abordés . . . 32
5.3.1. Détection de fraudes dans les systèmes de paiement mobile . . . 32
5.3.2. Détection de panne dans les systèmes de vélos en libre-service . . . 33
5.3.3. Segmentation d'images satellites . . . 33
5.3.4. Classification hiérarchique et multimodale . . . 33

Chapitre 6. Bibliographie de l'analyse de données et l'apprentissage automatique . . . 35

III. Visualisation de masses de données 39

Chapitre 7. Introduction à la visualisation de masses de données . . . 43

Chapitre 8. Dessin de grands graphes . . . 47
8.1. Introduction au dessin de grands graphes . . . 47
8.1.1. Définitions utiles pour la suite . . . 47
8.1.2. Dessin de graphes . . . 48
8.1.3. Faisceautage d'arêtes . . . 49
8.2. Utilisation de la dégénérescence pour le dessin multiniveaux de grands graphes . . . 50
8.2.1. Présentation de HCBL . . . 50
8.2.2. Résultats de HCBL . . . 52
8.3. Faisceautage multi-échelle d'arêtes moins bruité . . . 53
8.3.1. Présentation de la méthode de faisceautage hiérarchique . . . 53
8.3.2. Evaluation de la méthode de faisceautage hiérarchique . . . 54

Chapitre 9. Représentation de masses de données tabulaires dans une grille . . . 55
9.1. Introduction à la représentation de masses de données tabulaires . . . 55
9.1.1. Contexte . . . 55
9.1.2. Visualisation de données en espace contraint . . . 55
9.2. Projection de données dans une grille : Voronoi Relaxation Grid . . . 57
9.2.1. Description de VRGRid . . . 57
9.2.2. Évaluation de VRGRid . . . 59

Chapitre 10. Discussion sur la visualisation de masses de données . . . 61
10.1. Dessin de grands graphes . . . 61
10.1.1. Plongement des nœuds . . . 61
10.1.2. Faisceautage des arêtes . . . 61
10.2. Représentation de masses de données tabulaire dans une grille . . . 62
10.3. Travaux non abordés . . . 62

Chapitre 11. Bibliographie de la visualisation de masses de données . . . 63

IV. Symbiose de la visualisation et de l'apprentissage automatique 67

Chapitre 12. Introduction . . . 71

Chapitre 13. Visualisation pour l'apprentissage . . . 73
13.1. Visualisation pour l'authentification biométrique . . . 73
13.1.1. Justification . . . 73
13.1.2. Le Zoo Graph pour contourner les limitations du Zoo Plot . . . 74
13.2. Visualisation pour l'analyse de base de documents étiquetés . . . 76
13.2.1. Justification . . . 76
13.2.2. ACID: une Application pour Classer des Images dans des bases de Données . . . 77

13.3. Visualisation pour la compréhension des réseaux profonds . . . 79
13.3.1. Justification . . . 79
13.3.2. Représentation des flux de données dans un réseau . . . 81

Chapitre 14. Apprentissage automatique pour la visualisation d'information . . . 85
14.1. Analyse automatique de techniques de visualisation . . . 85
14.1.1. Illustration avec la comparaison de dessin de graphes . . . 85
14.2. Dessin de graphes : plongement dans une vue nœuds-liens . . . 87
14.2.1. Introduction . . . 87
14.2.2. Définition de réseaux sur graphe pour le dessin de graphe . . . 87
14.2.3. Résultats . . . 89
14.3. Suppression de chevauchements pour le dessin de graphes . . . 90
14.3.1. Introduction . . . 90
14.3.2. Présentation de Fast Overlap Removal By stochastic gradIent Descent for Graph Drawing . . . 90

Chapitre 15. Discussion sur la symbiose de la visualisation d'information et l'apprentissage automatique . . . 93
15.1. Visualisation pour l'apprentissage automatique . . . 93
15.1.1. Application à l'authentification biométrique . . . 93
15.1.2. Application à l'apprentissage supervisé . . . 93
15.1.3. Application aux réseaux de neurones profonds . . . 94
15.2. Apprentissage automatique pour la visualisation d'information . . . 94
15.2.1. Analyse automatique de techniques de visualisation d'information . . . 94
15.2.2. Dessin de graphes . . . 95
15.2.3. Suppression de chevauchements . . . 96

Chapitre 16. Bibliographie de la symbiose de la visualisation et de l'apprentissage automatique . . . 97

V. Perspectives de recherche 101

Chapitre 17. Introduction aux perspectives de recherche . . . 105

Chapitre 18. Visualisation pour l'apprentissage automatique . . . 107
18.1. Rendre les systèmes de prédiction plus acceptables grâce à l'analyse visuelle et interactive . . . 107
18.1.1. L'évaluation des explications . . . 107
18.1.2. La confiance apportée aux explications . . . 108
18.1.3. L'inclusion de conseils dans les interfaces . . . 108
18.1.4. Inclusion de sémantique . . . 108
18.2. Impliquer la société . . . 109
18.2.1. L'alignement avec la société . . . 109
18.2.2. Lien avec l'informatique écoresponsable . . . 110
18.3. Rendre les explications plus versatiles . . . 110
18.3.1. Plus de diversité de modèles et de données d'entrée . . . 110
18.3.2. Plus de diversité dans les techniques d'explications . . . 110
18.3.3. Plus de diversité dans les médiums de visualisation . . . 111
18.4. Combler le fossé entre les communauté apprentissage, visualisation et explication . . . 111
18.4.1. Sensibiliser mutuellement les différentes communautés . . . 111
18.4.2. Collaboration entre les communautés visualisation, apprentissage et explication . . . 112
18.4.3. Cadres logiciels . . . 112
18.4.4. Alignement des domaines de recherche . . . 112

Chapitre 19. Apprentissage pour la visualisation . . . 113
19.1. Améliorer les techniques de plongement de données avec de l'apprentissage profond . . . 113
19.1.1. Application au dessin de graphes . . . 113

19.1.2. Application aux données multidimensionnelles 114
19.2. Remplacer des métriques d'évaluations coûteuses par des réseaux 114
19.3. Systématiser l'évaluation de techniques de visualisation avec de l'apprentissage profond 115
19.4. Faire le lien avec la communauté analyse de documents 115

Chapitre 20. Bibliographie des perspectives de recherche 117

VI. Conclusion 121

Chapitre 21. Conclusion générale 123

Table des figures

1.1. Les trois thématiques de recherche abordées dans ce manuscrit concernent (a) l'analyse de données et l'Apprentissage Automatique (AA), (b) la Visualisation d'Information (VI) dans le contexte des masses de données, et (c) la symbiose entre ces deux aspects. 5
1.2. Résumé des thématiques de recherche étudiées, des encadrements de doctorants, et des tâches administratives effectuées depuis l'obtention de ma thèse en 2012. 6
1.3. Dépendances entre les différentes parties du manuscrit. Les parties parties II à III peuvent se lire indépendamment, mais sont des prérequis pour la partie IV qui est elle-même le prérequis de la partie V. 7

2.1. Graphe ego-centré de mes co-auteurs sur les travaux liés à l'analyse de données et l'apprentissage automatique. 13
2.2. Famille de travaux en lien avec cette partie. Seuls ceux encadrés en rouge sont présentés dans le manuscrit. . 13
3.1. Fonctionnement général de l'enrôlement et de l'utilisation de systèmes biométriques pour (a) l'identification et (b) la vérification. 15

3.2. Impact du vieillissement de la donnée biométrique sur 3 individus [1]. Les individus A et B ont clairement besoin d'utiliser un système adaptatif, tandis que l'individu C ne souffre pas du problème de vieillissement. . 18
3.3. Taxonomie des systèmes de mise à jour de la référence biométrique [61]. 19
3.4. Contrairement aux autres approches de la littérature, Enhanced Template Update [60] tire parti des données collectées considérées comme étant des données d'imposture. 20
3.5. Description du réseau siamois utilisé dans notre approche [30]. Chaque branche B et B' extraient les caractéristiques des exemples à comparer, une couche calcule leur différence et le reste du réseau (A) calcule la distance. 21

4.1. Illustration des différentes familles d'explications. Les explications par attribution de caractéristiques (a), détection de caractéristiques (b), exemple (c), contre-exemple (d) sont généralement représentées à l'aide d'images statiques, tandis que l'extraction du comportement du modèle (e) nécessite en général une application d'Analyse Visuelle et Interactive (AVI). 24
4.2. Distribution des méthodes d'explications dans les papiers analysés par notre état de l'art [42]. L'attribution de caractéristiques est clairement surreprésentée. 24
4.3. Etapes nécessaires pour calculer l'explication avec FEM. Seules les activations issues de la dernière couche de convolution sont utilisées. Elles sont individuellement binarisées puis combinées afin de générer la carte finale. 26
4.5. Illustration de la différence de résultat de Feature Explanation Method (FEM) appliqué avec différente valeur de K pour le seuillage ainsi qu'à différentes profondeurs du réseau. 27
4.4. Illustration des zones mises en avant dans une portion de vidéo par les différentes méthodes d'explication appliquées sur les deux branches du réseau. L'action sélectionnée pour les méthodes la nécessitant est : Defensive backhand block. 27
4.6. Application de FEM à la fin de chaque bloc d'un réseau conventionnel de type ResNet50 faisant l'inférence d'une image quelconque. On voit clairement que la résolution est meilleure pour les premières couches, mais également que la carte est moins informative. 28
4.7. Étapes nécessaires pour calculer l'explication avec Multi-Layer Feature Explanation Method (MLFEM). FEM est appliquée en sortie de plusieurs couches de convolution de résolution variable. Les explications sont ensuite fusionnées. 28
4.8. Illustration des stratégies de fusion testées pour MLFEM. 29
4.9. Illustration de cartes de densité de fixation de regard pour quelques images de la base Mexculture. 29

5.1. Illustration de la haute résolution obtenue avec la méthode H2O [62] que nous sommes en train d'améliorer. 32

7.1. Graphe égo centré de mes co-auteurs sur les travaux liés à la Visualisation d'Information (VI). 43
7.2. Illustrations diverses de l'incapacité des résumés statistiques à clairement représenter les jeux de données contrairement à la Visualisation d'Information (VI) (à partir du moment où nous sommes capables de générer une représentation qui ne ment pas). 44
7.3. Illustration d'une chaîne de traitements standard pour aller de la donnée brute, vers sa représentation [33]. . 44
7.4. Famille de travaux réalisés dans cette thématique. Seuls ceux encadrés en rouge sont présentés. 44

8.1. Illustration de l'effet pelote de poils sur un graphe de taille raisonnable. L'effet est naturellement pire sur un plus gros graphe. On s'attend à l'éviter au maximum avec les approches multiniveaux. 47
8.2. Illustration du concept de partitionnement hiérarchique de graphe et de graphe quotient. Le graphe de 8 nœuds a été partitionné en 3 niveaux. Chaque niveau est encodé par un graphe quotient. 48
8.3. Dessins du graphe représentant le dépôt git de cette HDR (les nœuds correspondent aux commits et fichiers) à l'aide de trois méthodes de dessin de graphe. On voit clairement que la lisibilité diffère en fonction des algorithmes, et qu'il y a toujours une marge de progression possible. 49
8.4. Dessins du graphe du dépôt git de cette HDR avec et sans utilisation de faisceautage d'arêtes. Le plongement des nœuds est identique dans les deux représentations. Dans la seconde, moins de pixels sont utilisés pour représenter les arêtes. 50
8.5. Représentation des données manipulées dans l'approche de dessin hiérarchique basée sur le calcul. 50
8.6. Illustration du partitionnement d'un graphe quelconque grâce à l'information de dégénérescence. Le partitionnement est appliqué récursivement jusqu'à obtenir un arbre.. 51
8.7. Comparaison de HCBL (2e ligne) à FM3 (1re ligne) sur certains graphes étudiés. La 3e ligne représente la partie centrale du graphe. On voit clairement que notre approche diminue l'aspect pelote de poils. 52
8.8. Illustration du souci de traversée de partitions non corrélées par les approches standards de faisceautage d'arêtes. 53
8.9. Hierarchie et relation du power graphe partitionné en 8.10a 53
8.10. Description du pipeline de partitionnement hiéarchique évitant les partitions non liées aux nœuds source et destination. Le graphe de partitionnement est en figure 8.9 page 53. 53
8.11. Evaluation de la méthode sur un graphe représentant le résultat d'un système d'authentification biométrique. 54

9.1. Illustration de quelques problèmes dans les projections 2D de données hautes Dimension. (a) souffre de problèmes de recouvrement corrigés dans (b) : il est difficile d'estimer correctement la densité locale des points. (b) n'a plus de recouvrement (au prix d'un dessin plus grand ou d'une taille d'éléments plus petite). Cependant, la majorité de l'espace ne contient pas d'information. 56
9.2. Résumé de l'algorithme VRGRid [31] décrit en section 9.2. 57
9.3. Illustration des enveloppes convexes $\mathcal{B}_0$ (qui est inexistante par construction en réalité) à $\mathcal{B}5$ associées à leurs cellules de bordures C_0 à C_5 pour un jeu de données à projeter sur une grille 9×9. 58
9.4. Illustration de l'initialisation d'un jeu de données de 78 points. Il a été normalisé pour être contenu dans la bordure extérieure $\mathcal{B}5$. 3 points artificiels ont été ajoutés sur la prochaine bordure. 58
9.5. Résultat de l'application de Lloyd sur la 1re itération de VRGRid. Les segments représentent les points de P_0 les plus proches des cellules de C_4. 59
9.6. Résultat de l'affectation sur une bordure, et illustration d'un point (le rouge) qui n'a pas été affecté sur la bordure courante et qui est positionné à l'extérieur. 59
9.7. Comparaison de la projection VRGRid sur quelques jeux de données avec SSM et DGRid. 60

10.1. Illustration d'un jeu de données sur lequel il n'est pas possible d'utiliser le faisceautage hiérarchique présenté dans cette partie. Ici les partitions (représentées par des couleurs différentes) sont éclatées dans le plongement (une même partition peut être à plusieurs endroits, une partition peut être visuellement incluse dans une autre). 62

12.1. Graphes égo centré de mes co-auteurs sur les travaux liés à la symbiose entre entre la Visualisation d'Information (VI) et l'Apprentissage Automatique (AA). 71
12.2. Famille de travaux réalisés dans cette thématique. Seuls ceux encadrés en rouge sont présentés. 72

13.1. Exemple de figures utilisées de façon standard d'évaluation d'un même système biométrique. Elles apportent des informations sur la performance, sans en expliquer la raison. 73

13.2. Description du pipeline de construction du Zoo Graph depuis la base de scores de vérification biométrique, jusqu'à la visualisation interactive finale. 74
13.3. Illustration de la différence entre le Zoo Plot et le Zoo Graph pour trois bases de données biométriques. . . . 76
13.4. La visualisation est encombrée en raison du grand nombre d'arêtes. Dans ce jeu de données, la mesure est une distance pas un score, donc les pires individus sont sous le EER. 76
13.5. Fonctionnement global d'ACID. Un vecteur de caractéristiques et une étiquette sont donnés pour chaque image qui représente un document. Notre logiciel construit un graphe où les documents sont rassemblés dans des classes en fonction de leur étiquette. Il permet à l'utilisateur de corriger les erreurs de classification. 1- Seules les caractéristiques sont utilisées pour créer le graphe de voisinage reliant les images en fonction de leurs similarités. 2- Le graphe quotient représentant les classes est créé en utilisant le graphe de voisinage et les étiquettes. 3- Une mesure globale est calculée pour indiquer la cohérence entre le graphe de voisinage et le graphe de quotient. 4- Le graphe quotient est affiché. Des corrections de classification sont proposées à l'utilisateur. 5- L'utilisateur peut interagir avec le logiciel pour explorer la base de données d'images. 6- L'utilisateur confirme un changement ; la base de données de documents étiquetés est mise à jour et le nouveau graphe quotient est calculé. 77
13.6. À gauche : le graphe de voisinage modélisant la similarité entre les images de documents, les couleurs indiquent la classe de chaque document. À droite : le graphe quotient correspondant où chaque nœud représente une classe d'images de documents. 78
13.7. Résultat visuel obtenu après la construction du graphe quotient. Le panneau central contient le graphe quotient. Les nœuds correspondent aux images et les arêtes relient les images similaires, les métanœuds contiennent tous les nœuds avec des étiquettes identiques et les méta-arêtes relient les métanœuds contenant des images liées. La couleur du métanœud indique sa cohésion interne. Par exemple, le métanœud rose clair « PARKING FEES » a une bonne cohésion (probablement aucune erreur de classification) alors que la couleur rouge du métanœud « ANOMALIES » montre qu'il pourrait contenir plusieurs erreurs d'étiquetage. Sur le panneau de droite, des suggestions de changements d'étiquettes sont proposées à l'utilisateur. Par exemple, ACID propose de déplacer une image du métanœud « ACTIVITIE » vers le métanœud « PARKING FEES ». ACID indique également que ce changement améliorera la cohésion du graphe global par un facteur de $0,41$. 79
13.8. Les 38 dimensions utilisées pour caractériser chaque article sont regroupées au sein de 9 groupes. Elles nous ont permis d'extraire correctement les informations d'intérêt pour chaque article et de les organiser. 80
13.9. Illustration de l'inférence et de la descente de gradient sur un lot pour un modèle d'Apprentissage Profond (AP). Le processus génère plusieurs données : prédictions, cartes d'activations, perte, métriques et gradients. Chacune d'entre elles peut être utilisée pour calculer des explications. 80
13.10. Catégorisation des méthodes d'explication. Nous pouvons faire la distinction entre les méthodes qui utilisent des moyens externes (post-hoc) et les méthodes qui utilisent les internes du modèle (intrinsèques), et entre les méthodes qui fournissent des explications uniquement pour l'entrée actuelle (locales) et celles pour le comportement général du modèle (globales). 80
13.11. Workflow de la méthode de visualisation d'activations par flux. 82
13.12. Illustration de l'approche sur le réseau LeNet5 et la base de données MNIST. Le modèle est performant et convenablement dimensionné. 83
13.13. Illustration de l'approche sur le réseau VGG16 et la base de données MNIST. Le modèle est performant et surdimensionné. 84

14.1. Framework de comparaison de représentation visuelle. 86
14.2. Représentation du même graphe en utilisant deux techniques de représentation différentes : la matrice d'adjacence en 14.2a, et la vue nœuds-liens en 14.2b. 87
14.3. Schéma de l'architecture de réseau choisie pour calculer le plongement d'un graphe. 88
14.4. Représentation de quelques graphes issus de la base Rome $N \approx \{10, 50, 75, 100\}$. Fscratch, Fscratch PMDS sont deux variantes de notre approche. Les autres sont des approches concurrentes. 89
14.5. Illustration du concept de chevauchements sur le graphe du dépôt git de ce manuscrit. 90
14.6. Fonctionnement général du l'algorithme FORBID qui cherche à minimiser la quantité déplacements des nœuds et le changement d'échelle du dessin. La variante FORBID utilise le dernier « Agencement optimisé » à chaque nouvelle étape de « Mise à l'échelle du dessin », tandis que FORBID' utilise le plongement initial. 91

14.7. Visualisation de graphes issus du jeu de données Graphviz avec une suppression des chevauchements calculées par FORBID, FORBID', PFS' GTREE et PRISM. Les nœuds sont colorés en rouge transparent si ils souffrent de chevauchements et bleu autrement. 92
15.1. Illustration de l'application d'AVI utilisant un PowerGraph pour évaluer un système biométrique. 93

15.2. Représentation des trois méthodes de comparaison de paires de matrices de confusion [54]. 94
15.3. Comparaison des encodages utilisés dans nos essais d'explication d'un couple données/réseau : les boites de confinement de la vue par flux, la représentation orientée pixels et les courbes fractales. Ici différentes couches de LeNet5 en utilisant la base MNIST sont utilisées pour illustrer les concepts [31]. 94
15.4. Autre type de représentation permettant une visualisation sur des réseaux plus complexes. 95
15.5. Netprune : application de simplification de Réseau de Neurones Profond (RNP), reposant sur différentes représentations interactives dédiées. 95
15.6. Illustration des différentes familles de distracteurs étudiés dans l'étude de perception visuelle [20]. 96
15.7. Cette image illustre l'intérêt d'utiliser une méthode de suppression du chevauchement qui n'a pas d'aprioris sur la forme des nœuds à déplacer. (a) La Joconde découpée en super-pixels non convexes avec SLIC [1]. (b) Version altérée de (a) où la taille de chaque polygone est doublée. (c) Résultat de FORBID (qui considère des glyphes rectangulaires). (d) Résultat de SORDID (qui prend en compte la forme des polygones). 96

17.1. Concepts présentés dans cette partie. 105

18.1. L'acceptation des systèmes d'AP peut se faire grâce à l'augmentation de la confiance apportée aux explications. 107
18.2. Les outils d'AVI pour l'explicabilité devraient aider la société à mieux appréhender les systèmes reposants sur l'Intelligence Artificielle (IA). 108
18.3. Illustration de différentes méthodes de représentation de cartes de saillances recensées dans la littérature. La première colonne représente l'image classée et la seconde colonne une carte de saillance. La troisième colonne correspond à la technique de représentation majoritairement utilisée. Les autres colonnes sont peu utilisées, mais probablement plus efficaces. 109
18.4. Représentation « multiple petits » d'explications de type cartes de saillance calculées avec LIME pour les 10 premières classes prédites sur l'image de test. 110
18.5. Différentes stratégies de fusion de cartes d'explication générées par LIME. La première ligne représente l'explication de la classe majoritaire. La seconde ligne représente une somme pondérée de toutes les cartes par la probabilité de la classe associée. La troisième ligne représente la moyenne de toutes les cartes. 111
18.7. Distribution des aires d'expertise parmi les auteurs d'un ensemble de papiers liant explicabilité pour les réseaux de neurones profonds et l'AVI. 112
18.6. Les outils d'AVI pour l'explicabilité devraient aider la société à mieux appréhender les systèmes reposant sur l'IA. 112
19.1. Changement de paradigme pour le plongement de données. 113

19.2. Amélioration de la méthode de dessin de graphes en détectant automatiquement des partitions, utilisant des réseaux adaptés à chaque partition. L'approche peut naturellement être hiérarchique en appliquant la procédure sur chaque partition à dessiner jusqu'à un certain critère d'arrêt. 114
19.3. Changement de paradigme pour le calcul de métrique d'évaluation. 114
19.4. Changement de paradigme d'évaluation de visualisation. 115

Acronymes

AA Apprentissage Automatique 5, 7, 8, 13, 16, 23, 31, 71, 72, 79, 85, 93, 105, 107, 123

ACP Analyse en Composante Principale 49

AP Apprentissage Profond xiii, xiv, 8, 20, 31, 33, 71–73, 79, 80, 105, 107, 109–115

APX Apprentissage Profond Explicable 79, 110, 111

AVI Analyse Visuelle et Interactive xi, xiv, 5, 7, 24, 32, 43, 72, 80, 93, 107–112

DDF Dynamique de Frappe au clavier 7, 14, 16–22, 31

FEM Feature Explanation Method xi, 25–29, 80

GradCam Grad-CAM 27, 29

GuidedBackProp Guided Back-propagation 27

GuidedGradCam Guided Grad-CAM 27

HCBL Hierarchical Corness Based Layout 50–52

IA Intelligence Artificielle xiv, 1, 8, 14, 79, 107–113

IAX IA Explicable 14, 79, 111, 112

MLFEM Multi-Layer Feature Explanation Method xi, 25, 27, 28, 80

RN Réseau de Neurones 26, 28, 31, 72, 113, 123

RNP Réseau de Neurones Profond xiv, 20, 21, 26, 80, 81, 85, 94, 95, 113, 115

TEE Taux d'Erreur Égale 17, 73

TFA Taux de Fausse Acception 17, 20, 73

TFR Taux de Faux Rejet 17, 19, 20, 73

VanillaGrad Vanilla Gradient-Based Propagation 27

VI Visualisation d'Information 5–8, 43, 44, 61, 71, 72, 85, 93, 94, 105, 107, 112, 113, 123

Première partie

Introduction

Chapitre 1.

Introduction générale

Contexte

Ce manuscrit retrace une partie de mes contributions depuis mon recrutement comme maître de conférences « au département informatique de l'Institut Universitaire de Technologie de Bordeaux » et au « Laboratoire Bordelais de Recherche en Inofrmatique » en septembre 2013. Le diagramme d'Euler de la figure 1.1 illustre le lien entre mes travaux qui concernent à la fois : (a) mon domaine de recherche initial (*c.-à-d.* le travail effectué en thèse et postdoctorat) en *analyse de données* et *Apprentissage Automatique (AA)*, (b) la compétence que j'ai acquise lors mon arrivée au LaBRI en *Visualisation d'Information (VI)* pour les masses de données, et (c) celle que je suis en train de développer sur la *symbiose entre la VI et l'AA*.

Ces trois thématiques sont toujours d'actualité et d'intérêt au moment de la rédaction de ce manuscrit. En effet, vu la multiplication de collecte de masses de données par les acteurs privés (*p. ex.* sites internet, applications mobiles, suivi de la consommation d'énergie) ou publics (*p. ex.* surveillance de masse dans certaines agglomérations, données ouvertes telles que https://www.data.gouv.fr/fr/), la mise en place de techniques et méthodes *d'analyse de données* efficaces est nécessaire. De plus, il faut restituer les informations, qu'elles soient (a) contenues dans ces données brutes, (b) obtenues par une extraction algorithmique, ou (c) extraites à l'aide de techniques d'AA. Cette restitution se fait souvent à l'aide de techniques de *VI* pouvant être intégrées dans des outils d'Analyse Visuelle et Interactive (AVI). Les modèles d'AA utilisés pour extraire l'information ont également besoin d'être compris afin de leur faire confiance : c'est ce qui amène l'intérêt de la *VI pour l'AA*. Les représentations visuelles ne doivent pas mentir pour représenter convenablement les données : c'est ce qui amène un des intérêts de l'*AA pour la VI* afin de vérifier qu'une machine puisse extraire les bonnes informations, en assumant qu'un humain en ferait autant. Un autre intérêt est de créer de nouvelles familles de méthodes de VI qui limitent le nombre d'aprioris ou d'heuristiques à fournir par le développeur de telles méthodes.

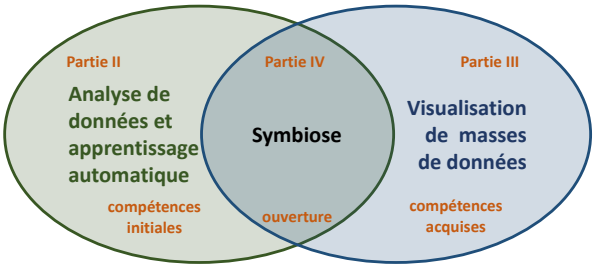


Figure 1.1. – Les trois thématiques de recherche abordées dans ce manuscrit concernent (a) l'analyse de données et l'Apprentissage Automatique (AA), (b) la Visualisation d'Information (VI) dans le contexte des masses de données, et (c) la symbiose entre ces deux aspects.

La figure 1.2 page suivante liste les thématiques de recherche auxquelles j'ai contribué, leurs valorisations principales, leurs encadrements doctoraux, ainsi que les tâches administratives auxquelles j'ai participé. La plupart des travaux ont été réalisés avec des doctorants que j'ai dirigés officiellement (Rémi Delassus [1] avec Guy Melançon, Adrien Halnaut [2] avec David Auber, et Korlan Rysbayeva [3] avec Nicholas Journet) ou avec lesquels j'ai collaboré (Paulo Henrique Pisani [4] avec André CPLF de Carvalho de l'université de São Paulo/Brésil, Loann Giovannangeli avec Romain Bourqui, et Luc-Etienne Pommé-Cassierou avec David Auber), ainsi qu'avec des étudiants de master (Kasi Ahmed Asif Fuad, Luca Bourroux, Alexei Zhukov, encadrés avec Jenny Benois-Pineau et Romain Bourqui). Certains travaux sont également issus de collaborations internationales avec et sans participation de doctorants avec André CPLF de Carvalho de l'université de São Paulo/Brésil, Anderson Rocha de l'université de Campinas/Brésil, et Marco Angelini de l'université de

1. R. Delassus. « Apprentissage automatique pour la détection d'anomalies dans les données ouvertes: application à la cartographie ». Thèse de doct. Université de Bordeaux, 2018
2. A. Halnaut. « Méthodes et outils d'analyse visuelle pour la compréhension, l'optimisation et l'élaboration de modèles de réseaux de neurones profonds ». Thèse de doct. Université de Bordeaux, 2024. En prép.
3. K. Rysbayeva. « Apprentissage par métriques profondes et classification de données hiérarchiques et multimodales: application aux rapports dépollution des sols ». Thèse de doct. Université de Bordeaux, 2023
4. P. H. Pisani. « Biometrics in a data stream context ». Thèse de doct. Université de Sao Paulo, 2017

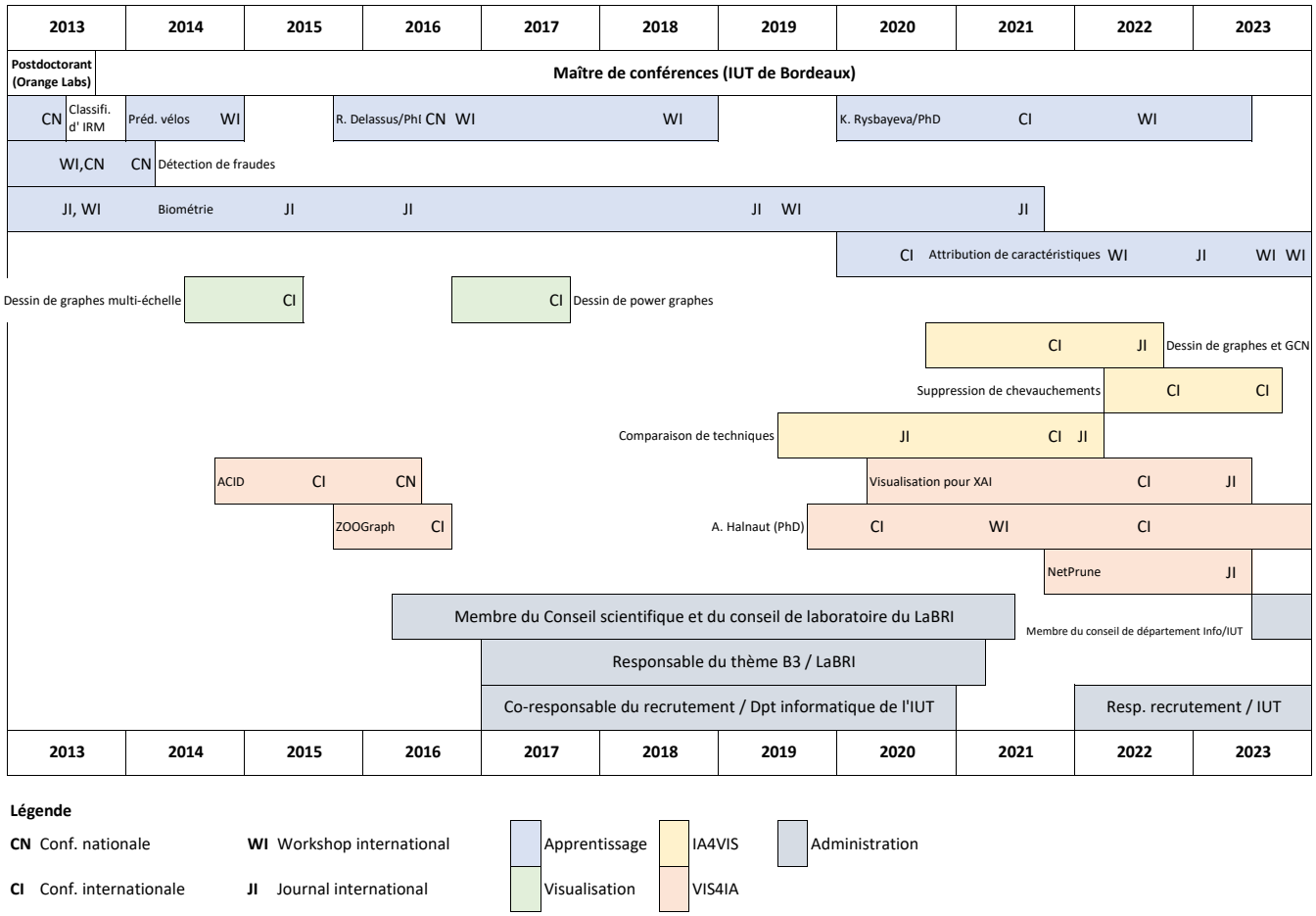


Figure 1.2. – Résumé des thématiques de recherche étudiées, des encadrements de doctorants, et des tâches administratives effectuées depuis l'obtention de ma thèse en 2012.

Rome/Italie.

Les matières que j'ai enseignées à l'IUT sont majoritairement techniques (*p. ex.* système d'exploitation, programmations diverses, algorithmique), et plutôt éloignées de mes travaux de recherche, même si j'ai pu participer à des enseignements liés à l'apprentissage et l'analyse de données (introduction a l'apprentissage, optimisation). Cela s'explique facilement, car les étudiants de l'IUT obtiennent un BAC+3 et s'insèrent rapidement dans le monde professionnel (c'est d'autant plus vrai avec la suppression DUT au profit du BUT). En contrepartie, j'enseigne tous les ans auprès d'un public BAC+5 (*c.-à-d.* en master bio-informatique et informatique, ou en école d'ingénieur d'informatique) des matières liées à mes travaux de recherche (*p. ex.* apprentissage, explicabilité, ou analyse de données et visualisation).

Ma responsabilité administrative majeure à l'IUT de Bordeaux sur cette décennie correspond à la gestion du recrutement des étudiants de 1re année du département informatique (opérateur de APB puis Parcoursup, organisation de l'évaluation des dossiers et de leur distribution aux collègues, des invitations des candidats à venir faire une épreuve, de la génération du classement, des échanges divers avec les candidats ...). Concernant l'administration de la recherche, j'ai été responsable du thème « Back to Bench and Beyond » (*c.-à-d.* le thème lié à la gestion de masses de données et la VI) avant la réorganisation des équipes du LaBRI en 2021, et j'ai été membre à différents conseils institutionnels (du département à l'IUT et scientifique et de laboratoire au LaBRI). Mes travaux de recherche ont été financés par différents projets sur lesquels j'étais le porteur (INS2I JCJC, GIS Albatros, Consulat Français au Brésil, projets locaux du laboratoire ou de l'université) ou participant (ANR, projet européen, projet d'investissement d'avenir). Je suis ainsi récipiendaire de la Prime d'Encadrement Doctoral et de Recherche depuis 2017.

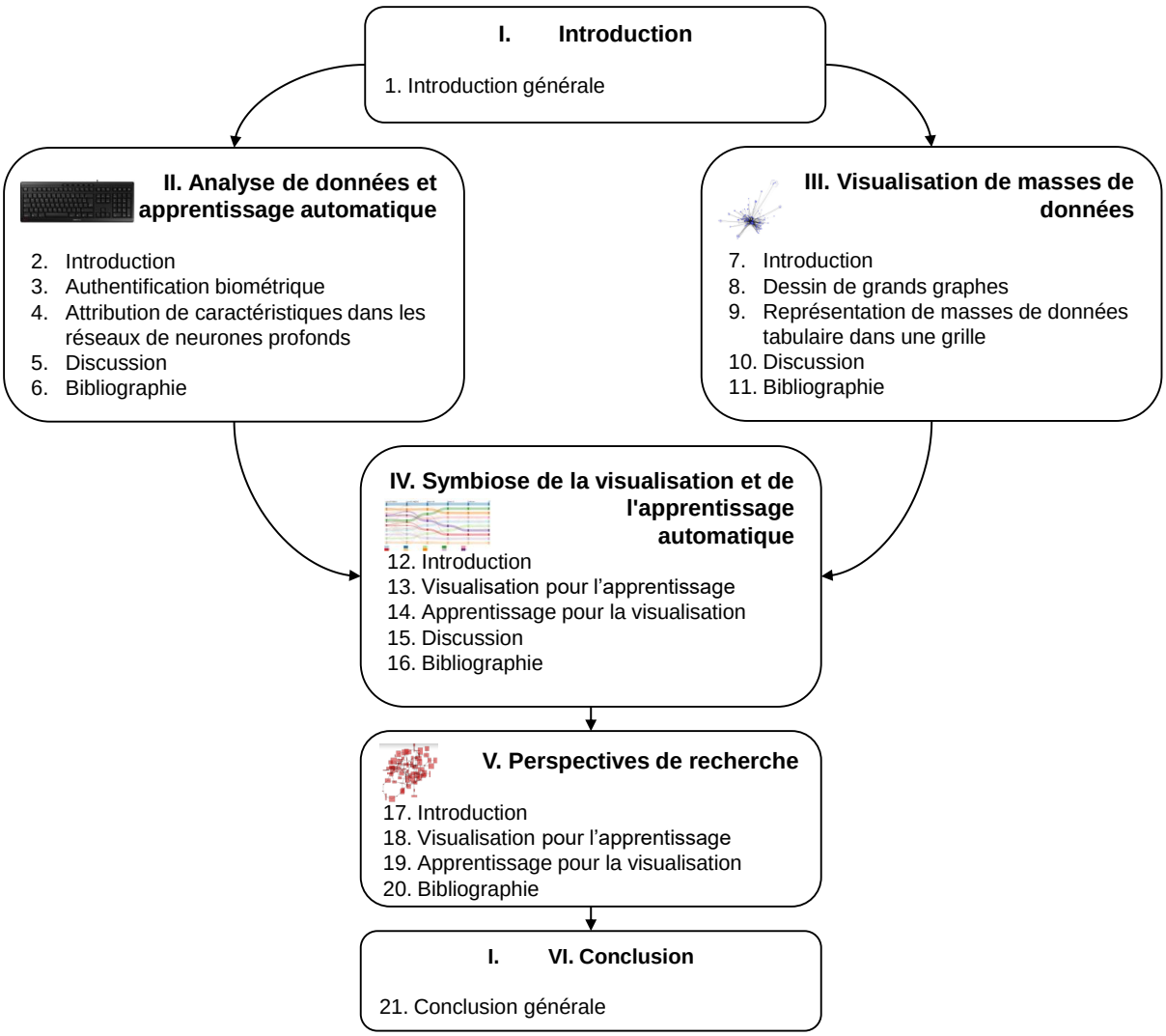


Figure 1.3. – Dépendances entre les différentes parties du manuscrit. Les parties parties II à III peuvent se lire indépendamment, mais sont des prérequis pour la partie IV qui est elle-même le prérequis de la partie V.

Organisation du manuscrit

La structure du manuscrit et le lien entre les différentes parties sont résumés dans la figure 1.3 ; la partie courante sert à introduire son contexte et son organisation.

La partie II synthétise la consolidation de mes compétences en analyse de données et AA. Le chapitre 2 introduit les différents aspects de cette thématique de recherche. Le chapitre 3 se présente des travaux en authentification biométrique. Les contributions portent sur l'authentification par Dynamique de Frappe au clavier (DDF), principalement sur les aspects de mise à jour de la référence biométrique. Le chapitre 4 s'intéresse à l'explicabilité des réseaux de neurones profonds (sans utiliser de techniques de AVI). Le chapitre 5 synthétise ces travaux, ainsi que ceux non abordés.

La partie III synthétise mon acquisition de compétences en VI. En effet, en arrivant au LaBRI, j'ai intégré une équipe spécialisée dans la visualisation de masses de données (nommée MaBIOVIS à ce moment) et j'ai dû changer de thématique de recherche pour m'y intégrer. Le chapitre 7 introduit les différents aspects de cette thématique de recherche. Le chapitre 8 montre mes travaux sur le dessin de grands graphes, avec une une méthode qui repose sur une approche hiérarchique pour décomposer le problème de placement de nœuds, suivi d'une approche de dessin de faisceau d'arêtes dans le cas particulier de graphes avec un partitionnement hiérarchique. Le chapitre 9 présente les travaux sur des méthodes de projection de données dans une grille. Le chapitre 10 synthétise ces travaux, ainsi que ceux qui n'ont pas été abordés.

La partie IV présente mon ouverture sur un nouveau pan

de recherche lié à la symbiose entre la VI et l'Intelligence Artificielle (IA). Je suis notamment moteur sur des aspects VI pour l'AA, mais je participe également aux aspects AA pour la VI. Le chapitre 12 introduit les différents aspects de cette thématique de recherche. Le chapitre 13 se focalise sur les aspects VI pour l'AA avec différentes contributions spécifiques à l'authentification biométrique et d'autres, plus générales pour l'Apprentissage Profond (AP). Le chapitre 14 aborde les aspects AA pour la VI sur des aspects liés à l'évaluation automatique de représentations visuelles et d'autres, liées au dessin de graphes. Le chapitre 15 synthétise ces travaux.

La partie V présente les perspectives de recherche que j'ai identifiées sur les aspects symbiose entre la VI et l'IA. Le premier chapitre décrit les perspectives en VI pour l'AA, tandis que le second décrit celles sur l'AA pour la VI.

La partie VI clôture brièvement ce manuscrit.

Afin de pouvoir les parcourir plus facilement, les quatre parties principales (*c.-à-d.* les parties II à V) disposent chacune de leur bibliographie : une unique référence bibliographique peut avoir une numérotation différente en fonction de la partie qui la référence. Afin de mettre en évidence mes publications et encadrements, le nom des étudiants encadrés apparaît souligné, et le mien en gras.

Deuxième partie

Analyse de données et apprentissage automatique

Chapitre 2. Introduction à l'analyse de données et l'apprentissage automatique 13
Chapitre 3. Authentification biométrique 15
3.1. Introduction à l'authentification biométrique 15
3.1.1. Généralités sur la biométrie 15
3.1.2. Modalités biométriques 16
3.1.3. Évaluation de systèmes d'authentification biométrique 17
3.1.4. Stratégies d'amélioration des performances des systèmes d'authentification biométriques 17
3.1.5. La dynamique de frappe au clavier : une biométrie comportementale 18
3.2. Mise à jour de la référence biométrique : une nécessité pour les biométries comportementales . . 19
3.2.1. Contexte 19
3.2.2. Méthode améliorée de mise à jour 19
3.3. Diminution de la quantité de captures pour l'enrôlement en dynamique de frappe au clavier . . 20
3.3.1. Contexte 20
3.3.2. Réseau siamois pour la dynamique de frappe au clavier 21
Chapitre 4. Attribution de caractéristiques pour les réseaux de neurones profonds . . . 23
4.1. Introduction à l'explicabilité pour l'explication des réseaux de neurones profonds 23
4.1.1. Attribution de caractéristiques 23
4.1.2. Détection de caractéristiques apprises 24
4.1.3. Explication par exemple factuel 25
4.1.4. Explication par contre-exemple 25
4.1.5. Extraction du comportement du modèle 25
4.2. Attribution de caractéristiques pour réseau de neurones sans prérequis de classe 25
4.2.1. Explication de caractéristiques avec FEM 26
4.2.2. Amélioration de la résolution de l'explication avec ML-FEM 27
Chapitre 5. Discussion sur l'analyse de données et l'intelligence artificielle 31
5.1. Authentification biométrique 31
5.1.1. Mise à jour de la référence biométrique 31
5.1.2. Adaptation aux réseaux de neurones profonds 31
5.1.3. De l'acceptabilité de l'authentification biométrique 31
5.2. Attribution de caractéristiques pour les réseaux de neurones profonds 32
5.2.1. De la faible résolution des explications 32
5.2.2. Évaluation des méthodes d'explication 32
5.3. Travaux non abordés 32
5.3.1. Détection de fraudes dans les systèmes de paiement mobile 32
5.3.2. Détection de panne dans les systèmes de vélos en libre-service 33
5.3.3. Segmentation d'images satellites 33
5.3.4. Classification hiérarchique et multimodale 33
Chapitre 6. Bibliographie de l'analyse de données et l'apprentissage automatique 35

Chapitre 2.

Introduction à l'analyse de données et l'apprentissage automatique

Contexte

Cette partie présente succinctement certaines contributions effectuées depuis mon arrivée au LaBRI dans les thématiques liées à l'analyse de données et l'Apprentissage Automatique (AA). Elles restent représentatives du travail que j'ai réalisé, et les compétences liées à leurs thématiques correspondent à celles que j'ai initialement acquises en thèse et que j'ai consolidées tout au long de ma carrière.

Elles concernent l'authentification biométrique (*c.-à-d.* la suite directe de mes travaux de thèse), ainsi que l'explication de décision de classification dans des réseaux de convolution en mode boite blanche (*c.-à-d.* un nouveau thème que je souhaite approfondir, et qui est discuté plus en détail dans le chapitre 13). Les travaux concernant la détection de fraude dans les systèmes de paiement mobile [67], la prédiction dans les systèmes de vélos en libre-service [22] , la segmentation d'image satellite [21], ou la classification hiérarchique de documents [71] ne sont pas abordés.

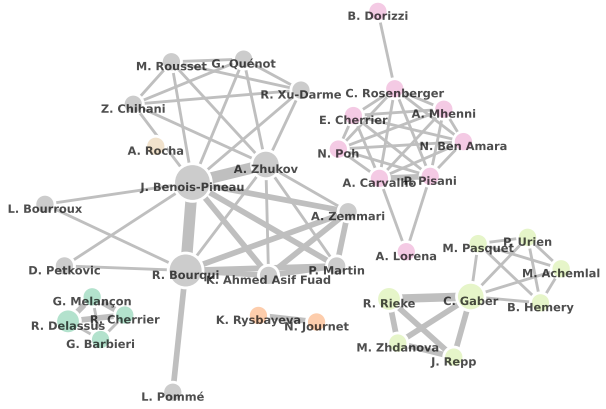


Figure 2.1. – Graphe ego-centré de mes co-auteurs sur les travaux liés à l'analyse de données et l'apprentissage automatique.

Le graphe ego-centré de mes co-auteurs, depuis l'obtention de mon doctorat et sur cette thématique, est présenté en figure 2.1. La taille des nœuds et des arêtes est proportionnelle au nombre de publications communes avec les collègues concernés sur cette thématique. La couleur dépend d'un algorithme de partitionnement de graphe, mais reflète globalement bien les origines des collaborations.

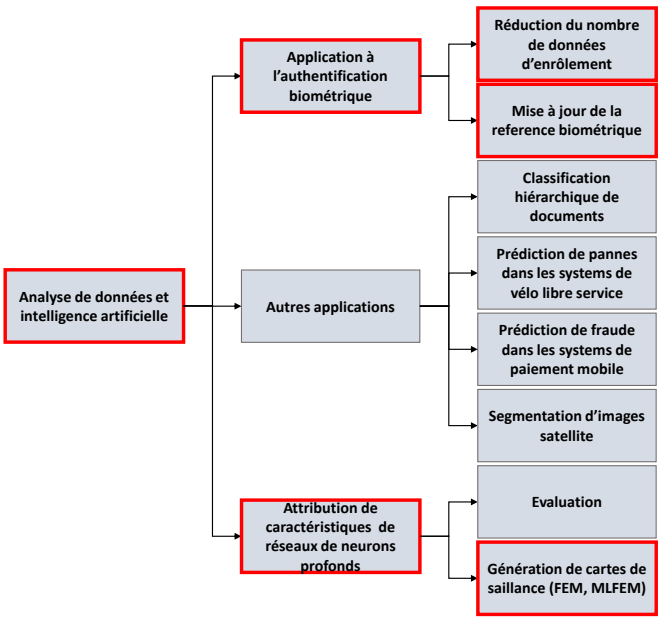


Figure 2.2. – Famille de travaux en lien avec cette partie. Seuls ceux encadrés en rouge sont présentés dans le manuscrit.

Organisation

Les différents concepts détaillés dans cette partie sont illustrés en figure 2.2 page précédente.

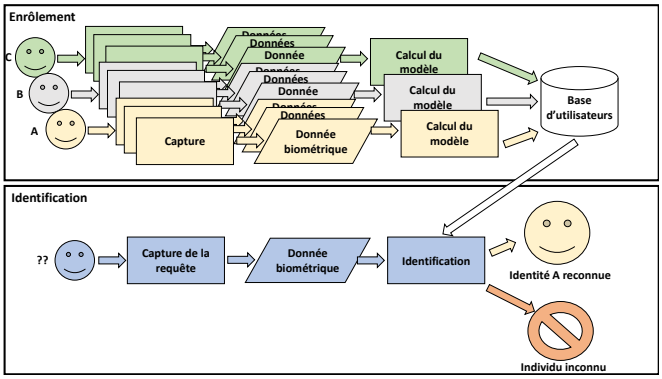
Le chapitre 3 présente des travaux liés à l'authentification biométrique. La première section décrit le fonctionnement général d'un système d'authentification biométrique et s'attarde plus en détail sur la Dynamique de Frappe au clavier (DDF) et la mise à jour de la référence biométrique. La seconde section présente un travail effectué sur la mise à jour de la référence biométrique. Ces travaux ont été réalisés en collaboration avec l'université de São Paulo/Brésil. La troisième section présente une étude exploratoire sur l'utilisation de réseaux siamois pour diminuer la quantité de données d'enrôlement dans le cas de l'authentification par DDF. Ces travaux ont été réalisés en collaboration avec l'université de Campinas/Brésil.

Le chapitre 4 présente des travaux liés à l'IA Explicable (IAX) et la proposition de méthodes d'explication de type attribution de caractéristiques. Ces travaux ont été réalisés entre autres avec des collègues des départements *Image & Son* et *Combinatoire et Algorithmique* du LaBRI. Dans le cas de l'explication de décision de réseaux de classification d'images, ces techniques proposées génèrent une carte de saillance, représentée à l'aide d'une carte de chaleur, qui met en évidence les pixels qui ont fortement contribué à la décision finale.

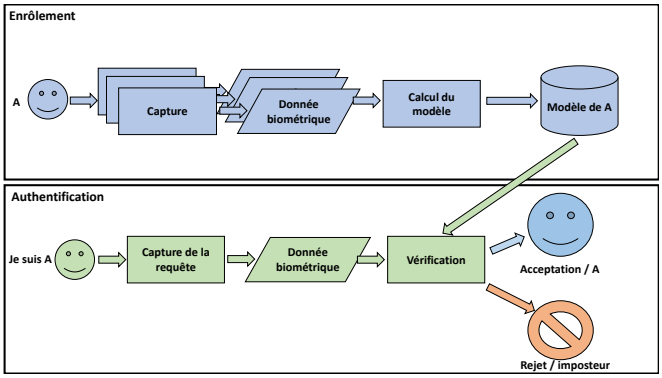
Le chapitre 5 présente une synthèse des travaux liés à cette thématique, qu'ils aient été présentés ou non.

Chapitre 3.

Authentification biométrique



(a) Fonctionnement général de l'identification.



(b) Fonctionnement général de l'authentification.

Figure 3.1. – Fonctionnement général de l'enrôlement et de l'utilisation de systèmes biométriques pour (a) l'identification et (b) la vérification.

3.1. Introduction à l'authentification biométrique

Mes travaux de thèse ont principalement concerné l'authentification biométrique, et j'ai également continué à travailler sur cette thématique après mon arrivée au LaBRI. Ce chapitre présente certains de mes travaux, principalement effectués en collaboration avec deux équipes brésiliennes. Cette première section décrit brièvement ce qu'est un système d'authentification biométrique, tandis que les deux suivantes présentent deux travaux : sur la mise à jour de la référence biométrique, et sur l'utilisation d'un réseau siamois pour extraire des caractéristiques plus pertinentes et améliorer les performances de reconnaissance.

3.1.1. Généralités sur la biométrie

Selon la CNIL [1], la biométrie :

> regroupe l'ensemble des techniques informatiques permettant de reconnaître automatiquement un individu à partir de ses caractéristiques physiques, biologiques, voire comportementales. Les données biométriques sont des données à caractère personnel, car elles permettent d'identifier une personne. Elles ont, pour la plupart, la particularité d'être uniques et permanentes (ADN, empreintes digitales, etc.).

Plus précisément, nous pouvons identifier deux familles de systèmes biométriques :

- les systèmes d'*identification* (voir la figure 3.1a) qui cherchent à identifier l'identité d'un individu à partir de la collecte de ses données biométriques ;
- les systèmes d'*authentification* (voir la figure 3.1b) qui cherchent à vérifier l'identité présumée d'un individu par rapport à son modèle à partir de ses données biométriques. C'est ce cadre d'utilisation qui nous intéresse.

Ainsi, l'authentification biométrique permet de reconnaître un individu par rapport à ce qu'*il est*. Il s'agit d'une alternative non mutuellement exclusive à d'autres types d'authentifications basées sur ce qu'*il sait* (*p. ex.* un mot de passe), ou ce qu'*il possède* (*p. ex.* un smartphone associé à un numéro de téléphone).

Un système d'authentification biométrique est constitué deux modules principaux : les modules d'enrôlement et d'authentification.

1. https://www.cnil.fr/fr/biometrie

Table 3.1. – Propriétés des modalités biométriques des systèmes majeurs.

Modalité	Universalité	Unicité	Permanence	Collectabilité	Acceptabilité	Performance
Modalités comportementales						
Démarche	Oui	Non	Faible	Oui	Oui	★★★
Signature dynamique	Oui	Oui	Faible	Oui	Oui	★★★★
Dynamique de frappe	Oui	Oui	Faible	Oui	Oui	★★★
Voix	Oui	Oui	Faible	Oui	Oui	★★★★
Modalités biologiques						
ADN	Oui	Oui	Oui	Faible	Faible	★★★★★
Groupe sanguin	Oui	Non	Oui	Faible	Non	★
Signal du cerveau (EEG)	Oui	Oui	Oui	Faible	Non	★★★★
Modalités morphologiques						
Iris	Oui	Oui	Oui	Oui	Un peu	★★★★★
Rétine	Oui	Oui	Oui	Oui	Un peu	★★★★★
Visage	Oui	Non	Faible	Oui	Oui	★★★★
Géométrie de la main	Oui	Non	Oui	Oui	Oui	★★★★
Veines de la main	Oui	Oui	Oui	Oui	Oui	★★★★★
Oreille	Oui	Oui	Oui	Oui	Oui	★★★★★
Empreinte digitale	Oui	Oui	Oui	Oui	Oui	★★★★

— le module d'*enrôlement* consiste à créer le modèle de l'utilisateur à partir de l'*acquisition* d'une ou plusieurs données biométriques. Elles sont traitées afin de calculer une *référence* stockée dans le système. C'est cette référence qui représente un individu.
— le module d'*authentification* permet de vérifier l'identité clamée par l'individu. Le système effectue *l'acquisition* d'une donnée *requête* et la compare à la *référence* appropriée. Si les deux correspondent, l'identité est vérifiée et l'individu s'est authentifié. Sinon le système considère l'individu comme étant un imposteur.

La donnée biométrique en sortie de capteur ne peut pas être utilisée telle quelle. De nombreuses méthodes d'extraction de caractéristiques issues des domaines du traitement du signal, ou de l'Apprentissage Automatique (AA), sont nécessaires pour calculer la référence ou effectuer sa comparaison avec la requête. Par exemple, il est usuel dans les systèmes de reconnaissance par empreinte digitale d'encoder une donnée biométrique par un ensemble de points clés d'une empreinte appelés minutiae. L'identité est vérifiée à partir du moment où l'intersection des minutiae de la requête avec celles de la référence est supérieure à un seuil (généralement fixé aux alentours de 10). La difficulté étant de mettre au point des méthodes d'extraction et de comparaison de minutiae qui soient robustes pour être utilisable en environnement opérationnel. Bien entendu, il existe d'autres techniques qui ne reposent pas sur les minutae, et les autres modalités ont des techniques qui leur sont propres.

Certaines modalités, comme la Dynamique de Frappe au clavier (DDF) [28], peuvent nécessiter la collecte d'un nombre important de données avant de pouvoir calculer une référence robuste. L'avènement du deep learning [44], et notamment des réseaux siamois [18], va rapidement supprimer cette barrière. Par exemple, il est usuel de mettre en place des systèmes de reconnaissance faciale nécessitant peu de données. Un réseau siamois est appris sur une base hors ligne (*c.-à-d.* un réseau apprend à projeter les visages dans un espace de dimensions réduites où les visages d'une même personne ont une distance euclidienne faible et les visages de différentes personnes une distance plus élevée). L'enrôlement consiste à calculer la projection de la donnée d'enrôlement grâce au réseau et la stocker. La vérification consiste à calculer la projection de la requête grâce au réseau et effectuer une distance euclidienne avec celle de la référence. Si la distance est inférieure à un seuil, l'identité est vérifiée. Ainsi peu de calculs sont nécessaires pour effectuer les différentes étapes (en comparaison aux méthodes standard d'il y a une dizaine d'années) et aucune donnée brute n'est stockée dans la référence.

3.1.2. Modalités biométriques

Les données biométriques collectées appartiennent à différentes *modalités biométriques* [36] organisées en trois catégories : les modalités *physiques*, *biologiques* et *comportementales*. Les modalités *physiques* reposent sur des caractéristiques physiques qui nous sont propres telles que les empreintes digitales, le visage, les iris ou la voix (en raison de la dépendance avec les cordes vocales). Les modalités *biologiques* reposent sur des caractéristiques biologiques qui nous sont propres telles que l'ADN, les ECG ou l'odeur. Les modalités *comportementales*

reposent sur notre capacité à exécuter des actions reconnaissables telles que la démarche, la DDF ou la voix (en raison de la dépendance au stress). Dans ce chapitre, nous allons nous intéresser particulièrement à la DDF qui permet d'authentifier un individu en fonction de sa façon de taper un mot de passe au clavier. Cette modalité est peu coûteuse, car elle ne nécessite pas de capteur particulier.

Les modalités sont quantifiables à l'aide de différentes propriétés [36] :

- L'*universalité* implique que tous les individus qui doivent être identifiés avec le système possèdent la donnée biométrique utilisée par le système.
- L'*unicité* implique que la donnée biométrique doit être relativement différente d'un individu à l'autre, afin de pouvoir les différencier.
- La *permanence* implique que la donnée peut être collectée tout au long de la vie de l'individu, ou de l'utilisation du système, de l'individu.
- La *possibilité de collecte* implique que la donnée biométrique doit pouvoir être collectée et mesurée afin de pouvoir être utilisée comme moyen de comparaison.
- L'*acceptabilité* implique que les utilisateurs doivent accepter le système et être d'accord de l'utiliser.

Les performances des différentes modalités sont variables sur ces critères (voir tableau 3.1 page précédente).

On parle de biométrie floue lorsqu'une modalité ne permet pas de différencier suffisamment les individus , mais peut les regrouper en catégories ayant une sémantique propre. Par exemple, on peut utiliser le visage pour déterminer l'origine ethnique d'une personne ou son genre. De telles informations ne sont pas suffisantes pour vérifier l'identité d'un individu, cependant elles peuvent servir de support pour apporter des informations supplémentaires et améliorer la performance globale du système de reconnaissance.

3.1.3. Évaluation de systèmes d'authentification biométrique

L'évaluation d'un système d'authentification biométrique suit des règles très proches de l'évaluation d'un système d'apprentissage supervisé générique. Une base d'*apprentissage* est utilisée pour calculer la *référence biométrique* de chaque individu enrôlé dans le système. Une base de *test* est utilisée pour vérifier la performance du système. Différentes stratégies peuvent être rencontrées dans la littérature. Les individus de la base de test sont strictement les mêmes que ceux de la base d'apprentissage, ou bien la base de test contient des individus absents de la base d'apprentissage (on s'attend donc à ce qu'ils soient rejetés). Tous les exemples de tous les individus sont comparés à toutes les références biométriques (en général, chaque individu dispose du même nombre de données biométriques) ou bien certains exemples de certains utilisateurs sont comparés à certaines références biométriques.

Quelle que soit la stratégie employée, deux collections de *scores* sont collectées. Celle qui concerne les comparaisons *intraclasse* (comparaison de l'exemple d'un individu à sa référence biométrique), et celle qui concerne les comparaisons *interclasse* (comparaison de l'exemple d'un individu avec la référence d'un autre). Une décision est obtenue en comparant un score à un *seuil de décision*. Certains systèmes produisent directement une décision sans passer par l'étape intermédiaire de score.

Plusieurs métriques d'erreurs sont utilisées dans la littérature, mais les principales sont :

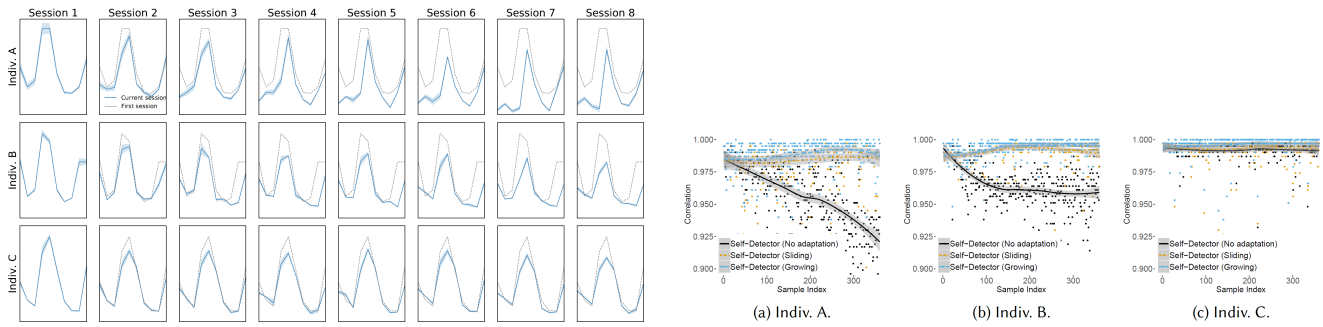
- Le Taux de Fausse Acception (TFA), ou « False Acceptance Rate (FAR) » en anglais, qui correspond au ratio d'exemples d'un individu capable d'être authentifié comme appartenant à un autre.
- Le Taux de Faux Rejet (TFR), ou « False Rejection Rate (FRR) » en anglais, qui correspond au ratio d'exemple d'un individu incapable d'être reconnu comme lui appartenant.

Ces taux peuvent être calculés globalement sur l'intégralité du jeu de test, ou individuellement pour chaque utilisateur. Dans ce dernier cas, il y a souvent un flou dans l'interprétation du Taux de Faux Rejet (TFR) de l'utilisateur A : est-ce qu'il est propriétaire de la référence biométrique ou des requêtes ? Dans nos travaux en visualisation (voir la partie III), nous faisons en sorte de ne pas avoir cette ambiguïté.

Les performances du système d'authentification peuvent dépendre d'un seuil de décision. Il est commun d'utiliser une méthode graphique pour représenter les performances en fonction de ce paramètre : la courbe ROC qui présente plusieurs variantes, dont la principale affiche (1 - TFR) en fonction du Taux de Fausse Acception (TFA). Un point particulier de cette courbe est le Taux d'Erreur Égale (TEE), ou « Equal Error Rate (EER) » en anglais, lorsque les TFA et TFR sont égaux. Il est couramment utilisé dans la littérature pour représenter les performances du système sans faire de compromis sur les rejets ou acceptations.

3.1.4. Stratégies d'amélioration des performances des systèmes d'authentification biométriques

La plupart des systèmes d'authentification biométrique souffrent de différentes lacunes impactant leur performance et leur utilisation en milieu opérationnel. Parmi les soucis principaux, nous pouvons noter :



(a) Variation de la donnée biométrique sur 3 individus en fonction de la session de collecte des données.

(b) Correlation entre la donnée biométrique et le modèle au cours du temps pour les 3 individus.

Figure 3.2. – Impact du vieillissement de la donnée biométrique sur 3 individus [61]. Les individus A et B ont clairement besoin d'utiliser un système adaptatif, tandis que l'individu C ne souffre pas du problème de vieillissement.

- le vieillissement de la donnée biométrique (plus le temps passe, plus la donnée diffère de la requête) qui impacte la performance ;
- la variabilité de la donnée biométrique qui impacte l'unicité et la performance ;
- la facilité à forger des données biométriques artificielles qui impacte l'acceptabilité et la performance.

Différentes techniques peuvent être utilisées pour contourner ces problèmes. Nous pouvons lister :

- l'utilisation de méthodes de mise à jour de la référence biométrique [61] pour diminuer les effets du vieillissement de la donnée biométrique ;
- l'utilisation de la multibiométrie [69] pour fusionner différentes sources d'informations, et lisser les erreurs des systèmes individuels ;
- l'utilisation de méthodes de détection de vie [8] pour vérifier que la donnée présentée n'est pas contrefaite.

3.1.5. La dynamique de frappe au clavier : une biométrie comportementale

La Dynamique de Frappe au clavier (DDF), ou « Keystroke Dynamics » en anglais est une modalité comportementale qui utilise la façon de taper au clavier d'un individu pour le reconnaître [28]. Il s'agit de la modalité majeure que j'ai étudiée pendant ma thèse et pour laquelle j'ai effectué quelques travaux de recherche, après mon arrivée au LaBRI, avec Paulo Henrique Pisani alors doctorant à l'université de São Paulo et Anderson Rocha professeur à l'université de Campinas. Cette modalité est dite à bas coût, car elle nécessite seulement l'utilisation d'un clavier standard, et consiste à capturer le moment où chaque touche est pressée par l'utilisateur afin de calculer différentes durées (*p. ex.* le temps de pression d'une touche ou le délai entre deux pressions successives). Cependant, il existe des variantes utilisant des claviers à capteur de pression, des micros [70], des téléphones [19], des smartphones [79], etc. Elle a été utilisée dans différents contextes tels que l'authentification continue [16] (pour détecter un changement d'utilisateur sur la machine), l'utilisation d'un mot de passe à usage unique [17], et plus souvent l'authentification statique (majoritairement avec un mot de passe partagé entre tous les utilisateurs [27], et plus occasionnellement un mot de passe unique).

En général, la référence biométrique est construite à l'aide de la capture de plusieurs dizaines de données biométriques (*c.-à-d.* de plusieurs dizaines de saisies du mot de passe). La donnée brute correspond aux listes des paires (horodatage, code de touche) lors de leur pression et de leur relâchement. De façon générale, la donnée extraite correspond à l'union de listes de durée calculées à partir des données brutes : le temps de pression d'une touche, le délai entre deux pressions successives, le délai entre deux relâchements successifs et le délai entre le relâchement d'une touche et la pression de la suivante (ce délai peut être négatif lorsque la saisie est rapide). Toutes les études n'utilisent pas la même combinaison de listes. Les méthodes les plus simples calculent différentes statistiques sur ces listes afin de générer le modèle. Ces statistiques sont utilisées pour calculer une distance avec la requête lors de la vérification.

Malgré des performances de reconnaissance globalement plus faibles que les modalités usuelles, cette modalité est facilement acceptée par les utilisateurs [1]. Il reste encore des efforts à faire sur les pratiques d'évaluation [29] afin de rendre les travaux plus facilement comparables.

La suite de ce chapitre présente deux innovations apportées à la DDF pour améliorer ses performances avec : une nouvelle méthode de mise à jour de la référence pour contourner le problème du vieillissement de la donnée, et une étude de faisabilité de l'utilisation de l'utilisation de réseaux siamois

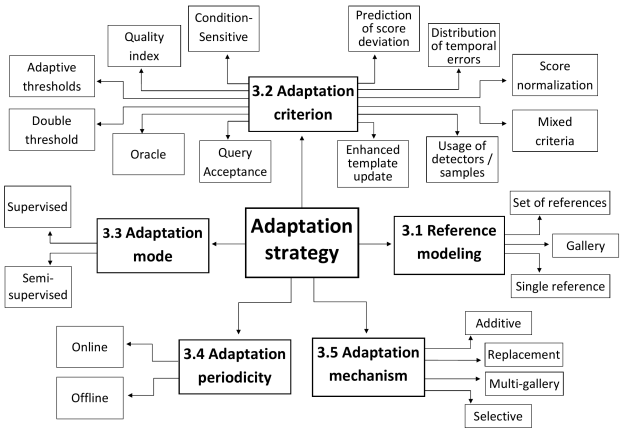


Figure 3.3. – Taxonomie des systèmes de mise à jour de la référence biométrique [61].

pour diminuer le nombre d'exemples nécessaires pour l'enrôlement.

3.2. Mise à jour de la référence biométrique : une nécessité pour les biométries comportementales

3.2.1. Contexte

Un des soucis majeurs des biométries comportementales est l'évolution de la donnée biométrique au cours du temps [31]. Dans le cas de la DDF, cela ce traduit principalement par une saisie de plus en plus rapide et stable du mot de passe. Au bout d'un certain temps, la différence est suffisamment importante pour augmenter le TFR. Il faut noter que le vieillissement de la donnée et son impact sur les performances ne sont pas le même en fonction des individus (voir la figure 3.2 page ci-contre).

Une solution à ce problème est la mise à jour de la référence biométrique ; j'ai participé à la rédaction d'un état de l'art conséquent qui détaille cette problématique et ses solutions avec des collègues de l'université fédérale d'ABC/Brésil, l'université de Sousse/Tunisie, l'entreprise Trustamp/USA, l'université de São Paulo/Brésil, et l'université de Normandie/France [61]. Nous avons catégorisé la stratégie d'un système de mise à jour de la référence suivant différents critères (voir la figure 3.3):

- la façon de *modéliser la référence biométrique*. En effet, la référence biométrique est modélisée de façon différente en fonction des systèmes (*p. ex.* une galerie qui liste les différentes données collectées, un modèle appris ou différentes agrégations) et sa modélisation impacte forcément sa mise à jour ;
- le *critère de mise à jour* de la référence biométrique indique le système de prise de décision de la mise à jour de la référence ;
- le *mode d'adaptation* permet de catégoriser deux familles majeures de systèmes : les systèmes supervisés et semi-supervisés ;
- la *périodicité* d'adaptation pour savoir si la mise à jour se fait par lot par intervalle ou individuellement en temps réel ;
- le *mécanisme d'adaptation* qui est l'algorithme proprement dit de mise à jour.

Ce manuscrit ne rentre pas plus dans les détails pour des raisons de place. Cependant, nous allons utiliser cette taxonomie pour décrire un système que nous avons développé et validé sur des données de DDF et de reconnaissance par la démarche par signaux de smartphone. Il s'agit d'une approche où le modèle est encodé par une galerie, le critère d'adaptation prend en compte l'acceptation d'une requête aussi bien que son rejet, et l'adaptation est semi-supervisée, hors-ligne et avec différents mécanismes d'adaptation.

3.2.2. Méthode améliorée de mise à jour

Les méthodes de mise à jour de la référence de l'état de l'art n'utilisent que les requêtes acceptées lors de l'utilisation du système : si une requête est acceptée, elle peut éventuellement prendre part au processus de mise à jour de la référence. Dans l'approche que nous proposons [60], nous utilisons également les données rejetées afin de construire un modèle représentant les imposteurs, et se servir à la fois du modèle de l'utilisateur et des imposteurs lors de l'authentification : si une requête est acceptée, elle peut éventuellement prendre part au processus de mise à jour de la référence ET si une requête est rejetée, elle peut prendre part à la mise à jour du modèle des imposteurs. Cette approche novatrice, illustrée en figure 3.4 page suivante a permis d'augmenter les performances générales.

Le système adaptatif fonctionne de façon classique au début de son fonctionnement (« Self-Update » dans la littérature anglophone). Cependant, lorsqu'une certaine quantité de données biométriques étiquetées comme appartenant à des imposteurs (*c.-à-d.* des individus ont essayé de s'authentifier, et le système a considéré qu'il s'agissait d'intrus) ont été collectées, le système adaptatif passe dans son mode avancé. Un modèle est créé avec les exemples d'imposteurs et ce modèle est également mis à jour lors de la collecte de nouvelles données considérées comme appartenant à des imposteurs. Le modèle de l'utilisateur authentique et celui des imposteurs sont tous les deux mis à contribution lors de l'authentification d'une nouvelle requête. Nous avons testé 4 stratégies de combinaisons différentes (voir le papier pour plus de détails [60]). La galerie de données d'imposteurs peut être également mise à

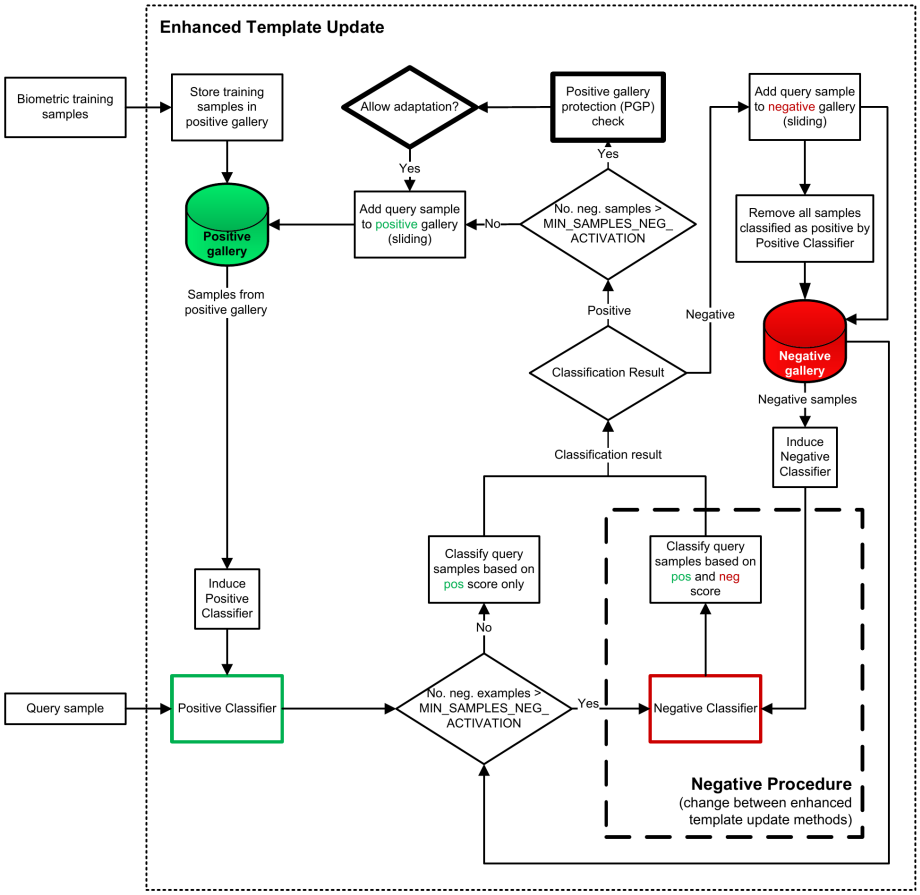


Figure 3.4. – Contrairement aux autres approches de la littérature, Enhanced Template Update [60] tire parti des données collectées considérées comme étant des données d'imposture.

contribution lors de la mise à jour de la galerie positive afin de limiter un peu plus le risque d'inclure des données d'imposteurs considérées comme étant légitimes (voir Positive Gallery Protection Check dans le papier [60]).

L'approche a été testée sur plusieurs jeux de données d'authentification par DDF et de reconnaissance de la démarche par téléphone portable. Notre approche c'est montrée plus efficace que celles de l'état de l'art dans la majorité des cas d'utilisation en termes de TFR. Le TFA a légèrement augmenté sur certaines combinaisons.

3.3. Diminution de la quantité de captures pour l'enrôlement en dynamique de frappe au clavier

3.3.1. Contexte

La DDF souffre d'une grande variabilité intraclasse : la façon de saisir un mot de passe est fortement variable au cours du temps. Pour contourner ce problème, la plupart des systèmes s'attendent à ce que l'utilisateur saisisse de nombreuses fois le mot de passe afin de capturer un maximum de variabilité. Ce nombre peut aller jusqu'à plusieurs dizaines [40], bien que certaines méthodes peuvent être performantes avec moins d'une dizaine de saisies [27]. Saisir plusieurs fois le mot de passe est relativement fastidieux pour l'utilisateur. Il est donc nécessaire d'identifier des stratégies qui permettent à la fois de diminuer le nombre de saisies d'enrôlement et d'améliorer les performances de reconnaissance.

Un constat général que l'on observe dans la littérature est que les modèles par Apprentissage Profond (AP) sont de plus en plus couramment utilisés sur différents domaines d'application. Un prérequis important est de disposer de masses de données pour pouvoir les entraîner. Nous ne disposons pas de ces masses de données dans la cas de la DDF. Cependant, une famille particulière de Réseau de Neurones Profond (RNP) convient à une telle approche : les réseaux siamois [18], ou « Siamese Network » en anglais. Nous avons vérifié si cette

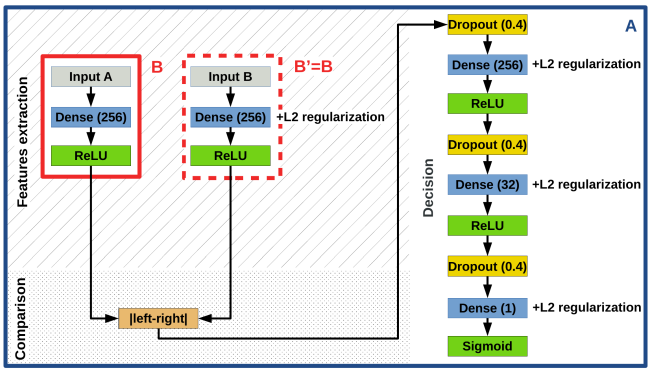


Figure 3.5. – Description du réseau siamois utilisé dans notre approche [30]. Chaque branche B et B' extraient les caractéristiques des exemples à comparer, une couche calcule leur différence et le reste du réseau (A) calcule la distance.

famille d'architecture est compatible avec l'utilisation de la DDF.

3.3.2. Réseau siamois pour la dynamique de frappe au clavier

Un RNP a besoin d'une quantité conséquente d'exemples d'apprentissage lors de l'entraînement. Cette quantité est souvent trop faible en DDF pour calculer un modèle efficace. Le réseau siamois [18] a la particularité de prendre deux données en entrée (au lieu d'une pour les réseaux standards) et de calculer une distance entre elles, au lieu de probabilité d'appartenance à des classes définies. Il est donc appris avec des paires d'exemples. Comme la combinatoire fait qu'il est possible de générer de nombreuses paires, intra et interclasses, avec une base d'apprentissage de DDF, il est possible d'apprendre efficacement un tel réseau. En fait, la problématique ouverte sur ce type d'architecture correspond à la sélection efficace de ces paires. Avec Anderson Rocha de l'Université de Campinas/Brésil, nous avons prouvé [30] que cette approche est réaliste dans le contexte de la DDF.

Reste à la communauté de définir des protocoles d'évaluation justes pour ce genre de technique, et de mettre au point des architectures spécifiques à ce type de données afin d'améliorer leur performance. En effet, des couches spéciales sont nécessaires pour effectuer une extraction des caractéristiques depuis les données brutes qui a un sens dans le cas de la DDF (notamment les informations de durées).

La stratégie que nous avons employée est d'entraîner un réseau unique pour tous les utilisateurs. Il y a plusieurs avantages à cette approche :

- gain en temps de calcul : un seul modèle doit être appris ;
- gain en capacité de stockage : un seul modèle doit être stocké ;
- gain en ergonomie : l'enrôlement d'un utilisateur est relativement rapide, car quelques exemples doivent être collectés, et aucun calcul ne doit être effectué. Il suffit de stocker les captures dans la *galerie* de l'utilisateur (éventuellement après extraction des caractéristiques par le réseau afin de ne pas stocker la donnée brute).

De cette façon, le réseau se contente de consommer deux données biométriques et de calculer une distance entre eux. L'architecture est la suivante (voir la figure 3.5) :

- deux branches strictement identiques servent à extraire des caractéristiques depuis leurs données en entrée. Dans ce travail préliminaire, nous ne fournissons pas les données brutes aux réseaux, mais des caractéristiques extraites de DDF standard. Sa profondeur et sa largeur ont été choisies de telle façon à maximiser les performances ;
- une différence absolue entre les deux vecteurs extraits est calculée ;
- un réseau multicouche calcule une distance depuis cette différence.

Un tel réseau peut être utilisé de différentes manières pour vérifier l'identité d'un individu. Nous avons testé les approches suivantes :

- chaque exemple de la galerie de l'individu clamé est comparé à la requête. Le score final de comparaison correspond à la moyenne des scores obtenus ;
- chaque exemple de la galerie de l'individu clamé est comparé à la requête. Le score final de comparaison correspond à la moyenne des k meilleurs scores obtenus ;
- seule la partie extraction des caractéristiques du réseau est utilisée. Les distances usuelles utilisées en DDF sont utilisées sur les données extraites obtenues.

L'évaluation a été effectuée sur la base CMU [40] où 50 participants ont saisi le mot de passe *.tie5Roanl* 400 fois durant 8 sessions. Un sous-ensemble des participants est utilisé pour calculer le réseau, tandis que le reste est utilisé pour calculer les performances du système obtenu. Une validation croisée à 3 ensembles est utilisée pour faire la séparation.

Les résultats obtenus [30] montrent que :

- de meilleures performances sont obtenues avec un réseau d'extraction peu profond, mais large ;
- augmenter le nombre d'exemples d'enrôlement dans la galerie augmente systématiquement les performances de reconnaissance ;

— lorsqu'un seul exemple est stocké dans la galerie, les performances du réseau sont systématiquement meilleures que celles des méthodes de l'état de l'art. Elles restent cependant trop faibles pour être utilisées en milieu opérationnel contraint ;
— quel que soit le nombre d'exemples dans la galerie, et la variante du réseau, le réseau obtient toujours de meilleures performances que les méthodes de l'état de l'art.

Les travaux en DDF utilisent les mêmes caractéristiques depuis plusieurs décennies. Leur originalité réside principalement dans la technique utilisée pour créer un modèle et comparer une requête à une référence à l'aide de ces caractéristiques. Nos travaux sur les réseaux siamois montrent qu'il est intéressant de travailler également sur l'extraction des caractéristiques (c'est ce que fait principalement le réseau au final) et qu'il est toujours possible d'en trouver des plus pertinentes.

Chapitre 4.

Attribution de caractéristiques pour les réseaux de neurones profonds

4.1. Introduction à l'explicabilité pour l'explication des réseaux de neurones profonds

Les techniques par Apprentissage Automatique (AA) peuvent souffrir à différents niveaux de problème d'interprétabilité et d'effet boite noire : nous ne savons pas toujours expliquer les raisons de leurs échecs et succès, alors que nous sommes capables d'évaluer leur efficacité. Les notions d'interprétabilité et d'explicabilité sont souvent utilisées de façons interchangeables pour désigner les méthodes concernant ce problème. Nous préférons définir l'interprétabilité comme l'ensemble des techniques qui permettent à un modèle d'expliquer par lui-même les raisons de sa réponse, et l'explicabilité comme l'ensemble des techniques qui permettent d'expliquer après coup les raisons de la réponse du modèle. C'est ce dernier point qui nous intéresse ; on parle de « posthoc analysis » dans la littérature anglophone.

Le chapitre 13 présente les approches orientées visualisation pour l'explication, tandis que ce chapitre discute des approches utilisées par la communauté en AA dans le cas de l'analyse de réseaux de neurones profonds. Les approches d'explications développées dans la communauté AA peuvent être catégorisées en 5 familles principales : l'attribution de caractéristiques d'entrées, la détection des caractéristiques apprises, l'explication par exemple ou contre-exemple et l'extraction du comportement du modèle. Même s'il existe des méthodes globales, de façon générale, ces méthodes s'intéressent à un exemple spécifique pour lequel une explication est attendue. Les techniques employées dans le chapitre 13 permettent plus aisément de faire des explications de modèles plutôt que des explications d'instances. Les sections suivantes listent les types d'explications qui peuvent être rencontrées dans la littérature, tandis que la figure 4.1 page suivante les illustre.

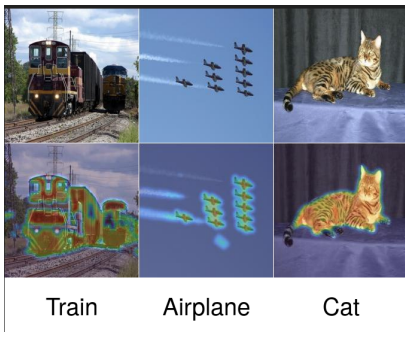
4.1.1. Attribution de caractéristiques

Les méthodes *d'attribution de caractéristiques d'entrée* répondent à la question « Sur quoi le modèle se concentre-t-il pour calculer la prédiction ? » et donnent des indices sur la question « Pourquoi le modèle renvoie-t-il cette sortie spécifique ? ». Ce type de méthode a été beaucoup plus étudié dans la littérature que les autres (voir figure 4.2 page suivante). En général, les méthodes calculent un score à chaque caractéristique de la donnée d'entrée : un score important implique une contribution importante pour le modèle [78]. Elles peuvent être classées suivant la procédure suivie pour calculer ce score. Nos contributions dans ce chapitre portent sur ce type d'explication. Dans les faits, toutes les caractéristiques d'entrée ne se voient pas attribuer un score, ce sont plutôt des unités topologiques qui se voient attribuer un score : une image est représentée par un tenseur 3D (x, y et canaux de couleur), tandis que son explication est représentée par un tenseur 2D (x,y) qui indique l'importance des pixels en totalité (et pas des composants des pixels).

Une première famille de méthodes utilise des modèles de substitution, capables de reproduire le comportement du modèle original pour un exemple donné. C'est ce modèle de substitution qui est utilisé pour générer l'explication. Généralement, ces méthodes peuvent fonctionner sur des modèles de type boite noire. LIME [64] utilise des classifieurs linéaires ou des arbres comme modèle de substitution. SHAP [48] est également une méthode couramment utilisée dans la littérature.

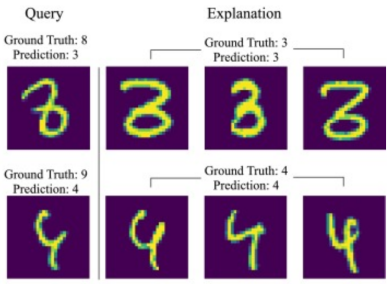
La seconde famille cherche à perturber les données en entrée du modèle afin d'analyser sa différence de comportement [25, 35]. Les perturbations consistent à supprimer ou modifier des valeurs d'entrée ; plus la sortie change plus la caractéristique modifiée est importante.

La troisième famille comprend les méthodes qui construisent des explications en se reposant sur les gradients. Grad-CAM [73] et Integrated Gradients [78] sont des exemples de cette famille. Elles reposent sur l'intuition que la magnitude des gradients indique l'emplacement des limites de décisions

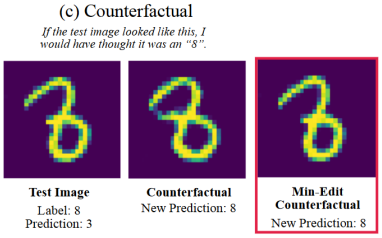


(a) Attribution de caractéristiques. H20 [62] est illustré dans cet exemple : la carte de chaleur indique les pixels importants pour la prise de décision

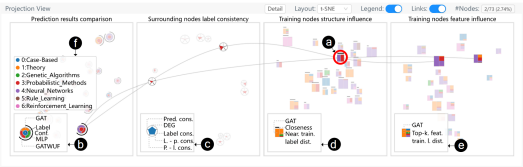
(b) Détection de caractéristiques apprises. Une méthode d'optimisation génère des images artificielles qui activent fortement des filtres sélectionnés [57]



(c) Explication par exemple. Ici l'utilisateur essaie de comprendre pourquoi son exemple est mal classé à l'aider d'exemples similaires [38]



(d) Explication par contre-exemple. [39] est illustré dans cet exemple



(e) Explication par extraction du comportement du modèle. Gnnlens [37] est illustré dans cet exemple ; de tels outils nécessitent un apprentissage et de l'intéraction

Figure 4.1. – Illustration des différentes familles d'explications. Les explications par attribution de caractéristiques (a), détection de caractéristiques (b), exemple (c), contre-exemple (d) sont généralement représentées à l'aide d'images statiques, tandis que l'extraction du comportement du modèle (e) nécessite en général une application d'Analyse Visuelle et Interactive (AVI).

sensibles à la modification. Ces limites identifient les zones où peu de changements dans les entrées produisent une grande modification dans la sortie. Ainsi, ces méthodes cherchent à relier ces zones aux caractéristiques responsables.

La dernière catégorie de méthodes exploite les propriétés du réseau ou utilise des règles manuelles de propagation du réseau pour propager les informations des dernières couches vers la donnée d'entrée sans utiliser les gradients. Cette catégorie inclue : Layer Relevance Propagation [11] qui propage les scores de prédiction depuis la dernière couche vers l'entrée en utilisant une série de règles de propagation [54], le réseau de déconvolution [83, 84] qui projette la carte d'activation d'une couche de convolution vers l'espace d'entrée en effectuant les opérations inverses des couches intermédiaires, la méthode de Martens et Provost [51] qui extrait des explications de taille minimum (*p. ex.* une liste de mots) en supprimant des caractéristiques (*p. ex.* des mots) itérativement de l'exemple à expliquer (*p. ex.* un document composé de mots), ou les méthodes qui exploitent directement les activations [26, 85]. Certaines techniques utilisent les informations issues d'attentions calculées par le réseau. Dans ce cas, un poids important d'attention suggère que les caractéristiques correspondantes ont également une importance forte. Ces valeurs d'attention peuvent être extraites depuis les dernières couches du réseau [77], des couches spécifiques [68] ou agrégées depuis le réseau [2].

4.1.2. Détection de caractéristiques apprises

La *détection des caractéristiques apprises* associe des ensembles de concepts aux neurones, aux groupes de neurones, ou aux

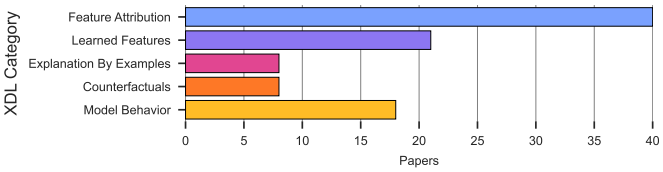


Figure 4.2. – Distribution des méthodes d'explications dans les papiers analysés par notre état de l'art [42]. L'attribution de caractéristiques est clairement surreprésentée.

couches en fonction des caractéristiques qu'ils peuvent reconnaître. Elles répondent à la question « Qu'a-t-il appris au cours du processus d'apprentissage ? ». Plusieurs méthodes se basent sur les activations [32] et espaces latents [83] pour extraire de telles informations. On peut noter la méthode de maximisation des activations [32] dont le concept a été amélioré et généralisé par Olah et al. [58], ou la caricaturisation [50]. Certains travaux utilisent l'espace latent généré par le réseau pour extraire des règles [10] ou des arbres de décision [20] qui approximent le comportement du modèle et sont auto descriptifs (*c.-à-d.* lire l'ensemble des règles ou l'arbre explique les caractéristiques recherchées). Une autre possibilité consiste à associer des concepts aux couches ou neurones [13]. Par exemple TCAV [41] permet de décrire la sensibilité du modèle à certain concepts haut niveau (définis par l'utilisateur avec un apprentissage additionnel). Certains travaux essaient également de reproduire la donnée d'entrée afin de visualiser ce qui a été pertinent pour la reconnaissance en utilisant les gradients [49] ou des réseaux générateurs de données [55].

4.1.3. Explication par exemple factuel

L'*explication par exemple* consiste à rechercher des exemples (issus d'un jeu de données spécifique) qui produisent une sortie similaire à celui analysé afin de permettre à l'utilisateur d'inférer les similitudes dans l'espace d'entrée [14]. Ce type d'explication est similaire au raisonnement par analogie des humains [45]. Ces méthodes répondent à la question « Quels échantillons sont considérés comme similaires par le modèle? ». Une propriété de ces explications est qu'elles doivent être dans un format compréhensible par les humains (*c.-à-d.* texte, image, petit vecteur). Les méthodes de cette catégorie utilisent régulièrement des techniques de k plus proches voisins dans l'espace latent par rapport à l'exemple d'entrée. Mais elles peuvent également chercher à générer des données artificielles ayant une réponse similaire.

4.1.4. Explication par contre-exemple

L'*explication par contre-exemple* consiste à rechercher des exemples, proches de la donnée à expliquer, appartenant à une autre classe de telle façon que le nombre d'éditions pour passer de la donnée de base à son contre-exemple soit minimal [43]. Ces contre-exemples sont donc proches au vu du modèle, mais produisent une sortie différente ; ils répondent à la question : « qu'est-ce que je dois changer pour obtenir un résultat différent ? » et sont particulièrement utiles pour analyser des recours (*c.-à-d.* les actions requises pour modifier favorablement la décision des modèles [80]). Les contre-exemples peuvent être générés automatiquement [63] en satisfaisant des contraintes d'édition ou extrait depuis un jeu de données [4].

4.1.5. Extraction du comportement du modèle

Les méthodes *d'extraction du comportement du modèle* cherchent à identifier comment fonctionnent les modèles en extrayant des informations depuis différentes sources (de l'entrée à la sortie en passant par les poids ou les activations). Elles répondent à la question « Comment le modèle réagit-il dans une situation donnée ? », où la situation est un ensemble d'exemples similaires ou l'ensemble des données. Ces techniques combinent de la recherche de motifs, de l'interaction avec l'utilisateur, et des méthodes des autres types afin d'extraire les points communs du modèle en fonction de situation spécifique. En exemples de cette catégorie, nous pouvons trouver les méthodes qui utilisent les motifs d'activation de la dernière couche dense pour comprendre et contrôler la sortie des modèles génératifs [12], les méthodes qui combinent les motifs d'activation et les caractéristiques apprises pour expliquer les erreurs d'explication sur des contre-exemples [47], ou des méthodes qui combinent des motifs entre les entrées et sorties du modèle avec les caractéristiques extraites pour extraire les politiques suivies par un agent appris avec une technique d'apprentissage par renforcement [81]. Ces méthodes sont donc majoritairement traitées par la communauté visualisation, car la communauté d'apprentissage s'est spécialisée dans les approches où les résultats sont statiques. Nos contributions dans le chapitre 13 portent sur ce type d'explications.

4.2. Attribution de caractéristiques pour réseau de neurones sans prérequis de classe

En collaboration avec des membres des équipes « Image&Son », ainsi que « Combinatoire et Algorithmes » du LaBRI, nous avons proposé deux méthodes d'explication par extraction de caractéristiques.

La première, Feature Explanation Method (FEM), est une approche rapide qui utilise uniquement la carte d'activation générée par la dernière couche de convolution. Elle ne repose pas sur l'utilisation de gradients comme GradCAM [72], ou de calcul depuis la fin du réseau vers le début comme LRP [54]. Contrairement à la plupart des méthodes de la littérature, elle est agnostique de la classe d'intérêt : elle utilise donc ce que le réseau a globalement calculé. Ce point est important à signaler, car les méthodes de la littérature sont en général spécifiques pour une classe cible.

La seconde, Multi-Layer Feature Explanation Method (ML-FEM), considère qu'utiliser uniquement la dernière couche de

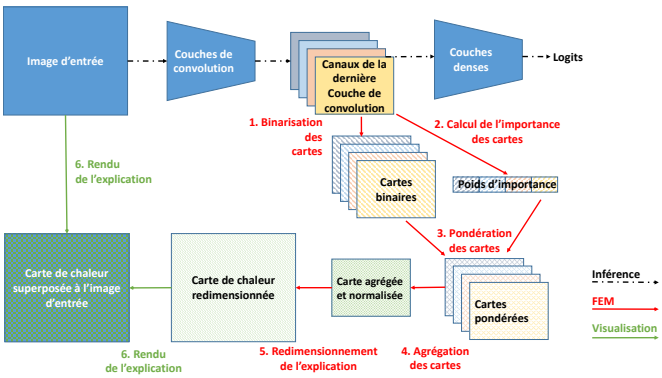


Figure 4.3. – Etapes nécessaires pour calculer l'explication avec FEM. Seules les activations issues de la dernière couche de convolution sont utilisées. Elles sont individuellement binarisées puis combinées afin de générer la carte finale.

convolution impacte trop la résolution de l'explication. Elle calcule donc FEM sur plusieurs couches, puis effectue la fusion de ces cartes afin de générer l'explication finale.

4.2.1. Explication de caractéristiques avec FEM

FEM (Feature Explanation Method) [26] cherche à mettre en avant les données d'entrée importante pour la classification (*c.-à-d.* les pixels dans le cas d'une classification d'image). Elle diffère de LIME, car elle est dépendante du modèle et ne fonctionne que dans un contexte de Réseau de Neurones Profond (RNP). Elle diffère de Gradcam, car elle ne nécessite pas de calculer les gradients; elle est donc plus rapide. Elle diffère de LRP, car elle est indépendante de la classe résultat (nous ne devons pas spécifier une classe d'intérêt).

Nous l'avons définie pour fonctionner dans un contexte de Réseau de Neurones (RN) convolutionnel, mais elle peut se généraliser à des réseaux denses ; *p. ex.* la méthode a été vendue dans un papier qui utilise un réseau multimodal de classification de vidéo~[26].. Comme pour GradCam, FEM repose sur la détection de caractéristiques importante au niveau de la dernière couche de convolution (*c.-à-d.* les activations importantes calculées au niveau de la dernière couche de convolution). Cet emplacement dans le réseau se justifie pour plusieurs raisons :

— il s'agit des informations extraites par le réseau et utiles pour sa partie classification qui se déroule au niveau des dernières couches denses après la linéarisation de cette carte d'activation ;
— il s'agit de la dernière partie du réseau où il reste encore un peu d'information topologique (à partir du moment où le réseau n'est pas trop profond avec du « pooling »), la suite est constituée uniquement de couches denses pour lesquelles les coordonnées 2D des pixels sont perdues.

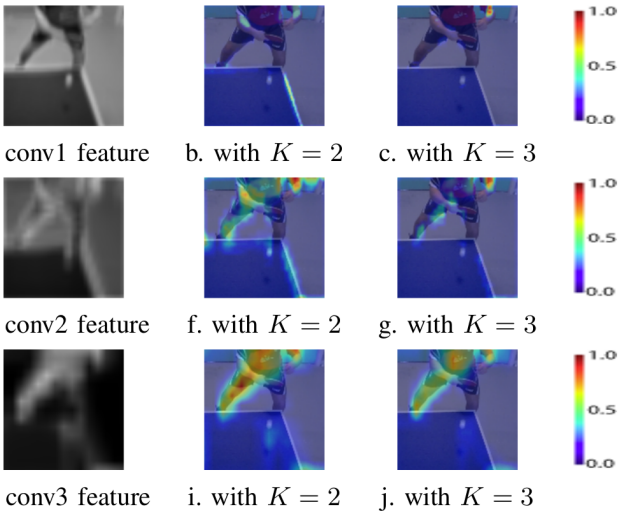
Le calcul de l'explication se fait suivant 5 étapes :

1. La première étape consiste à collecter l'ensemble des activations, appelé *carte de caractéristiques*, calculées au niveau d'une couche convolution, après l'application de la fonction d'activation. Nous partons du principe que cette activation est une activation ReLU, et donc que toutes les valeurs sont positives. Nous disposons donc d'une carte par couche de convolution, et seule l'une d'entre elles est utilisée par FEM (la dernière). Chaque carte est décomposée en canal, avec un canal par filtre de convolution.
2. Lors d'une seconde étape, nous calculons une version binaire de chaque canal. Nous faisons l'hypothèse que les données d'activation d'une couche suivent une distribution gaussienne [1]. Les caractéristiques importantes sont codées dans la queue droite de la gaussienne (il s'agit donc des événements rares à forte activation) et peuvent donc être collectées grâce à la règle de $K\sigma$ afin de construire une carte binaire d'importance b_l^c pour le canal c de la couche l,. Avec K un paramètre indiquant la quantité d'activation à filtrer, a_l^c les activations du canal c de la couche l, μ_l^c sa valeur moyenne et σ_l^c son écart type :

$$b_l^c(x,y) = \begin{cases} 1 & \text{si } a_l^c(x,y) \geq \mu_l^c + K * \sigma_l^c \\ 0 & \text{sinon} \end{cases} \tag{4.1}$$

3. Les réseaux de neurones profonds sont souvent définis de façon empirique, et il est connu que le nombre de filtres (*c.-à-d.* le nombre de canaux) de convolution est surestimé. Pour cette raison, tous les filtres n'ont pas une importance équivalente dans la décision. Nous prenons cet aspect en compte en calculant un poids d'importance pour chaque canal afin de les pondérer. La mesure d'importances w_l^c correspond à la valeur moyenne des activations pour un filtre donné (*c.-à-d.* $w_l^c = \mu_l^c$).
4. Finalement, la carte d'explication e est construite en calculant une combinaison linéaire des canaux binaires en utilisant leur poids d'importance, suivi d'une normalisation MinMax

1. Nous savons pertinemment que cette hypothèse est fausse, notamment à cause des activations de type ReLU, mais cela nous permet de simplifier l'approche. Cependant, d'après les travaux de master d'Alexey Zhukov, utiliser la vraie distribution n'apporte pas de meilleurs résultats.



a. conv1 feature b. with $K = 2$ c. with $K = 3$

e. conv2 feature f. with $K = 2$ g. with $K = 3$

h. conv3 feature i. with $K = 2$ j. with $K = 3$

Figure 4.5. – Illustration de la différence de résultat de FEM appliqué avec différente valeur de K pour le seuillage ainsi qu'à différentes profondeurs du réseau.

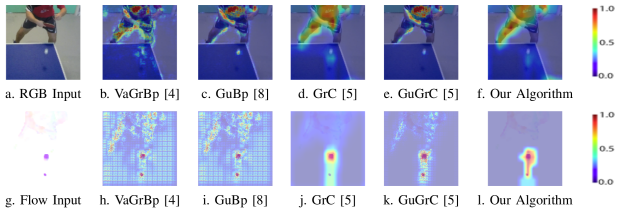


Figure 4.4. – Illustration des zones mises en avant dans une portion de vidéo par les différentes méthodes d'explication appliquées sur les deux branches du réseau. L'action sélectionnée pour les méthodes la nécessitant est : Defensive backhand block.

$$e_l^0(x,y) = \sum_c b_c^l(x,y) w_l^c$$

$$e(x,y) = \frac{e^0(x,y) - min\left(\left\{e^0(i,j), \forall_{i,j}\right\}\right)}{max\left(\left\{e^0(i,j), \forall_{i,j}\right\}\right) - min\left(\left\{e^0(i,j), \forall_{i,j}\right\}\right)}$$

5. et d'un redimensionnement à la taille de l'image initiale avec une interpolation linéaire.

L'explication est rendue à l'utilisateur de façon classique en construisant une carte de chaleur superposée à l'image initiale.

FEM a été évaluée dans un contexte d'authentification d'action sportive depuis des vidéos de personnes jouant au tennis de table [52]. Ses résultats ont été comparés à différentes méthodes de l'état de l'art Vanilla Gradient-Based Propagation (VanillaGrad) [74], Guided Back-propagation (GuidedBackProp) [75], Grad-CAM (GradCam) [72], et Guided Grad-CAM (GuidedGradCam) [72] (voir la figure 4.4).

Nous avons également vérifié si utiliser les caractéristiques calculées dans la dernière couche, plutôt qu'une autre est justifiée, ainsi que l'impact de K pour le seuillage. La figure 4.5 illustre ces différences et doit être comparée avec la figure 4.4 qui utilise la dernière convolution et $K = 1$. On peut observer que les résultats sont les plus cohérents avec l'application de FEM sur la dernière couche de convolution, car elle prend en compte à la fois la balle et le joueur, et que $K = 3$ est plus intéressant, car cela permet de se focaliser sur les détails en mouvement (capturés grâce à la convolution 3d). Nous verrons dans section 4.2.2 qu'il y a toujours un intérêt à utiliser les informations fournies par les couches intermédiaires.

L'article de FEM [7] propose plus de détails sur les expériences effectuées. On peut cependant retenir que FEM est une approche intéressante, car peu coûteuse et donnant des résultats similaires, mais non identiques aux méthodes de l'état de l'art. Nous n'avons pas initialement quantifié si cette différence est en faveurs ou défaveur de FEM, mais la partie suivante supposerait que si.

4.2.2. Amélioration de la résolution de l'explication avec ML-FEM

FEM permet de générer une carte de saillance des informations importantes à la décision d'un système de classification d'image. Par configuration, elle se limite à l'information apportée par la dernière couche de convolution, car nous avions observé que c'était la plus pertinente. Cependant, même si les informations obtenues dans les couches précédentes sont individuellement de moindre qualité, nous faisons le postulat que les fusionner permettrait d'améliorer l'explication finale. En effet, les informations produites par chaque couche sont différentes et complémentaires (voir figure 4.6 page suivante). Les premières couches extraient des concepts abstraits, tandis que les dernières couches détectent des concepts plus spécifiques. Pour cette raison, nous proposons MLFEM [15] qui est son évolution, et effectue la fusion de plusieurs explications FEM calculées à différentes profondeurs du réseau.

L'approche décrite dans la figure 4.7 page suivante se résume de la façon suivante :

1. plusieurs explications intermédiaires FEM sont calculées en sortie de différentes couches de convolution ;
2. les cartes FEM sont agrégées afin de produire l'explication finale ;
3. l'explication finale est rendue de façon standard.

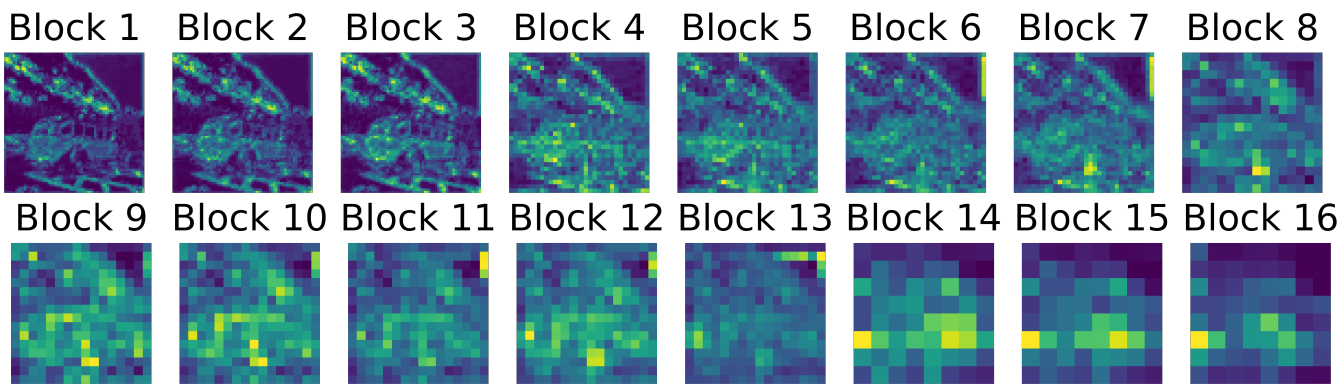


Figure 4.6. – Application de FEM à la fin de chaque bloc d'un réseau conventionnel de type ResNet50 faisant l'inférence d'une image quelconque. On voit clairement que la résolution est meilleure pour les premières couches, mais également que la carte est moins informative.

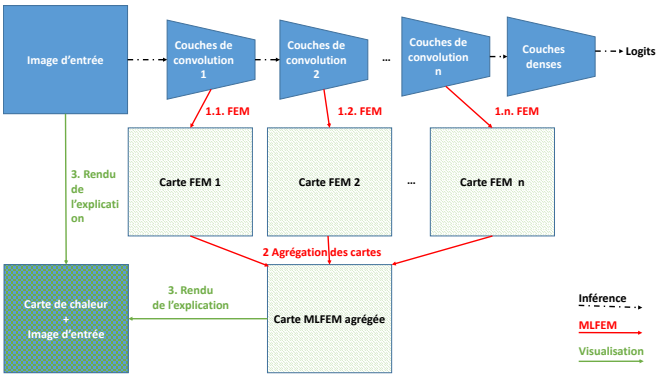


Figure 4.7. – Étapes nécessaires pour calculer l'explication avec MLFEM. FEM est appliquée en sortie de plusieurs couches de convolution de résolution variable. Les explications sont ensuite fusionnées.

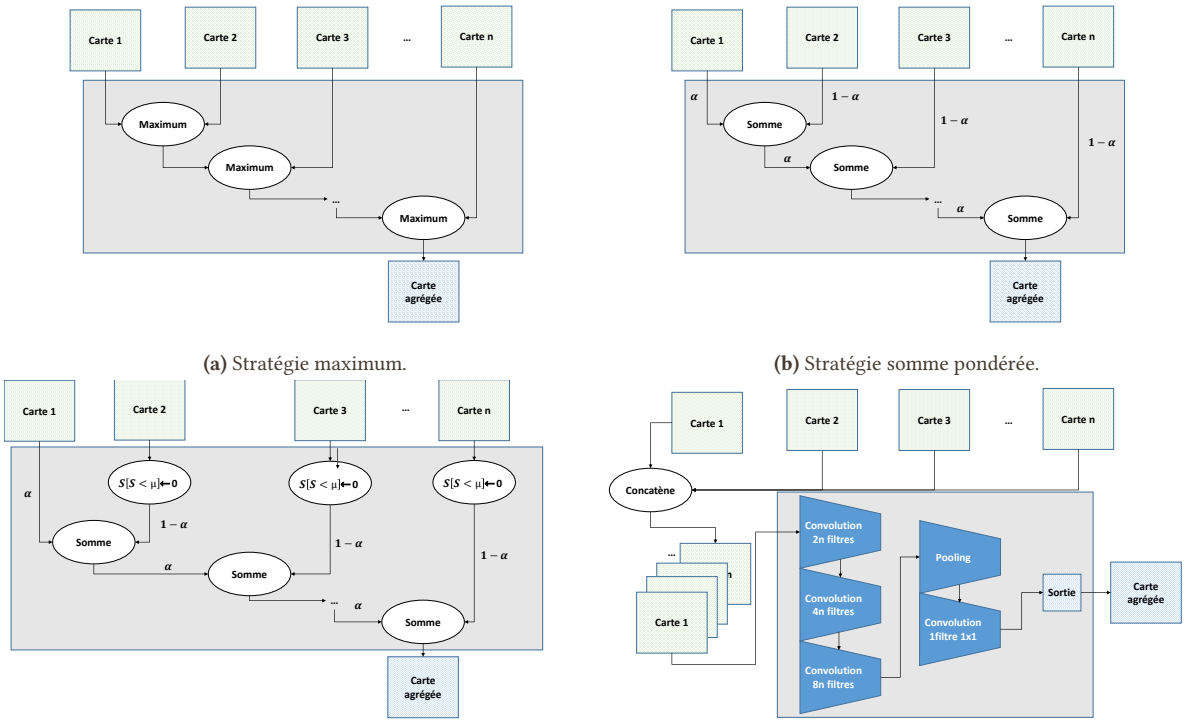
Ce qui amène à deux questions : (1) comment choisir les couches de convolution sur lesquelles appliquer FEM ? Et (2) comment agréger les cartes ? La première question est, à notre avis, dépendante du problème. C'est donc à l'utilisateur de MLFEM de choisir l'approche la plus pertinente (*p.ex.* en sortie de blocs logiques du réseau, plutôt qu'en sortie de chaque couche). La deuxième question nécessite de l'expérimentation pour identifier les approches pertinentes.

Nous avons testé (figure 4.8 page ci-contre) des méthodes de fusion algébrique, ainsi qu'une autre qui repose sur l'utilisation d'un réseau de fusion mais nécessite une vérité terrain contrairement à toutes les approches de la littérature. Pour les méthodes algébriques, nous avons choisi d'effectuer la fusion en accumulant les résultats de fusion depuis le début du réseau vers la fin: $S = fusion\,(S_4, fusion\,(S_3, fusion\,(S_1, S_2)))$ afin de n'avoir que des opérateurs qui prennent deux arguments. Le premier argument est systématiquement une carte FEM, tandis que le deuxième est une carte FEM pour le tout premier calcul et un résultat intermédiaire pour les autres. Les opérateurs algébriques sont les suivants :

- Le *maximum* qui calcule le maximum, élément par élément, entre deux cartes $S = maximum(S_a, S_b)$;
- La *somme pondérée* qui effectue la somme pondérée, élément par élément, entre deux cartes (l'une d'entre elles a donc plus d'importance $S = somme(\alpha.S_a, (a - \alpha).S_b)$);
- L'opérateur *meilleur* qui ne sélectionne que les caractéristiques importantes de la carte à ajouter $S = somme(S_a, S_m)$ tel que $S_m(x, y) = \begin{cases} S_b(x, y) & \text{si } S_b(x, y) \geq \mu_{S_b} \\ 0 & \text{sinon} \end{cases}$

Une approche supplémentaire utilise un RN et consiste à empiler les cartes d'activation avec pour objectif de reproduire une carte de densité de fixation de regard obtenue dans des expérimentations psycho-visuelles où des humains ont observé des images avant de les classer. Ces cartes de densité sont obtenues en général à l'aide de données issues de eye-tracking auxquelles un noyau gaussien a été appliqué. A titre d'exemple, la figure 4.9 page suivante présente les cartes de densité de fixation de regard pour quelques images de la base Mexculture. Le réseau contient 3 couches de convolutions suivi d'un pooling ce qui génère un tenseur 8 fois plus profond et 4 fois moins grand que le tenseur d'entrée. Une somme pondérée est effectuée sur ce dernier tenseur, suivit d'un redimensionnement pour obtenir la taille de l'image d'origine.

L'évaluation est décrite en détails dans la publication dédiée à MLFEM [15]. On peut retenir que la méthode de fusion ce basant sur un RN est plus performante que les autres. C'est normal, étant donnée qu'elle est apprise à l'aide d'une base de donnée contenant des images avec fixation ; elle sait donc générer des cartes proches de ce que l'on attend. Cependant, si elle est capable de les générer, c'est également car l'information nécessaire est déjà contenue dans les cartes FEM à



(a) Stratégie maximum.

(b) Stratégie somme pondérée.

(c) Stratégie meilleur.

(d) Stratégie NET.

Figure 4.8. – Illustration des stratégies de fusion testées pour MLFEM.

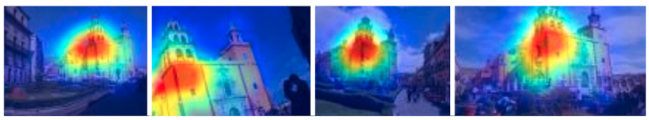


Figure 4.9. – Illustration de cartes de densité de fixation de regard pour quelques images de la base Mexculture.

agréger. FEM peut extraire de l'information pertinente, même dans les premières couches. GradCam est en général moins performante que les autres méthodes : elle ne met pas en avant des informations proches de la carte de fixation. FEM n'est pas toujours battue par les méthodes de fusion simple.

La véracité des prédictions ne semble pas avoir d'effet significatif sur la qualité de l'explication. C'est compréhensible étant donné que la méthode n'est pas sensible à la classe d'intérêt

Chapitre 5.

Discussion sur l'analyse de données et l'intelligence artificielle

Cette partie a présenté quelques travaux sur l'analyse de données et l'Apprentissage Automatique (AA) auxquels j'ai contribué avec plusieurs collègues du LaBRI. J'ai ainsi pu consolider mes compétences acquises lors de mon doctorat et mon postdoctorat. Ce chapitre effectue la synthèse de mes travaux en authentification biométrique et attribution de caractéristiques pour les réseaux de neurones profonds. Il décrit également brièvement des travaux pertinents pour cette partie, mais non abordés.

5.1. Authentification biométrique

5.1.1. Mise à jour de la référence biométrique

J'ai continué à travailler sur la référence biométrique en proposant une nouvelle méthode de mise à jour (voir la section 3.2.2) qui a l'originalité d'utiliser les requêtes rejetées dans sa procédure de mise à jour. Avec plusieurs collègues de différentes universités, j'ai également participé à la diffusion d'un état de l'art conséquent sur le sujet [61]. Je poursuis actuellement ma collaboration avec Paulo Henrique Pisani de l'université ABC/Brésil et un de ses étudiants en master sur la mise à jour de la référence. Nos travaux actuels, non encore publiés, cherchent à générer des systèmes de mise à jour modulaires, individuels, et adaptables au cours du temps, plutôt que des systèmes figés et globaux pour tous les utilisateurs.

5.1.2. Adaptation aux réseaux de neurones profonds

Depuis nos essais sur l'utilisation de réseaux siamois en 2019 (voir la section 3.3.2), d'autres travaux ont été effectués par la communauté. Nous pouvons citer les travaux de Lis et al. qui utilisent des réseaux siamois pour l'authentification par mot de passe partiel [9]. TypeNet [3] généralise l'approche sur de l'authentification par texte libre en utilisant une architecture de type Réseau de Neurones (RN) récurrent. Plus récemment, une approche à base de transformeur s'est révélée encore plus performante [76]. On peut donc s'attendre à de nouvelles avancées en Dynamique de Frappe au clavier (DDF) en utilisant les dernières nouveautés en Apprentissage Profond (AP) et en travaillant sur des méthodes de mise à jour de la référence spécifique. Une autre stratégie pour augmenter la quantité de données disponibles pour l'enrôlement est la génération de données artificielles (quelque chose de relativement standard en traitement d'image avec différentes techniques usuelles d'augmentation de données). De tels générateurs commencent à exister en DDF [53]. On peut aussi s'attendre à ce que des réseaux générateurs [34] servent à ce genre d'approches.

5.1.3. De l'acceptabilité de l'authentification biométrique

La biométrie est l'unique système d'authentification basé sur ce que l'on est. Bien que ses performances soient variables d'une modalité à l'autre, elles ne sont pas forcément corrélées avec l'acceptabilité.

Cette notion d'acceptabilité a d'ailleurs évolué ces dernières années. Certaines entreprises semblent décidées à ne plus faire de reconnaissance faciale [1]. Il y a également des discussions pour ne plus l'utiliser par la police aux USA [2], ou certaines villes [3]. Il y a donc plus de place pour les biométries de faible qualité, comme la DDF pour usage personnel. Il y a clairement un besoin d'explicabilité (voir la partie IV).

1. https://www.francetvinfo.fr/replay-radio/nouveau-monde/nouveau-monde-pourquoi-ibm-renonce-t-il-a-la-reconnaissance-faciale_3982317.html (à prendre avec des pincettes, il n'y a plus d'annonce officielle)
2. https://www.technologyreview.com/2020/06/26/1004500/a-new-us-bill-would-ban-the-police-use-of-facial-recognition/
3. https://edition.cnn.com/2020/09/09/tech/portland-facial-recognition-ban/index.html

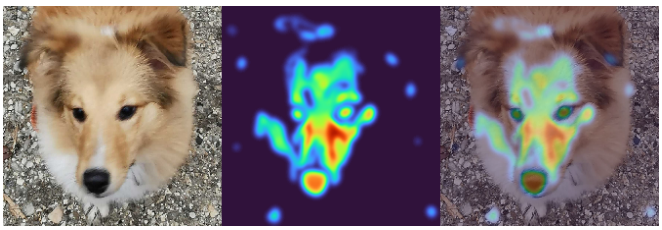


Figure 5.1. – Illustration de la haute résolution obtenue avec la méthode H2O [62] que nous sommes en train d'améliorer.

5.2. Attribution de caractéristiques pour les réseaux de neurones profonds

L'attribution de caractéristiques est la méthodologie la plus employée dans la littérature pour expliquer les réseaux de neurones profonds. Cependant, il n'existe pas de protocole d'évaluation standard, et il a été aussi montré que certaines implémentations ne sont pas dépendantes du modèle ou des données [6] ou que les explications encodent des motifs spécifiques au réseau [82] : il est donc difficile de leur faire confiance malgré le fait qu'elles soient faciles à lire. Il reste encore de nombreuses avancées à effectuer afin d'améliorer leur robustesse et leur évaluation.

5.2.1. De la faible résolution des explications

Nous avons présenté deux approches d'extraction de caractéristiques d'entrée pertinentes pour la classification d'images (voir les sections 4.2.1 et 4.2.2). Elles fonctionnent en mode boite blanche car elles ont besoin d'accéder aux activations à différents endroits de l'architecture. Nous avons comparé ces approches à GradCam, une méthode de l'état de l'art couramment utilisée [86] avec différentes techniques d'évaluation. Certaines fonctionnent avec une référence (une carte de saillance visuelle acquise à l'aide d'une expérience utilisateur), d'autres non. La comparaison montre que nos approches sont plus proches de ce qu'attend un humain par rapport à ce qui est produit par GradCam. FEM et MLFEM sont des approches basées sur l'utilisation des activations ; la résolution de leur sortie est donc limitée à la taille de ces tenseurs. Nous pensons que la perte d'information n'est pas négligeable, et que des approches basées sur les occlusions n'en souffrent pas. En contrepartie, ces approches basées sur l'occlusion sont systématiquement calculatoirement plus coûteuses et la résolution de l'explication dépend de la technique d'occlusion. Nous travaillons actuellement sur des approches boite noire hiérarchiques d'occlusion pour augmenter la résolution de l'explication, tout en diminuant le temps de calcul, en ne se focalisant que sur les parties de l'image les plus pertinentes [62] (voir la figure 5.1 pour un aperçu de la résolution de l'explication).

5.2.2. Évaluation des méthodes d'explication

Il faut noter que l'évaluation de telles approches nécessite d'être standardisée. En effet, nous pouvons trouver d'autres techniques d'évaluation dans la littérature [33, 59, 72] qui sont toutes aussi pertinentes. Il faut maintenant se poser la question de comment évaluer convenablement de telles approches en identifiant les méthodes d'évaluation les plus appropriées. Il a aussi été montré à plusieurs reprises que de telles approches d'explicabilité ne sont pas toujours robustes et pertinentes [5, 56] et peuvent même être contrefaites [24]. Nous n'avons pas encore effectué de telles analyses sur nos approches, mais il est important de le faire dans la suite. Je souhaite que cet aspect évaluation fasse partie de mes prochains travaux d'encadrement de thèse.

Une carte de chaleur, superposée ou non à l'image à expliquer, est souvent utilisée pour représenter ce type d'explication. Elle sert plus rarement à effectuer un fondu vers noir sur l'image à expliquer en occultant les pixels non intéressants. Nous pensons que cette deuxième approche est plus lisible que la première, mais il n'existe pas d'évaluation dans la littérature pour vérifier cet aspect. La façon de représenter convenablement ces explications nécessite cependant d'être investiguée dans des travaux futurs.

Dans tous les cas, l'utilisateur se contente de regarder l'image statique générée pour identifier les zones d'intérêt dans la donnée source. Le chapitre 13 présente d'autres types de méthodes d'explications reposant sur de l'Analyse Visuelle et Interactive (AVI).

5.3. Travaux non abordés

5.3.1. Détection de fraudes dans les systèmes de paiement mobile

Lors de mon postdoctorat chez Orange Labs, j'ai travaillé en collaboration avec des collègues de Fraunhofer Darmstadt/Allemagne sur la détection de fraude dans des systèmes de paiement mobile en recherchant des motifs dans des graphes de transaction [65]. Ils travaillaient sur un moteur générique de recherche de motifs dans un graphe [66], tandis que je travaillais sur des règles spécifiques à notre cas d'utilisation.

5.3.2. Détection de panne dans les systèmes de vélos en libre-service

Lors de l'encadrement de la thèse de Rémi Delassus, nous avons travaillé sur des techniques de détection de pannes dans les systèmes de vélo en libre-service [22]. Les quantités de données à manipuler pour ces travaux étaient relativement importantes et ont nécessité d'utiliser des approches de type traitement de masses de données (en utilisant une plateforme Hadoop).

5.3.3. Segmentation d'images satellites

Nous avons également travaillé sur des techniques de segmentation d'image satellite afin de détecter des bâtiments [21], en utilisant des approches d'AP multimodales dans le cadre du concours DeepGlobe 2018 [23].

5.3.4. Classification hiérarchique et multimodale

Lors de l'encadrement de la thèse de Korlan Rysbayeva, nous avons travaillé sur des techniques de classification hiérarchique et multimodale d'images [71]. Les contributions sont liées à la définition de telles architectures en prenant à la fois en compte l'aspect multimodalité (en fusionnants différents sous-réseaux profonds) et hiérarchiques (en mettant au point des fonctions de coût dédiées). Nous avons également utilisé des approches de type réseau siamois (toujours sous les mêmes contraintes) afin de pouvoir calculer des distances entre les images. Ces distances seront utiles lors de la création d'applications métiers liées à l'utilisation du jeu de données de cette thèse. Une technique d'explication par l'exemple a été développée en générant des graphes de proximité d'images à partir des descripteurs calculés par les réseaux.

De façon générale, des techniques de fusion d'information ont été utilisées dans beaucoup de mes travaux, et je pense que je chercherai toujours à en employer lorsque le contexte le permet.

Chapitre 6.

Bibliographie de l'analyse de données et l'apprentissage automatique

[1] M. El-Abed, R. **Giot**, B. Hemery et C. Rosenberger. « A study of users' acceptance and satisfaction of biometric systems ». Dans : *44th IEEE International Carnahan Conference on Security Technology (ICCST'10)*. 2010, p. 1-10 (cf. p. 18).

[2] S. Abnar et W. Zuidema. « Quantifying Attention Flow in Transformers ». Dans : *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020 (cf. p. 24).

[3] A. Acien, A. Morales, R. Vera-Rodriguez, J. Fierrez et J. V. Monaco. « Typenet: Scaling up keystroke biometrics ». Dans : *2020 IEEE International Joint Conference on Biometrics (IJCB)*. IEEE. 2020, p. 1-7 (cf. p. 31).

[4] A. Adadi et M. Berrada. « Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI) ». Dans : *IEEE Access* 6 (2018), p. 52138-52160 (cf. p. 25).

[5] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt et B. Kim. « Sanity Checks for Saliency Maps ». Dans : *Proceedings of the 32nd International Conference on Neural Information Processing Systems*. NIPS'18. Curran Associates Inc., 2018, p. 9525-9536 (cf. p. 32).

[6] J. Adebayo, J. Gilmer, M. Muelly, I. J. Goodfellow, M. Hardt et B. Kim. « Sanity Checks for Saliency Maps ». Dans : (2018), p. 9525-9536 (cf. p. 32).

[7] K. Ahmed Asif Fuad, P.-E. Martin, R. **Giot**, R. Bourqui, J. Benois-Pineau et A. Zemmari. « Features Understanding in 3D CNNs for Actions Recognition in Video ». Dans : *2020 Tenth International Conference on Image Processing Theory, Tools and Applications (IPTA)*. 2020, p. 6 (cf. p. 27).

[8] Z. Akhtar, C. Micheloni et G. L. Foresti. « Biometric liveness detection: Challenges and research opportunities ». Dans : *IEEE Security & Privacy* 13.5 (2015), p. 63-72 (cf. p. 18).

[9] D. Aspinall et M. Just. « "Give me letters 2, 3 and 6!": Partial password implementations and attacks ». Dans : *Financial Cryptography and Data Security: 17th International Conference, FC 2013, Okinawa, Japan, April 1-5, 2013, Revised Selected Papers 17*. Springer. 2013, p. 126-143 (cf. p. 31).

[10] M. G. Augasta et T. Kathirvalavakumar. « Rule extraction from neural networks - A comparative study ». Dans : *International Conference on Pattern Recognition, Informatics and Medical Engineering (PRIME-2012)*. IEEE, 2012 (cf. p. 25).

[11] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller et W. Samek. « On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation ». Dans : *PLOS ONE* 10.7 (2015). Sous la dir. d'O. D. Suarez, e0130140 (cf. p. 24).

[12] D. Bau, S. Liu, T. Wang, J.-Y. Zhu et A. Torralba. « Rewriting a Deep Generative Model ». Dans : *Computer Vision – ECCV 2020*. Springer International Publishing, 2020, p. 351-369 (cf. p. 25).

[13] D. Bau, B. Zhou, A. Khosla, A. Oliva et A. Torralba. « Network Dissection: Quantifying Interpretability of Deep Visual Representations ». Dans : *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017 (cf. p. 25).

[14] V. Belle et I. Papantonis. « Principles and Practice of Explainable Machine Learning ». Dans : *Frontiers in Big Data* 4 (2021) (cf. p. 25).

[15] L. Bourroux, R. Bourqui, J. Benois-Pineau et R. **Giot**. « Multi Layered Feature Explanation Method for Convolutional Neural Networks ». Dans : *3rd International Conference on Pattern Recognition and Artificial Intelligence (ICPRAI 2022)*. 2022, p. 12 (cf. p. 27, 28).

[16] P. Bours et H. Barghouthi. « Continuous authentication using biometric keystroke dynamics ». Dans : *The Norwegian Information Security Conference (NISK)*. T. 2009. 2009 (cf. p. 18).

[17] P. Bours et E. Masoudian. « Applying keystroke dynamics on one-time pin codes ». Dans : *2nd International Workshop on Biometrics and Forensics*. IEEE. 2014, p. 1-6 (cf. p. 18).

[18] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger et R. Shah. « Signature verification using a" siamese" time delay neural network ». Dans : *Advances in neural information processing systems* 6 (1993) (cf. p. 16, 20, 21).

[19] P. Campisi, E. Maiorana, M. L. Bosco et A. Neri. « User authentication using keystroke dynamics for cellular phones ». Dans : *IET Signal Processing* 3.4 (2009), p. 333-341 (cf. p. 18).

[20] M. Craven et J. Shavlik. « Extracting Tree-Structured Representations of Trained Networks ». Dans : *Advances in Neural Information Processing Systems*. Sous la dir. de D. Touretzky, M. C. Mozer et M. Hasselmo. T. 8. MIT Press, 1996 (cf. p. 25).

[21] R. Delassus et R. **Giot**. « CNNs Fusion for Building Detection in Aerial Images for the Building Detection Challenge ». Dans : *Proceedings of CVPR Workshop DeepGlobe: A Challenge for Parsing the Earth through Satellite Images (CVPRW)*. 2018 (cf. p. 13, 33).

[22] R. Delassus, R. **Giot**, R. Cherrier, G. Barbieri et G. Melançon. « Broken Bikes Detection Using CitiBike Bikeshare System Open Data ». Dans : *IEEE Symposium Series in Computational Intelligence 2016 (SSCI 2016). Workshop on Computational Intelligence in Vehicles and Transportation Systems (CIVTS 2016)*. 2016, p. 6 (cf. p. 13, 33).

[23] I. Demir, K. Koperski, D. Lindenbaum, G. Pang, J. Huang, S. Basu, F. Hughes, D. Tuia et R. Raskar. « Deepglobe 2018: A challenge to parse the earth through satellite images ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2018, p. 172-181 (cf. p. 33).

[24] A.-K. Dombrowski, M. Alber, C. J. Anders, M. Ackermann, K.-R. Müller et P. Kessel. « Explanations Can Be Manipulated and Geometry is to Blame ». Dans : *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Curran Associates Inc., 2019 (cf. p. 32).

[25] R. C. Fong et A. Vedaldi. « Interpretable Explanations of Black Boxes by Meaningful Perturbation ». Dans : *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2017 (cf. p. 23).

[26] K. A. A. Fuad, P.-E. Martin, R. Giot, R. Bourqui, J. Benois-Pineau et A. Zemmari. « Features Understanding in 3D CNNs for Actions Recognition in Video ». Dans : *2020 Tenth International Conference on Image Processing Theory, Tools and Applications (IPTA)*. IEEE, 2020 (cf. p. 24, 26).

[27] R. **Giot**, M. El-Abed, B. Hemery et C. Rosenberger. « Unconstrained Keystroke Dynamics Authentication with Shared Secret ». Dans : *ELSEVIER International journal on Computers & Security (COSE)* 30.6-7 (2011), p. 427-445 (cf. p. 18, 20).

[28] R. **Giot**, M. El-Abed et C. Rosenberger. « Keystroke Dynamics Overview ». Dans : *Biometrics / Book 1*. Sous la dir. de D. J. Yang. T. 1. InTech, 2011. Chap. 8, p. 157-182 (cf. p. 16, 18).

[29] R. **Giot**, B. Dorizzi et C. Rosenberger. « A Review on the Public Benchmark Databases for Static Keystroke Dynamics ». Dans : *ELSEVIER International journal on Computer & Security (COSE)* 55 (2015), p. 46-61 (cf. p. 18).

[30] R. **Giot** et A. Rocha. « Siamese Networks for Static Keystroke Dynamics Authentication ». Dans : *Information Forensics and Security (WIFS), IEEE International Workshop on*. 2019, p. 6 (cf. p. 21).

[31] R. **Giot**, C. Rosenberger et B. Dorizzi. « Performance Evaluation of Biometric Template Update ». Dans : *International Biometric Performance Testing Conference (IBPC 2012)*. 2012, p. 1-4 (cf. p. 19).

[32] R. Girshick, J. Donahue, T. Darrell et J. Malik. « Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation ». Dans : *2014 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2014 (cf. p. 25).

[33] T. Gomez, T. Fréour et H. Mouchère. « Metrics for saliency map evaluation of deep learning explanation methods ». Dans : *International Conference on Pattern Recognition and Artificial Intelligence*. Springer. 2022, p. 84-95 (cf. p. 32).

[34] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville et Y. Bengio. « Generative adversarial networks ». Dans : *Communications of the ACM* 63.11 (2020), p. 139-144 (cf. p. 31).

[35] S. Greydanus, A. Koul, J. Dodge et A. Fern. « Visualizing and Understanding Atari Agents ». Dans : *Proceedings of the 35th International Conference on Machine Learning*. Sous la dir. de J. Dy et A. Krause. T. 80. Proceedings of Machine Learning Research. PMLR, 2018, p. 1792-1801 (cf. p. 23).

[36] A. K. Jain, P. Flynn et A. A. Ross. *Handbook of biometrics*. Springer Science & Business Media, 2007 (cf. p. 16, 17).

[37] Z. Jin, Y. Wang, Q. Wang, Y. Ming, T. Ma et H. Qu. « Gnnlens: A visual analytics approach for prediction error diagnosis of graph neural networks ». Dans : *IEEE Transactions on Visualization and Computer Graphics* (2022) (cf. p. 24).

[38] E. M. Kenny, C. Ford, M. Quinn et M. T. Keane. « Explaining black-box classifiers using post-hoc explanations-by-example: The effect of explanations and error-rates in XAI user studies ». Dans : *Artificial Intelligence* 294 (2021), p. 103459 (cf. p. 24).

[39] E. M. Kenny et M. T. Keane. « On generating plausible counterfactual and semi-factual explanations for deep learning ». Dans : *Proceedings of the AAAI Conference on Artificial Intelligence*. T. 35. 13. 2021, p. 11575-11585 (cf. p. 24).

[40] K. S. Killourhy et R. A. Maxion. « Comparing anomaly-detection algorithms for keystroke dynamics ». Dans : *2009 IEEE/IFIP International Conference on Dependable Systems & Networks*. IEEE. 2009, p. 125-134 (cf. p. 20, 21).

[41] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas et R. sayres. « Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV) ». Dans : *Proceedings of the 35th International Conference on Machine Learning*. Sous la dir. de J. Dy et A. Krause. T. 80. Proceedings of Machine Learning Research. PMLR, 2018, p. 2668-2677 (cf. p. 25).

[42] B. La Rosa, G. Blasili, R. Bourqui, D. Auber, G. Santucci, R. Capobianco, E. Bertini, R. **Giot** et M. Angelini. « State of the Art of Visual Analytics for eXplainable Deep Learning ». Dans : *Computer Graphics forum (CGF)* 42.1 (2023), p. 319-355 (cf. p. 24).

[43] T. Laugel, M.-J. Lesot, C. Marsala, X. Renard et M. Detyniecki. « The Dangers of Post-hoc Interpretability: Unjustified Counterfactual Explanations ». Dans : *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, 2019 (cf. p. 25).

[44] Y. LeCun, Y. Bengio et G. Hinton. « Deep learning ». Dans : *nature* 521.7553 (2015), p. 436-444 (cf. p. 16).

[45] Z. C. Lipton. « The Mythos of Model Interpretability: In Machine Learning, the Concept of Interpretability is Both Important and Slippery. » Dans : *Queue* 16.3 (2018), p. 31-57. issn : 1542-7730 (cf. p. 25).

[46] K. Lis, E. Niewiadomska-Szynkiewicz et K. Dziewulska. « Siamese Neural Network for Keystroke Dynamics-Based Authentication on Partial Passwords ». Dans : *Sensors* 23.15 (2023), p. 6685 (cf. p. 31).

[47] M. Liu, S. Liu, H. Su, K. Cao et J. Zhu. « Analyzing the Noise Robustness of Deep Neural Networks ». Dans : *2018 IEEE Conference on Visual Analytics Science and Technology (VAST)*. IEEE, 2018 (cf. p. 25).

[48] S. M. Lundberg et S.-I. Lee. « A Unified Approach to Interpreting Model Predictions ». Dans : *Advances in Neural Information Processing Systems*. Sous la dir. d'I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan et R. Garnett. T. 30. Curran Associates, Inc., 2017 (cf. p. 23).

[49] A. Mahendran et A. Vedaldi. « Understanding Deep Image Representations by Inverting Them ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015 (cf. p. 25).

[50] A. Mahendran et A. Vedaldi. « Visualizing Deep Convolutional Neural Networks Using Natural Pre-images ». Dans : *International Journal of Computer Vision* 120.3 (2016), p. 233-255 (cf. p. 25).

[51] D. Martens et F. Provost. « Explaining Data-Driven Document Classifications ». Dans : *MIS Quarterly* 38.1 (2014), p. 73-99 (cf. p. 24).

[52] P.-E. Martin, J. Benois-Pineau, R. Péteri et J. Morlier. « Fine grained sport action recognition with twin spatio-temporal convolutional neural networks ». Dans : *Multimedia Tools and Applications* 79.27 (2020), p. 20429-20447 (cf. p. 27).

[53] D. Migdal et C. Rosenberger. « Statistical modeling of keystroke dynamics samples for the generation of synthetic datasets ». Dans : *Future Generation Computer Systems* 100 (2019), p. 907-920. issn : 0167-739X (cf. p. 31).

[54] G. Montavon, A. Binder, S. Lapuschkin, W. Samek et K.-R. Müller. « Layer-Wise Relevance Propagation: An Overview ». Dans : *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*. Springer International Publishing, 2019, p. 193-209 (cf. p. 24, 25).

[55] A. Nguyen, A. Dosovitskiy, J. Yosinski, T. Brox et J. Clune. « Synthesizing the preferred inputs for neurons in neural networks via deep generator networks ». Dans : *Advances in Neural Information Processing Systems*. Sous la dir. de D. Lee, M. Sugiyama, U. Luxburg, I. Guyon et R. Garnett. T. 29. Curran Associates, Inc., 2016 (cf. p. 25).

[56] G. Nguyen, D. Kim et A. Nguyen. « The effectiveness of feature attribution methods and its correlation with automatic evaluation scores ». Dans : *Advances in Neural Information Processing Systems* 34 (2021), p. 26422-26436 (cf. p. 32).

[57] C. Olah, A. Mordvintsev et L. Schubert. « Feature visualization ». Dans : *Distill* 2.11 (2017), e7 (cf. p. 24).

[58] C. Olah, A. Mordvintsev et L. Schubert. « Feature Visualization ». Dans : *Distill* 2.11 (2017) (cf. p. 25).

[59] V. Petsiuk, A. Das et K. Saenko. « Rise: Randomized input sampling for explanation of black-box models ». Dans : *arXiv preprint arXiv:1806.07421* (2018) (cf. p. 32).

[60] P. H. Pisani, R. **Giot**, A. C. P. L. F. de Carvalho et A. C. Lorena. « Enhanced Template Update: Application to Keystroke Dynamics ». Dans : *ELSEVIER International journal on Computer & Security (COSE)* 60 (2016), p. 134-153 (cf. p. 19, 20).

[61] P. H. Pisani, A. Mhenni, R. **Giot**, E. Cherrier, N. Poh, A. C. P. L. F. de Carvalho, C. Rosenberger et N. E. Ben Amara. « Adaptive Biometric Systems: Review and Perspectives ». Dans : *ACM Computing Surveys (CSUR)* 52.5 (2019) (cf. p. 18, 19, 31).

[62] L.-E. Pommé, R. **Giot** et R. Bourqui. « H^2O: Heatmaps by Hierarchical Occlusion ». Dans : *Proceedings of the 20th International Conference on Content-based Multimedia Indexing (CBMI), special session on Explainability in Multimedia Analysis (ExMA).* 2023 (cf. p. 24, 32).

[63] R. Poyiadzi, K. Sokol, R. Santos-Rodriguez, T. D. Bie et P. Flach. « FACE ». Dans : *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society.* ACM, 2020 (cf. p. 25).

[64] M. T. Ribeiro, S. Singh et C. Guestrin. « "Why Should I Trust You?" » Dans : *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* ACM, 2016 (cf. p. 23).

[65] R. Rieke, R. **Giot** et C. Gaber. « Predictive Security Analysis - Concepts, Implementation, first Results in Industrial Scenario ». Dans : *CYBER SECURITY & PRIVACY EU FORUM 2013.* CSP FORUM. 2013 (cf. p. 32).

[66] R. Rieke et Z. Stoynova. « Predictive security analysis for event-driven processes ». Dans : *Computer Network Security: 5th International Conference on Mathematical Methods, Models and Architectures for Computer Network Security, MMM-ACNS 2010, St. Petersburg, Russia, September 8-10, 2010. Proceedings 5.* Springer. 2010, p. 321-328 (cf. p. 32).

[67] R. Rieke, M. Zhdanova, J. Repp, R. **Giot** et C. Gaber. « Fraud Detection in Mobile Payments Utilizing Process Behavior Analysis ». Dans : *The 2nd International Workshop on Recent Advances in Security Information and Event Management (RaSIEM 2013).* 2013 (cf. p. 13).

[68] B. L. Rosa, R. Capobianco et D. Nardi. « Explainable Inference on Sequential Data via Memory-Tracking ». Dans : *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence.* International Joint Conferences on Artificial Intelligence Organization, 2020 (cf. p. 24).

[69] A. A. Ross, K. Nandakumar et A. K. Jain. *Handbook of multibiometrics.* T. 6. Springer Science & Business Media, 2006 (cf. p. 18).

[70] J. Roth, X. Liu, A. Ross et D. Metaxas. « Biometric authentication via keystroke sound ». Dans : *2013 international conference on biometrics (ICB).* IEEE. 2013, p. 1-8 (cf. p. 18).

[71] K. Rysbayeva, R. **Giot** et N. Journet. « Hierarchical and Multimodal Classification of Images from Soil Remediation Reports ». Dans : *Proceedings of the 16th International Conference on Document Analysis and Recognition (ICDAR 2021).* 2021, p. 160-175 (cf. p. 13, 33).

[72] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh et D. Batra. « Grad-cam: Visual explanations from deep networks via gradient-based localization ». Dans : *Proceedings of the IEEE international conference on computer vision.* 2017, p. 618-626 (cf. p. 25, 27, 32).

[73] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh et D. Batra. « Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization ». Dans : *2017 IEEE International Conference on Computer Vision (ICCV).* IEEE, 2017 (cf. p. 23).

[74] K. Simonyan, A. Vedaldi et A. Zisserman. « Deep inside convolutional networks: Visualising image classification models and saliency maps ». Dans : *In Workshop at International Conference on Learning Representations.* Citeseer. 2014 (cf. p. 27).

[75] J. T. Springenberg, A. Dosovitskiy, T. Brox et M. Riedmiller. « Striving for simplicity: The all convolutional net ». Dans : *arXiv preprint arXiv:1412.6806* (2014) (cf. p. 27).

[76] G. Stragapede, P. Delgado-Santos, R. Tolosana, R. Vera-Rodriguez, R. Guest et A. Morales. « Mobile Keystroke Biometrics Using Transformers ». Dans : *2023 IEEE 17th International Conference on Automatic Face and Gesture Recognition (FG).* IEEE Computer Society, 2023, p. 1-6 (cf. p. 31).

[77] H. Strobelt, S. Gehrmann, H. Pfister et A. M. Rush. « LSTMVis: A Tool for Visual Analysis of Hidden State Dynamics in Recurrent Neural Networks ». Dans : *IEEE Transactions on Visualization and Computer Graphics* 24.1 (2018), p. 667-676 (cf. p. 24).

[78] M. Sundararajan, A. Taly et Q. Yan. « Axiomatic Attribution for Deep Networks ». Dans : *Proceedings of the 34th International Conference on Machine Learning.* Sous la dir. de D. Precup et Y. W. Teh. T. 70. Proceedings of Machine Learning Research. PMLR, 2017, p. 3319-3328 (cf. p. 23).

[79] K.-W. Tse et K. Hung. « Behavioral biometrics scheme with keystroke and swipe dynamics for user authentication on mobile platform ». Dans : *2019 IEEE 9th Symposium on Computer Applications & Industrial Electronics (ISCAIE).* IEEE. 2019, p. 125-130 (cf. p. 18).

[80] S. Venkatasubramanian et M. Alfano. « The philosophical basis of algorithmic recourse ». Dans : *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency.* ACM, 2020 (cf. p. 25).

[81] J. Wang, L. Gou, H. Shen et H. Yang. « DQNViz: A Visual Analytics Approach to Understand Deep Q-Networks ». Dans : *IEEE Transactions on Visualization and Computer Graphics* 25.1 (2019), p. 288-298. issn : 1941-0506 (cf. p. 25).

[82] A. Widdicombe, S. Julier et B. Kim. « Saliency Maps Contain Network" Fingerprints" ». Dans : *ICLR 2022 Workshop on PAIR^{2}Struct: Privacy, Accountability, Interpretability, Robustness, Reasoning on Structured Data.* 2022 (cf. p. 32).

[83] M. D. Zeiler et R. Fergus. « Visualizing and Understanding Convolutional Networks ». Dans : *Computer Vision – ECCV 2014.* Springer International Publishing, 2014, p. 818-833 (cf. p. 24, 25).

[84] M. D. Zeiler, D. Krishnan, G. W. Taylor et R. Fergus. « Deconvolutional networks ». Dans : *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition.* IEEE, 2010 (cf. p. 24).

[85] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva et A. Torralba. « Learning Deep Features for Discriminative Localization ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2016 (cf. p. 24).

[86] A. Zhukov, J. Benois-Pineau et R. **Giot**. « Evaluation of Explanation Methods of AI - CNNs in Image Classification Tasks with Reference-based and No-reference Metrics ». Dans : *Advances in Artificial Intelligence and Machine Learning (OAJAIML)* 3 (2023), p. 620-646 (cf. p. 32).

Troisième partie

Visualisation de masses de données

Chapitre 7. Introduction à la visualisation de masses de données 43
Chapitre 8. Dessin de grands graphes 47
8.1. Introduction au dessin de grands graphes 47
8.1.1. Définitions utiles pour la suite 47
8.1.2. Dessin de graphes 48
8.1.3. Faisceautage d'arêtes 49
8.2. Utilisation de la dégénérescence pour le dessin multiniveaux de grands graphes 50
8.2.1. Présentation de HCBL 50
8.2.2. Résultats de HCBL 52
8.3. Faisceautage multi-échelle d'arêtes moins bruité 53
8.3.1. Présentation de la méthode de faisceautage hiérarchique 53
8.3.2. Evaluation de la méthode de faisceautage hiérarchique 54
Chapitre 9. Représentation de masses de données tabulaires dans une grille 55
9.1. Introduction à la représentation de masses de données tabulaires 55
9.1.1. Contexte 55
9.1.2. Visualisation de données en espace contraint 55
9.2. Projection de données dans une grille : Voronoi Relaxation Grid 57
9.2.1. Description de VRGRid 57
9.2.2. Évaluation de VRGRid 59
Chapitre 10. Discussion sur la visualisation de masses de données 61
10.1. Dessin de grands graphes 61
10.1.1. Plongement des nœuds 61
10.1.2. Faisceautage des arêtes 61
10.2. Représentation de masses de données tabulaire dans une grille 62
10.3. Travaux non abordés 62
Chapitre 11. Bibliographie de la visualisation de masses de données 63

Chapitre 7.

Introduction à la visualisation de masses de données

Contexte

Mon arrivée au LaBRI comme maître de conférences m'a permis de changer de thématique de recherche pour m'orienter vers la Visualisation d'Information (VI) pour les masses de données. J'ai pu diversifier mes thématiques de recherche académique en abordant des aspects en dehors de ma zone de confort, et de participer à des enseignements liés à l'analyse et la visualisation de données à l'ENSEIRB-MATMECA. Ce chapitre illustre quelques travaux sur ce domaine, tout d'abord sur le dessin de grands graphes avec une méthode de dessin de graphes multiéchelle, et une méthode de faisceautage d'arêtes sur des graphes ayant des nœuds classés hiérarchiquement, puis sur une méthode de dessin de données orientées pixels.

Le graphe ego-centré de mes co-auteurs sur cette partie est présenté en figure 7.1. La taille des nœuds et des arêtes est proportionnelle au nombre de publications communes avec les collègues concernés. La couleur dépend d'un algorithme de partitionnement de graphe, mais reflète globalement bien les origines des collaborations.

Les techniques et méthodes pour la VI aident leurs utilisateurs à comprendre les données visualisées. Cette compréhension n'ait pas possible en analysant directement les données brutes ou des statistiques descriptives. D'ailleurs, comment représenter à l'utilisateur ces données brutes si ce n'est en les visualisant ? Il est également connu que les mêmes statistiques peuvent être obtenues sur des données totalement différentes, et donc en induire une mauvaise interprétation (voir la figure 7.2 page suivante). Le dicton « Une image valant 1000 mots » sous-entend qu'il est souvent plus aisé de comprendre un message à partir d'une représentation visuelle, plutôt qu'une description textuelle. C'est d'autant plus vrai pour l'analyse de données lorsqu'on se rend compte que les descriptions statistiques peuvent faire croire à ce que des jeux de données différents soient similaires alors que leur représentation visuelle montre clairement que ça n'est pas le cas.

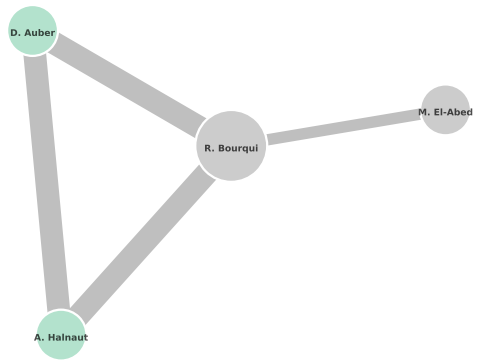
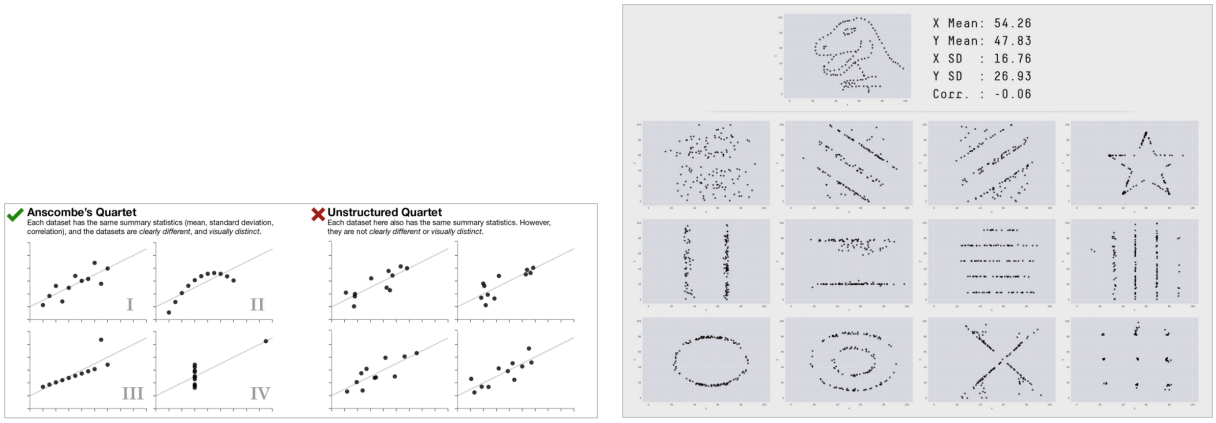


Figure 7.1. – Graphe égo centré de mes co-auteurs sur les travaux liés à la Visualisation d'Information (VI).

Malheureusement, dès que la quantité de données devient importante, les limites de perception humaines [54] empêchent de les appréhender, et les limites techniques ou algorithmiques rendent impossible de les traiter en temps raisonnable dans un contexte d'Analyse Visuelle et Interactive (AVI). De façon générale, la visualisation de masse de données impose d'utiliser des techniques et méthodes différentes pour pallier ces deux problèmes. Le mantra de Shneiderman [59] : « Overview first, zoom and filter, then details-on-demand » présente un des paradigmes couramment utilisés pour représenter des masses de données et résoudre les limites de perceptions humaines principalement. Schématiquement, plutôt que de chercher à visualiser toutes les données, une représentation approximative est effectuée pour identifier des zones d'intérêt, puis permettre à l'utilisateur de réclamer l'obtention de plus de détails sur les données associées à cette zone.

Classiquement, les limites de traitement sont souvent résolues en utilisant des approches hiérarchiques de traitement et visualisation de données. Le principe étant de partitionner les données pour traiter plusieurs sous-ensembles disjoints plus petits afin de moins souffrir de l'utilisation d'algorithmes quadratiques en la taille des données. Ces sous-partitions peuvent



(a) Illustration de jeux de données différents ont les mêmes résumés statistiques. L'image de gauche représente le quartet d'Anscombe [4] pour lequel les données sont clairement différentes lorsque représentées avec un nuage de point et l'image de gauche représente des jeux différents mais issus de la même distribution.

(b) Nuages de points totalement différents avec les mêmes résumés statistiques [52].

Images extraites de https://www.autodesk.com/research/publications/same-stats-different-graphs

Figure 7.2. – Illustrations diverses de l'incapacité des résumés statistiques à clairement représenter les jeux de données contrairement à la Visualisation d'Information (VI) (à partir du moment où nous sommes capables de générer une représentation qui ne ment pas).

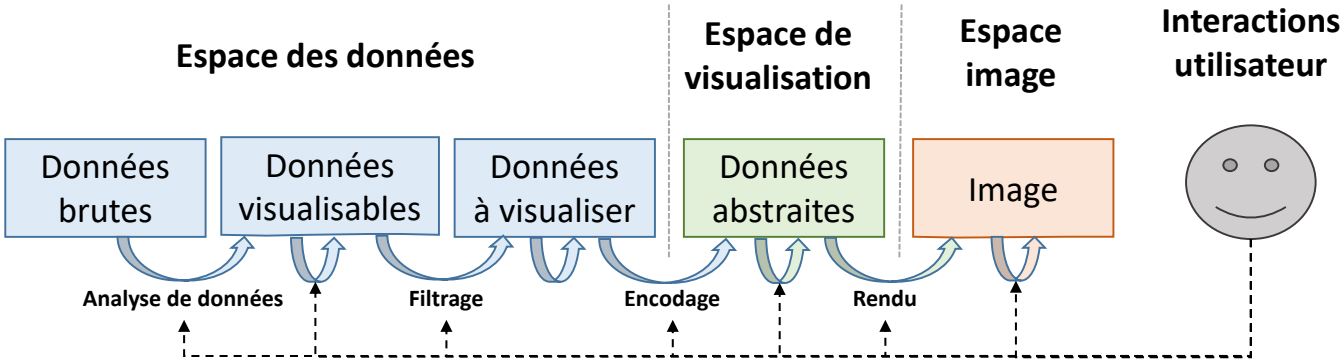


Figure 7.3. – Illustration d'une chaîne de traitements standard pour aller de la donnée brute, vers sa représentation [33].

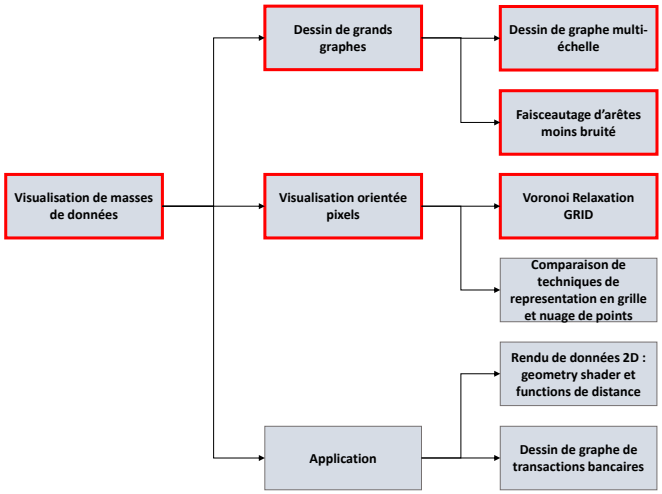


Figure 7.4. – Famille de travaux réalisés dans cette thématique. Seuls ceux encadrés en rouge sont présentés.

également être partitionnées récursivement pour réduire au maximum la taille des ensembles.

La figure 7.3 présente un modèle de traitement standard de VI [33]. La chaine part de la donnée brute qui doit être traitée et analysée afin d'extraire de l'information nécessaire à la résolution d'une tâche d'intérêt. Comme indiqué précédemment, il est souvent nécessaire d'effectuer une étape de sélection des données à l'aide de différentes techniques de filtrage. Les données sélectionnées doivent être encodées à travers différentes abstractions afin de pouvoir les rendre dans une image. Différentes possibilités d'interaction doivent être offertes à l'utilisateur afin de contrôler ces étapes.

Organisation

Les différents concepts détaillés dans cette partie sont illustrés en figure 7.4 page ci-contre.

Le chapitre 8 décrit des travaux liés à la visualisation de très grands graphes. La section 8.1 liste les différentes notations et travaux clés liés au dessin de grands graphes. La section section 8.2 présente notre contribution au dessin de grands graphes sur l'aspect positionnement des nœuds. La section 8.3 présente notre contribution au dessin de grands graphes sur l'aspect positionnement des points de contrôle des arêtes.

Le chapitre 9 décrit des travaux liés à la visualisation en grille. La section 9.1 introduit la thématique ainsi que les travaux clés liés à la projection de données en grille. La section 9.2 décrit notre méthode VRGrid.

Le chapitre 10 présente une synthèse des travaux liés à cette thématique, qu'ils aient été présentés ou non.

Chapitre 8.

Dessin de grands graphes

8.1. Introduction au dessin de grands graphes

Les graphes sont utilisés pour modéliser des problèmes qui nécessitent de représenter des relations entre différentes entités. Il est souvent nécessaire de les visualiser afin appréhender leur topologie. La littérature sur le dessin de graphes est foisonnante [10], tout comme les techniques de représentation de graphes. Dans ce chapitre, nous nous intéressons à la « vue nœuds-liens » qui dessine le graphe dans un espace 2D où chaque nœud est représenté par un glyphe (souvent un disque), et chaque arête par une poly ligne. Il existe des méthodes généralistes qui reposent souvent sur des calculs de forces d'attraction ou de répulsion (*p. ex.* FM3 [29]), ou des méthodes dédiées à des sous-classes de graphes particuliers (*p. ex.* graphes planaires [41], arbres [66]).

Cependant, dès que les graphes ont une taille conséquente, le temps de calcul devient prohibitif, et le positionnement des nœuds ne permet pas de représenter convenablement le graphe dans un espace-écran. Même les méthodes de calcul hiérarchique peuvent souffrir de lenteur ou de faible qualité de rendu (*c.-à-d.* l'effet pelote de poils, ou « hair-ball effect » dans la littérature anglophone, illustré dans la figure 8.1) en raison de méthodes de partitionnement sous-optimales.

Bien entendu, le dessin de grands graphes a été longuement étudié durant les dernières décennies. Nous pouvons noter deux approches majeures pour traiter le problème : les approches multiniveaux qui cherchent à simplifier le problème en manipulant des graphes simplifiés [29], et les approches algébriques [32] qui cherchent à simplifier le problème en faisant des calculs approchés. Dans nos travaux, nous avons suivi une approche multiniveaux.

La section 8.1.1 présente les notations nécessaires à la description de la section 8.1.2 sur le positionnement de nœuds de grands graphes et le chapitre section 8.1.3 sur le positionnement des arrêtes de grands graph.

8.1.1. Définitions utiles pour la suite

Soit un graphe simple $G = (V, E)$ avec V l'ensemble de ses nœuds et $E \subseteq V \times V$ l'ensemble de ses arêtes.

Un arbre libre $T = (V, E)$ correspond à un graphe sans cycle, tandis qu'un arbre enraciné est un arbre ayant un nœud qui correspond à sa racine et des arêtes dirigées depuis cette racine vers les feuilles. Un graphe partitionné $CG = (G, T)$ correspond à un graph G associé à un arbre enraciné T dont les feuilles représentent les nœuds de G. Chaque nœud interne v de T correspond à une partition des nœuds de G. Chaque niveau de T correspond à un niveau du partitionnement hiérarchique La profondeur du partitionnement hiérarchique correspond à la hauteur de l'arbre.

Un graphe quotient [12] $Q_G = (V_Q, E_Q)$ d'une partition $(C_1, \ldots, C_k)$ des nœuds de $G = (V, E)$ est défini comme :

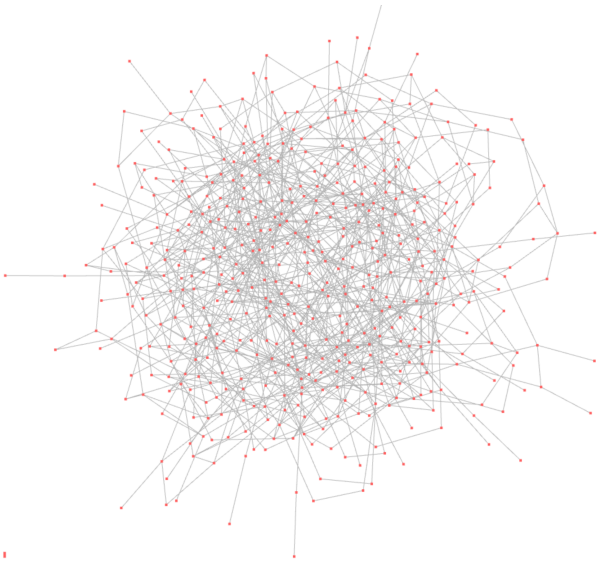


Figure 8.1. – Illustration de l'effet pelote de poils sur un graphe de taille raisonnable. L'effet est naturellement pire sur un plus gros graphe. On s'attend à l'éviter au maximum avec les approches multiniveaux.

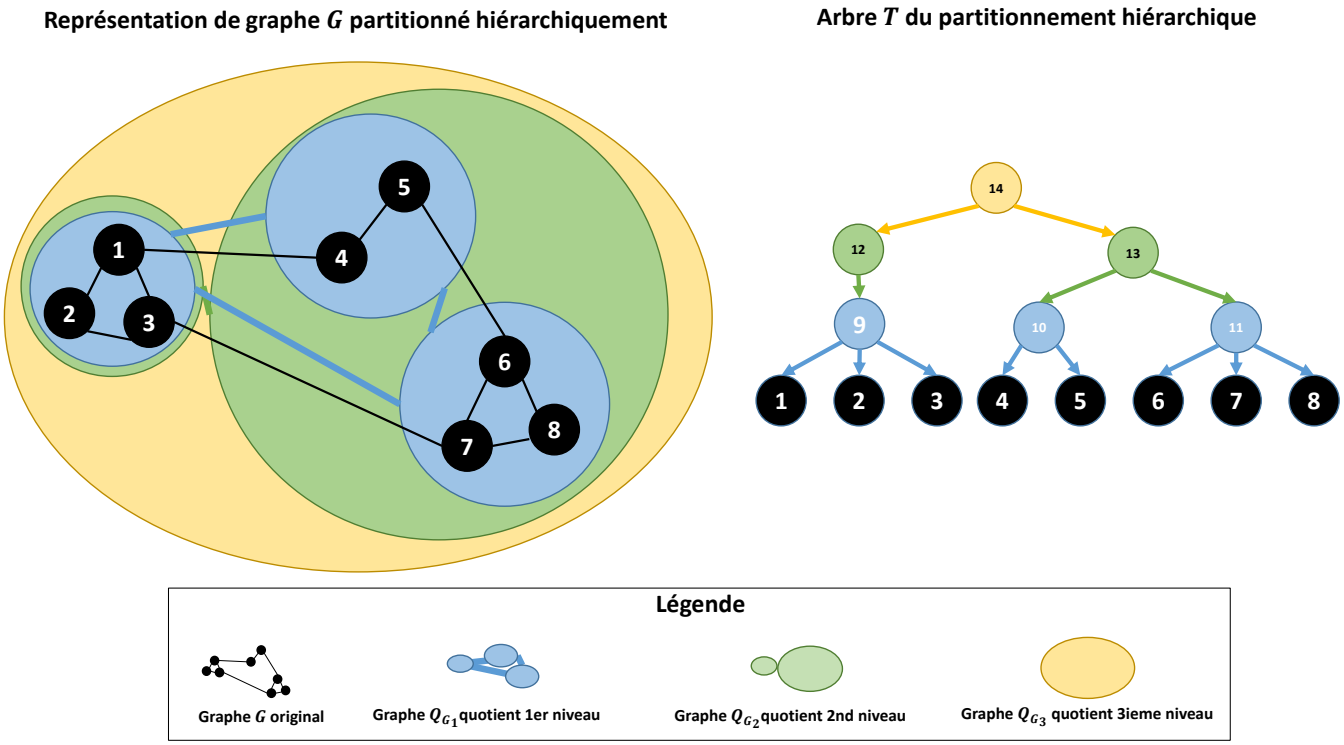


Figure 8.2. – Illustration du concept de partitionnement hiérarchique de graphe et de graphe quotient. Le graphe de 8 nœuds a été partitionné en 3 niveaux. Chaque niveau est encodé par un graphe quotient.

$V_Q = \{C_1, \ldots, C_k\}$ avec $(C_i, C_j) \in E_Q$ si et seulement si $i \neq j$ et $\exists u \in C_i$ et $\exists v \in C_j$ tel que $(u, v) \in E$.

Pour un graphe partitionné $CG = (G, T)$, à chaque niveau l de T correspond un graphe quotient $Q_G^l = (V_Q^l, E_Q^l)$ où V_Q^l est la partition représentée par les nœuds du niveau l de T. Un méta-nœud correspond à un nœud du graphe quotient et représente une partition C_i du graphe original et une métaarête lie deux métanœuds. La figure 8.2 illustre ces différents concepts.

Un sous-graphe H de G induit par un ensemble $C \subseteq V$ est le k-dégéré de G si et seulement si $\forall_v \in C, degree_H(v) >= k$, et H et le sous-graphe maximal ayant cette propriété [58]. Évidemment le $(k+1)$-dégénéré d'un graphe G est un sous-graphe du $k - dégénéré$ de ce graphe. La dégénérescence d'un nœud v de G correspond à c si il appartient au $c - dégénéré$ mais pas au $(c+1)$-dégénéré. Cette information est utilisée par la suite pour partitionner un graphe.

La matrice Laplacienne d'un graphe G non orienté est définie par $L_G = D_G - A_G$, avec D_G sa matrice des degrés et A_G sa matrice d'adjacence. Ainsi cette matrice est définie par :

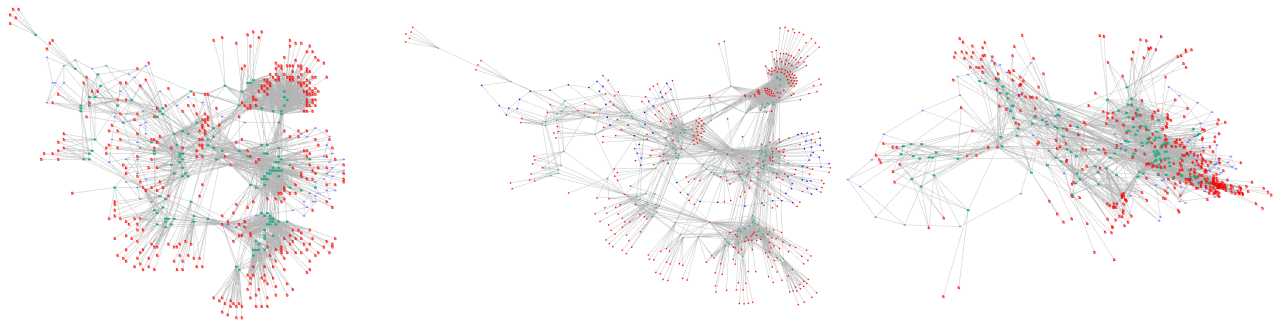
$$L_G(i,j) = \begin{cases} D(i,j) = degré(v_i) & \text{si } i = j \\ 0 - A_G(i,j) & \text{si } i \neq j \\ 0 & \text{sinon} \end{cases} \tag{8.1}$$

8.1.2. Dessin de graphes

Les algorithmes calculant des forces sont communs dans les approches multiniveaux. Ils contournent le problème de complexité calculatoire des méthodes standards de calculs de force [20, 23] en utilisant un partitionnement hiérarchique du graphe, et en faisant un compromis entre le temps de calcul et l'esthétisme du positionnement final.

FM^3 [29] est une des méthodes les plus rapides de la littérature qui propose des positionnements visuellement intéressants avec une complexité de $O(|V| log(|V|) + |E|)$. Le graphe est compressé récursivement en utilisant des règles simulant des galaxies jusqu'à obtenir un nombre fixé de nœuds. Ensuite, la méthode utilise un algorithme de force avec expansion multipôle pour dessiner chaque niveau du partitionnement hiérarchique.

GRIP [24] utilise une approche différente basée sur des distances entre nœuds (« Maximal Independent Set Filtration ») pour élire de façon itérative des représentants jusqu'à ce qu'il ne reste que quelques nœuds. Des méthodes standards de calcul de force sont utilisées pour dessiner les différents niveaux.



(a) FM^3 [29]. **(b)** Fruchterman [23]. **(c)** GRIP [24].

Figure 8.3. – Dessins du graphe représentant le dépôt git de cette HDR (les nœuds correspondent aux commits et fichiers) à l'aide de trois méthodes de dessin de graphe. On voit clairement que la lisibilité diffère en fonction des algorithmes, et qu'il y a toujours une marge de progression possible.

Cependant, bien que ces approches peuvent dessiner de gros graphes, le rendu souffre de l'effet « pelote de poils » [26] : une quantité importante de bruit due aux différents croisements d'arêtes (voir la figure 8.1 page 47). Pour des réseaux vraiment compliqués, d'autres méthodes basées sur la détection de communauté ou de structures topologiques particulières peuvent mieux convenir. Certains travaux utilisent une décomposition en composants biconnectés du graphe pour extraire une structure arborescente du réseau [5, 60]. TopoLayout [6] généralise cette approche et extrait récursivement les caractéristiques topologiques du graphe et utilise des algorithmes spécialisés sur ces structures pour les dessiner. Des méthodes similaires ont également été proposées en utilisant une décomposition hiérarchique du graphe associé à un algorithme de dessin hybridant force et «space filling» [16, 39].

La dégénérescence [58] est un élément important pour certaines des méthodes de détection de communautés, car elle présente des résultats intéressants tout en offrant une complexité en $O(|E|)$. Baur et al. proposent d'afficher le graph en 2,5D avec la couche inférieure contenant le 1-dégénéré, du graphe et la couche la plus haute le plus grand dégénéré [11]. Le sous-graphe avec la plus grande dégénérescence est dessiné avec un algorithme de dessin spectral et les graphes de dégénérescence plus petite sont progressivement insérés dans le dessin en utilisant un algorithme de force. Une autre méthode utilise l'information de dégénérescence pour dessiner un graphe [3]. Les nœuds sont dessinés dans des cercles concentriques à l'aide de coordonnées polaires où le rayon de la coordonnée d'un nœud dépend de sa dégénérescence et de celle de ses voisins.

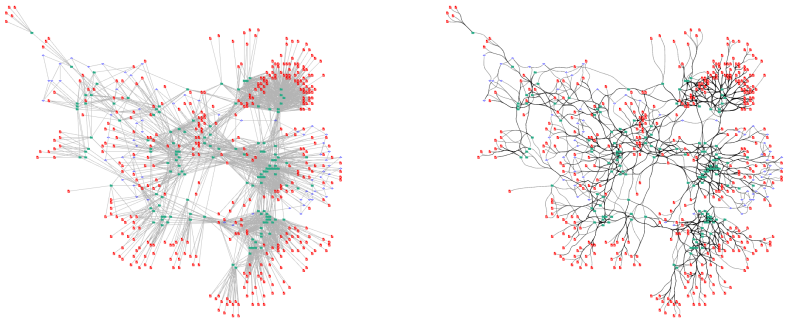
La méthode HDE [32] extrait d'abord m pivots du graphe qui correspondent à m points de vue différents. Elle construit une matrice de distance entre les $|V|$ nœuds du graph et ces m pivots sur laquelle est appliquée une Analyse en Composante Principale (ACP) qui retient les premières dimensions pour le dessin en 2 ou 3 dimensions. ACE [45] utilise les vecteurs propres de la matrice Laplacienne du graphe. Les calculs sont accélérés grâce à l'utilisation d'une méthode algébrique qui calcule une approximation des vecteurs propres à l'aide d'une hiérarchie de matrices : les vecteurs propres de la plus haute matrice sont calculés puis utilisés pour estimer ceux des niveaux précédents. Les d vecteurs propres les plus représentatifs sont utilisés pour un dessin en d dimensions. Ces méthodes offrent de bonnes performances calculatoires, mais la qualité visuelle des résultats n'est satisfaisante que pour les grilles [6].

La figure 8.3 illustre l'application de trois algorithmes de dessins sur un même graphe.

8.1.3. Faisceautage d'arêtes

Lorsque les nœuds d'un graphe sont positionnés, il faut dessiner les arêtes. La façon la plus simple consiste à les représenter sous la forme d'une ligne non brisée entre le nœud source et le nœud destination (voir la figure 8.4a page suivante). Cependant, cette approche crée énormément de bruit : beaucoup de pixels sont utilisés pour dessiner ces arêtes, et beaucoup de nœuds sont potentiellement traversés par des arêtes issues d'autres nœuds source/destination. Représenter les arêtes avec une courbe simple rend le dessin potentiellement plus agréable à regarder, mais ne résout pas du tout le problème de superposition.

De tels soucis sont corrigibles en utilisant des approches de faisceautage d'arêtes, ou « Edge bundling » dans la littérature anglaise. [50]. Le principe est de permettre aux arêtes de suivre des courbes complexes plutôt que des chemins droits afin d'éviter des zones, ou des nœuds, et de regrouper plusieurs arêtes en faisceaux afin de limiter le nombre de pixels utilisés. La figure 8.4b page suivante illustre l'application d'une méthode de faisceautage sur le graphe du dépôt git de cette HDR.



(a) Pas de faisceautage d'arêtes. Représentation des arêtes avec une ligne non brisée.

(b) Faisceautage d'arêtes. Représentation des arêtes avec des b-splines cubiques.

Figure 8.4. – Dessins du graphe du dépôt git de cette HDR avec et sans utilisation de faisceautage d'arêtes. Le plongement des nœuds est identique dans les deux représentations. Dans la seconde, moins de pixels sont utilisés pour représenter les arêtes.

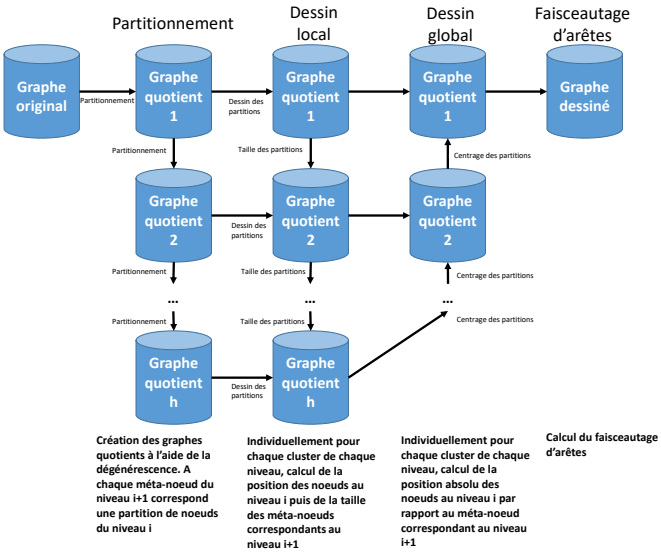


Figure 8.5. – Représentation des données manipulées dans l'approche de dessin hiérarchique basée sur le calcul.

Différentes taxonomies de méthodes de faisceautages existent [22, 50]. Nous avons choisi d'utiliser celle de Fröhlich et al. Le calcul des faisceaux peut être fait suivant trois stratégies majeures : la diminution d'un coût, l'utilisation d'informations issues de l'espace géométrique, ou d'informations issues de l'espace image.

Parmi les méthodes basées sur la minimisation d'un coût, nous pouvons citer certaines qui cherchent à minimiser la quantité de pixels utilisés pour représenter les arêtes [25, 55] et d'autres qui cherchent à minimiser une fonction d'énergie sur un modèle de force qui détermine la courbe des arêtes [37, 67].

Les méthodes géométriques cherchent à générer un maillage en dehors des emplacements contenant des nœuds afin de router les arêtes dessus (*p. ex.* [43, 48]).

Les deux premières familles de méthodes travaillent sur des espaces manipulés avant le dessin. D'autres approches travaillent directement sur l'espace image. En général, ces méthodes sont itératives en essayant de dessiner les arêtes dans une image puis en collectant des informations afin de modifier leur représentation lors d'une prochaine itération [38, 62].

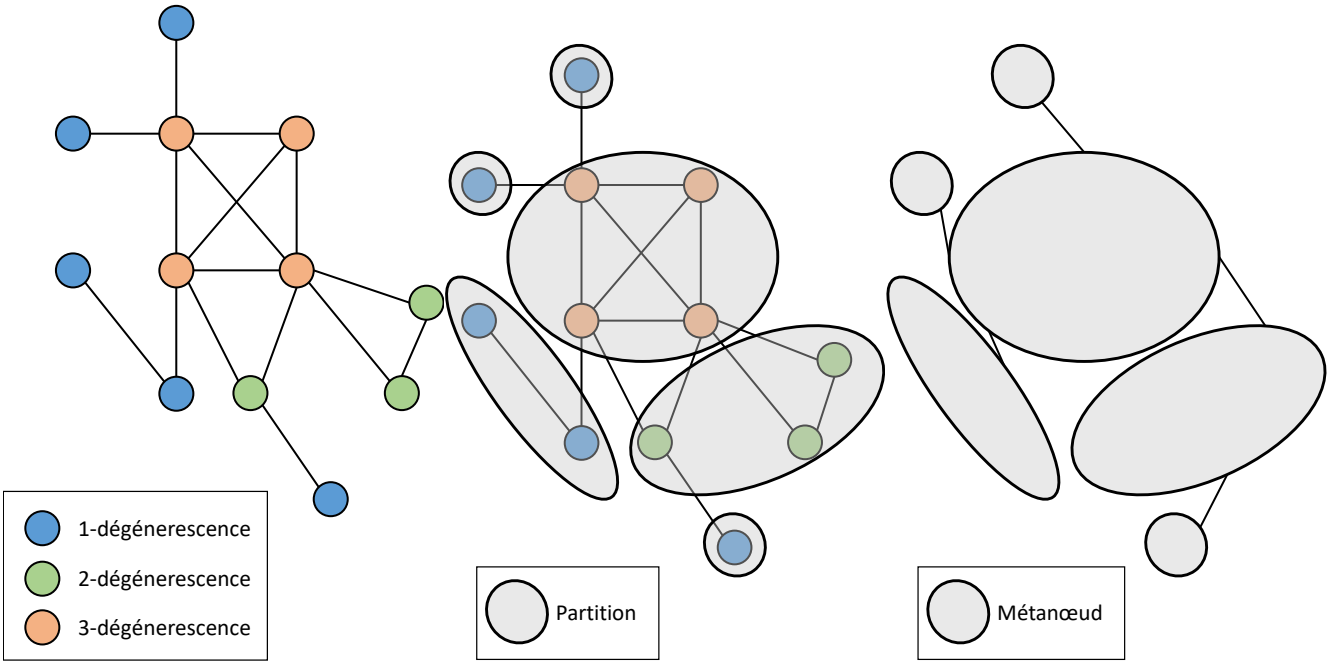
8.2. Utilisation de la dégénérescence pour le dessin multiniveaux de grands graphes

En faisant l'hypothèse qu'un partitionnement de données cohérent avec la topologie du graphe devrait améliorer la qualité du partitionnement et du dessin, nous avons proposé Hierarchical Corness Based Layout (HCBL)[28] qui repose sur la décomposition hiérarchique du graphe en fonction de sa valeur k de dégénérescence, associé à l'utilisation de faisceautage d'arêtes. Cette approche a permis de (a) calculer les positions des nœuds de grands graphes beaucoup plus rapidement qu'en utilisant des méthodes de dessin de l'état de l'art [29] en (b) mettant en avant les cœurs du réseau afin de faciliter la détection visuelle de communauté et (c) fournissant un dessin qui minimise le bruit en étant libre de superposition de nœuds.

8.2.1. Présentation de HCBL

Principe de l'approche

Nous proposons une approche multiniveaux « bottom-up » (voir la figure 8.5). Un partitionnement hiérarchique du graphe est d'abord réalisé en utilisant le graphe quotient de chaque niveau associé à ses mesures de dégénérescence. Cette procédure se termine lorsque le graphe courant est un arbre ou



(a) Dégénerescence d'un graphe quelconque **(b)** Partitionnement suivant la dégénérescence et les composantes connexes **(c)** Graphe quotient résultat

Figure 8.6. – Illustration du partitionnement d'un graphe quelconque grâce à l'information de dégénérescence. Le partitionnement est appliqué récursivement jusqu'à obtenir un arbre..

contient une seule composante. Ensuite les graphes quotient de la hiérarchie sont dessinés de façon « bottom-up ». Cela permet d'éviter des chevauchements de nœuds, car la taille d'un métanœud est spécifiée comme étant la taille de la boite englobante de la partition concernée. Finalement les positions finales des nœuds originaux sont récursivement calculées en traversant la hiérarchie du haut vers le bas. Une variante de la technique de faisceautage de Holten [36] regroupe les arêtes le long de l'arbre hiérarchique (*c.-à-d.* l'arbre qui représente le clustering) afin de faciliter l'identification des dégénérés.

Partitionnement hiérarchique basé sur la dégénérescence

HCBL calcule les informations de dégénérescence [9] de chaque nœud du graphe et le partitionne suivant les composantes connexes de chacune des classes d'équivalence. Cette étape produit un partitionnement à plat où chaque partition est connexe et contient des nœuds ayant la même dégénérescence. HCBL calcule ensuite le graphe quotient [12] associé à cette partition afin d'obtenir un premier niveau d'abstraction du réseau. Chaque nœud du graphe quotient représente une partition connectée de nœuds ayant la même valeur de dégénérescence. Les mêmes étapes sont répétées itérativement sur le graphe quotient résultat jusqu'à obtenir une unique partition. Même si le nombre d'itérations, et donc le nombre de niveaux de la hiérarchie, n'a pas été borné, nos expériences montrent qu'il dépasse rarement 10. La figure 8.6 illustre le concept sur un niveau de la hiérarchie pour un graphe quelconque.

Dessin de partition sensible à l'aire et la topologie

HCBL dessine les graphes de quotient de la hiérarchie avec une approche « bottom-up » en utilisant des algorithmes de dessin de graphes qui assurent l'absence de recouvrement de nœuds à n'importe quel niveau de la hiérarchie. En s'inspirant de TopoLayout [6], HCBL utilise des algorithmes de dessins dédiés à la topologie de la partition à dessiner. Trois cas sont considérés pour une partition $C = (V_C, E_C)$:

- Si C a une densité $\frac{|V_C|(|V_C|-1)}{2}$ supérieure à $0,8$ (*c.-à-d.* est une quasi-clique) : HCBL utilise une méthode de dessin circulaire.
- Si C est un arbre : HCBL utilise une version sensible à l'espace de dessin (*c.-à-d.* il n 'y a pas de recouvrement de nœuds, même si ils sont de taille différente, et la taille de l'espace utilisé pour le dessin est minimisée lorsque c'est possible) de Rings [64] par Archambault et al. [5]. Cette méthode est particulièrement efficace pour mettre en évidence les sous-arbres isomorphes, ainsi que les symétries.

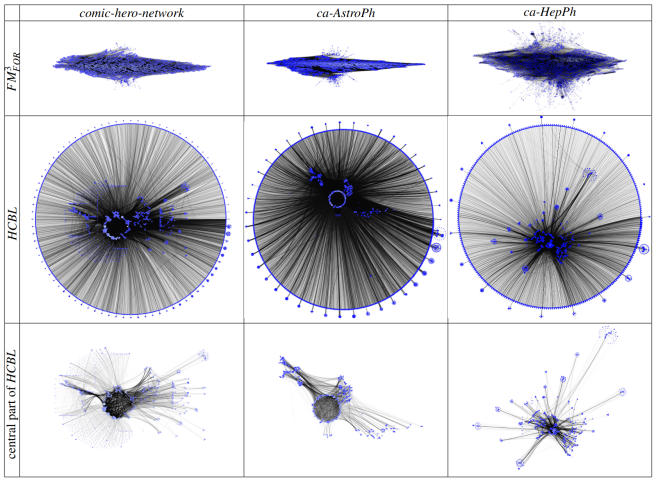


Figure 8.7. – Comparaison de HCBL (2e ligne) à FM3 (1re ligne) sur certains graphes étudiés. La 3e ligne représente la partie centrale du graphe. On voit clairement que notre approche diminue l'aspect pelote de poils.

— Sinon, HCBL utilise FM^3 [29] suivi de Fast Overlap Removal (FOR) [18] (modifié pour considérer des nœuds circulaires, plutôt que des nœuds carrés).

Ces trois cas sont suffisamment génériques pour être présents dans la plupart des graphes.

La taille du dessin d'une partition est reportée comme étant la taille du métanœud qui la représente. Ainsi, le métagraphe qui contient ce métanœud sera dessiné de telle façon à laisser l'espace nécessaire à la partition dans le dessin final.

Expansion des métanœuds

Une fois que la position des nœuds et métanœuds a été calculée relativement à leur parent dans la hiérarchie, il est nécessaire de calculer leur coordonnée finale absolue. Cette opération est effectuée en suivant une approche « top-down » en centrant le dessin d'une partition sur la coordonnée du métanœud qui la représente. Comme (i) la taille de chaque métanœud correspond à la taille de la partition qu'il représente, et que (ii) nous avons utilisé des méthodes de dessins sans recouvrement de nœuds, cette substitution ne génère pas de recouvrement de nœuds.

Faisceautage d'arêtes

Même si cette étape est indépendante du placement des nœuds, l'étape finale d'HCBL consiste à réduire le bruit dû aux croisements d'arêtes. Une méthode de faisceautage [36] est utilisée sur toutes les arêtes inter partitions afin de faciliter la détection visuelle de partitions sur le dessin global du graphe.

En dessinant les métanœuds de la hiérarchie, HCBL a également construit une représentation de cet arbre hiérarchique tout en positionnant ses nœuds aux coordonnées absolues des métanœuds correspondants (les feuilles de cet arbre correspondent aux nœuds du graphe initial, tandis que les nœuds correspondent aux métanœuds des différents niveaux de la hiérarchie). Contrairement à la méthode originale de faisceautage qui utilise le centre des partitions comme point de contrôle pour router les arêtes, HCBL utilise la bordure des partitions. Pour chaque arête e, la métaarête M_l représentant e dans le niveau hiérarchique le plus haut où les métanœuds source et destination sont des clusters différents est identifiée. Son intersection avec ses extrémités représente les deux points de contrôle du milieu de l'arête. Les autres points de contrôle vers la source (respectivement vers la destination) sont calculés itérativement en calculant l'intersection entre le segment qui rejoint le centre du métanœud courant (respectivement la destination) et le dernier point de contrôle et le périmètre du métanœud courant.

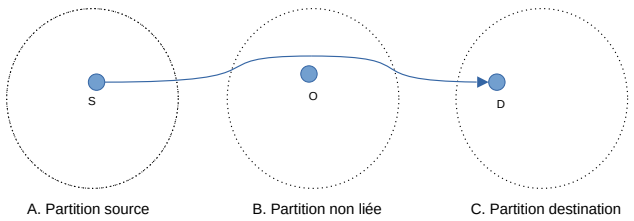
8.2.2. Résultats de HCBL

Pour évaluer HCBL, nous avons utilisé un jeu de données composé de 23 graphes réels. Leur nombre de nœuds varie de 4039 à 8797692 avec une médiane de 60388 alors que le nombre d'arêtes varie de 28980 à 6899377 avec une médiane de 408102. Plusieurs graphes sont issus de la base SNAP [49] : ca-AstroPh, ca-CondMat, ca-GrQc, ca-HepPh, ca-HepTh, cit-HepPh, cit-HepTh, cit- Patents, email-Enron, email-EuAll, facebook combined, gplus combined, soc-Epinions1, soc-LiveJournal1, socpokec- profiles, soc-Slashdot0811, twitter combined, wiki- Vote, wiki-Talk. pgraph est un graphe d'homologie de proteine [1] ; Cheswick-2005 est une tomographie d'internet générée en 2005 par le Cheswick's Internet Mapping Project [14] ; hero-social-network and comic-hero-network represent relations between Marvel characters [2].

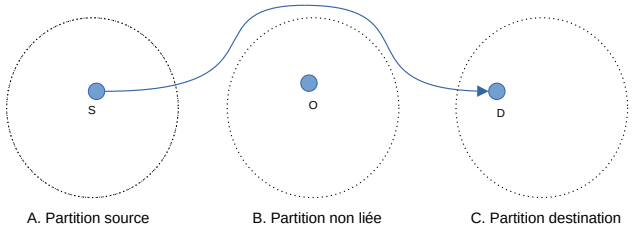
HCBL est comparé à (i) FM^3, (ii) à FM^3 suivit de notre version modifiée de *FOR* (FM^3_{FOR}), et à (iii) une variante de HCBL qui utilise uniquement FM^3_{FOR} pour dessiner les partitions $HCBL_{FM^3_{FOR}}$.

L'évaluation détaillée est disponible dans le papier associé [28]. On peut retenir que :

— les temps de calcul HCBL sont meilleurs que ceux des méthodes de base FM^3 et FM^3_{FOR} ainsi que de $HCBL_{FM^3_{FOR}}$ pour tous les graphes, excepté *soc-LiveJournal1*, probablement car dessiner tous les sous-graphes avec FM^3 produit de petits clusters et réduit la probabilité de corriger des problèmes de superposition.
— les résultats sont visuellement plus attirants que les méthodes de l'état de l'art. On peut noter la diminution significative de l'effet pelote de poils.



(a) Faisceautage standard. L'arête de S vers D évite O, mais traverse la partition B qui n'a aucun lien avec A et C.



(b) Faisceautage attendu. L'arête de S vers D évite O, ainsi que la parition B qui n'a aucun lien avec A et C.

Figure 8.8. – Illustration du souci de traversée de partitions non corrélées par les approches standards de faisceautage d'arêtes.

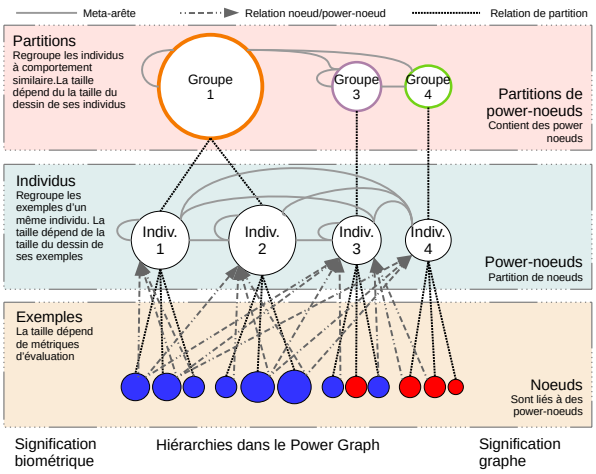


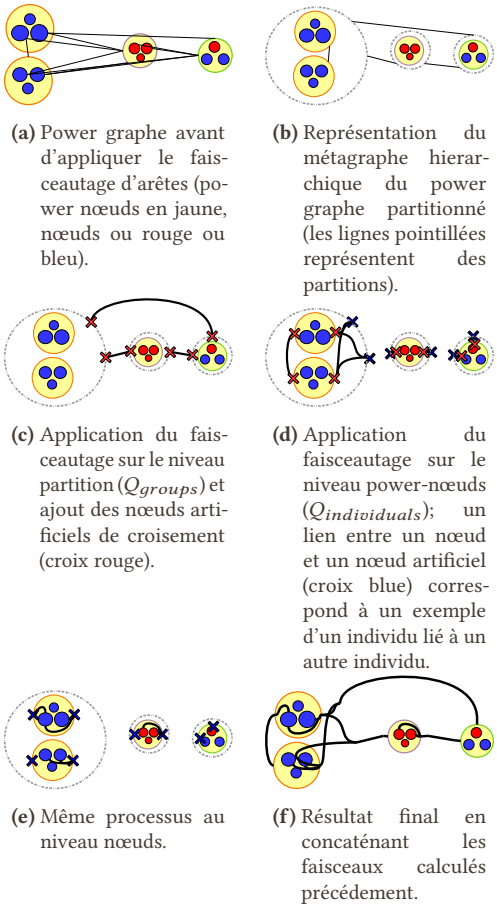
Figure 8.9. – Hierarchie et relation du power graphe partitionné en 8.10a

8.3. Faisceautage multi-échelle d'arêtes moins bruité

La section 8.2 a brièvement présenté une technique de faisceautage d'arêtes multi-échelle qui aide à lire les représentations de graphes ayant une décomposition hierarchique. Cependant elle ne cherche pas à minimiser les superpositions nœuds/partitions ; en particulier, rien n'empêche une arête de traverser une partition composée de nœuds n'appartenant à aucune sous-partion incluant les nœuds de départ ou d'arrivée (voir la figure 8.8). Cette section présente une nouvelle méthode qui prend en compte ce type de contraintes lors du calcul des points de controle des arêtes [27]. Elle a été mise au point dans un contexte particulier d'évaluation de système biométrique qui manipule des « power graphes » (voir la section 13.1). En raison de ce contexte, elle est présentée comme si le partitionnement hierarchique n'avait que deux niveaux, cependant, elle est trivialement généralisable à n'importe quel graphe hierarchique de profondeur plus élevée.

8.3.1. Présentation de la méthode de faisceautage hiérarchique

Pour s'assurer que les arêtes ne traversent pas des partitions étrangères aux partitions source et destination, il suffit d'utiliser une approche « top bottom » qui calcule le faisceautage en partant du niveau de hierarchie le plus proche de la racine de l'arbre de partitionnement, et en redescendant progressivement dans la hiérarchie pour raffiner les arêtes.



(a) Power graphe avant d'appliquer le faisceautage d'arêtes (power nœuds en jaune, nœuds ou rouge ou bleu).

(b) Représentation du métagraphe hierarchique du power graphe partitionné (les lignes pointillées représentent des partitions).

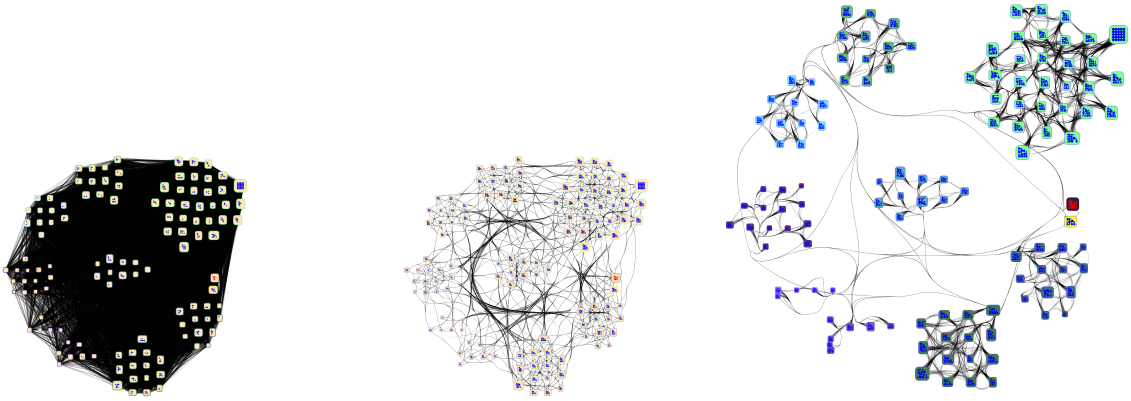
(c) Application du faisceautage sur le niveau partition (Q_{groups}) et ajout des nœuds artificiels de croisement (croix rouge).

(d) Application du faisceautage sur le niveau power-nœuds ($Q_{individuals}$); un lien entre un nœud et un nœud artificiel (croix blue) correspond à un exemple d'un individu lié à un autre individu.

(e) Même processus au niveau nœuds.

(f) Résultat final en concaténant les faisceaux calculés précédement.

Figure 8.10. – Description du pipeline de partitionnement hiéarchique évitant les partitions non liées aux nœuds source et destination. Le graphe de partitionnement est en figure 8.9.



(a) Visualisation sans faisceautage d'arêtes. **(b)** Utilisation d'un algorithme de faisceautage d'arête standard. **(c)** Application de notre méthode de faisceautage d'arêtes. La différence des coordonnées des nœuds ne vient pas de la méthode de faisceautage.

Figure 8.11. – Evaluation de la méthode sur un graphe représentant le résultat d'un système d'authentification biométrique.

La première étape consiste à calculer le faisceautage des arêtes du graphe quotient représentant la partition du niveau le moins haut dans la hiérarchie (la racine de l'arbre hierarchique). Cette étape permet de générer les fragments d'arêtes qui contournent les partitions de niveau plus profond de la hierarchie. Les intersections entre chaque arête et le contour de ses nœuds source et destination sont identifiées et collectées pour l'étape suivante.

L'étape suivante consiste à calculer les portions des faisceaux depuis le niveau de hierarchie précédent vers le niveau de hiérarchie suivant. Le calcul doit être fait indépendement pour chaque métanœud (*c.-à-d.* partition dans le niveau à traiter) de la partition précédente : un nouveau graphe est créé à partir des métanœud de la partition actuelle auquels sont ajoutés des nœuds qui représentent les intersections calculées précédement (ils sont naturellement positionnés à l'emplacement de ces intersections), les arêtes correspondent aux métaarêtes ainsi que des arêtes entre les nœuds d'intersection et les métanœuds contenant les nœuds originaux impliqué dans l'arête originale. Un calcul de faisceautage est effectué sur ce graphe et les intersections entre les faisceaux et les métanœuds sont collectés. Le processus est répété jusqu'à atteindre le niveau final du partitionnement (*c.-à-d.* les nœuds originaux).

Les arêtes finales sont simplement constituées de la concaténation des fragments d'arêtes calculés précédement. La figure 8.9 page précédente illustre l'algorithme sur le cas particulier évalué dans la section suivante.

8.3.2. Evaluation de la méthode de faisceautage hiérarchique

L'approche a été évaluée dans un contexte d'évaluation de système biométrique où les nœuds représentent des données biométrique et les partitions des utilisateurs (les données d'un utilisateur sont partitionnées dans la partition qui représente cet utilisateur). Un niveau supplémentaire de partition permet de regrouper les utilisateurs au comportement similaire. Il y a donc trois niveaux de partitionnement. Contrairement à la description de la méthode, les arêtes ne relient pas deux nœuds, mais un nœud et une partition utilisateur. Nous utilisons donc le concept de power graphe [57] pour modéliser le graphe et l'algorithme de calcul des faisceaux est légèrement modifié pour permettre à un coté des arêtes de s'arèter au niveau de la partition individu, plutot que d'un nœud donnée biométrique (voir la figure 8.9 page précédente).

La figure 8.11 illustre le résultat sur un graphe qui représente le résultat d'un algorithme d'authentification biométrique sur une base de données quelconque.

- L'image (a) représente le graphe sans utiliser de faisceautage d'arêtes : on voit clairement que celui-ci est tellement dense qu'il n'ai pas possible de distinguer des motifs de relations.
- L'image (b) représente le même graphe dessiné en utilisant une méthode standard de faisceautage d'arêtes : le rendu est moins bruité, mais il y a toujours beaucoup de croisements d'arêtes (dus à la traversé de partitions non liées) qui rendent difficile la détection de motifs de relation.
- L'image (c) représente ce même graphe dessiné en utilisant notre méthode de faisceautage d'arêtes : l'espace utilisé par les arêtes est nettement moins important, et il est plus facile de distinguer les motifs de relations (cependant, il est toujours nécessaire de disposer de systèmes d'intéractions pour désambiguiser le rendu).

Chapitre 9.

Représentation de masses de données tabulaires dans une grille

9.1. Introduction à la représentation de masses de données tabulaires

9.1.1. Contexte

Le chapitre 8 concernait la visualisation de masses de données stockées sous forme de graphes. Il s'agit plutôt d'un cas particulier, car les données sont généralement disponibles sous forme tabulaire (voir de tenseur multidimensionnel dans le cas de l'analyse de réseaux de neurones profonds) dans lesquelles chaque exemple est représenté par un vecteur.

Il est commun d'utiliser des approches de plongement de données pour les représenter dans un nuage de point 2D. Différentes méthodes ont été proposées au cours de ces 50 dernières années telles que MDS [46], IsoMap [63], LLE [56], T-SNE [65], ou plus récemment Umap [53]. Elles ont toutes leurs forces et faiblesses, tendent à faire en sorte que des exemples proches soient projetés proches, et des exemples éloignés soient projetés éloignés. Cependant, la quantité d'information perdue avec la réduction de dimensions est relativement importante. Ainsi, la plupart des méthodes partagent ces deux faiblesses : (a) il y a beaucoup de gaspillage d'espace (*c.-à-d.* des zones vides dans le dessin), et (b) il n'y a aucune garantie que deux points ne soient pas positionnés au même endroit (c'est vrai dans l'espace de projection qui est continu, et amplifié dans l'espace de dessin qui est discret). La figure 9.1 page suivante illustre ces deux problèmes.

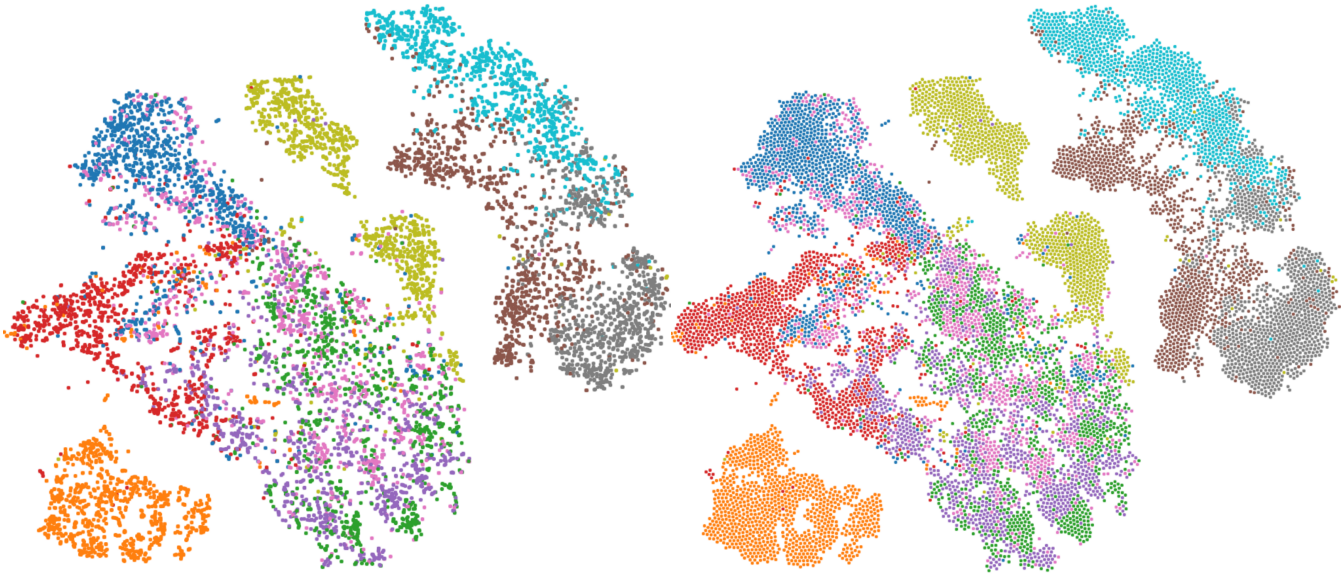
Pour cette raison, nous avons choisi une autre approche avec la thèse d'Adrien Halnaut : nous utilisons des projections orientées pixels qui cherchent à minimiser le nombre de pixels utilisés, tout en empêchant la superposition de points.

9.1.2. Visualisation de données en espace contraint

Les projections de données standards listées précédemment sont efficaces pour représenter les données sur un écran, mais leur construction nécessite d'utiliser des zones de dessins vides pour séparer des groupes d'éléments. Il en résulte souvent un chevauchement des données lorsque l'espace de dessin est trop petit. Il existe diverses méthodes de projection qui visent à optimiser la lisibilité par rapport à l'encodage précis de la distance dans la représentation des données. Cette section donne un aperçu de ces méthodes existantes. Nous avons considéré qu'il existe deux familles de méthodes : les méthodes de projection sur une courbe et les méthodes de projection en grille.

Méthodes de projection sur une courbe

Les méthodes orientées pixels permettent de minimiser la perte de pixels (*c.-à-d.* la non-utilisation de pixels dans le dessin) en utilisant des méthodes de représentation compactes. Elles reposent souvent sur l'utilisation de courbes. Ces courbes peuvent être des spirales, ou des courbes fractales qui sont un moyen efficace de représenter les jeux de données : le principe de base consiste à ordonner le jeu de données initial, puis à projeter ces exemples ordonnés le long de la courbe (*c.-à-d.* la donnée triée en position n correspond au n^{e} pixel de la courbe). Les travaux de Keim sur la visualisation orientée pixel [42] montrent que cette technique est utilisée dans plusieurs domaines d'application. Les courbes fractales telles que celle de Hilbert [35] sont conçues pour préserver la proximité de leurs éléments une fois dessinés dans un espace 2D et conviennent parfaitement à ce type de tâches. Ces courbes fractales ne sont pas limitées à aux structures orthogonales ; GosperMap [7] utilise les courbes de Gosper pour présenter des données hiérarchiques dans une structure visuelle indéterminée : il en résulte une visualisation de type carte géographique. On considère ce genre d'approche comme étant une projection de données dans un espace 1D (l'ordre dans la courbe fractale), ce qui se fait au prix d'une grande perte d'information dans le cas de données complexes.



(a) Données projetées souffrant de recouvrements. Le diamètre des nœuds est de 6 pixels.

(b) Données projetées après suppression des recouvrements. Le diamètre des nœuds est de 5 pixels.

Source : https://labribkb.github.io/gist/

Figure 9.1. – Illustration de quelques problèmes dans les projections 2D de données hautes Dimension. (a) souffre de problèmes de recouvrement corrigés dans (b) : il est difficile d'estimer correctement la densité locale des points. (b) n'a plus de recouvrement (au prix d'un dessin plus grand ou d'une taille d'éléments plus petite). Cependant, la majorité de l'espace ne contient pas d'information.

Méthodes de projection en grille

Les méthodes de projection basées sur une grille peuvent être classées en deux sous-ensembles : le premier projette les données directement dans une grille à partir d'un espace à hautes dimensions, tandis que l'autre utilise une méthode de réduction dimensionnelle intermédiaire avant d'organiser le résultat de dimension inférieure dans une grille.

IsoMatch [21] utilise la technique de réduction de dimension IsoMap [63] pour projeter des données depuis un espace de hautes dimensions vers un espace bidimensionnel, suivi de l'algorithme hongrois [47] pour les adapter dans une structure spécifique (généralement une grille, mais la méthode fonctionne, quelle que soit la forme). L'algorithme hongrois permet de résoudre des problèmes d'affectation en $O(n^4)$, ce qui rend cette méthode difficilement utilisable avec un jeu de données de taille raisonnable.

OoDAnalyzer [13] est un outil qui utilise une vue basée sur une grille pour détecter les valeurs aberrantes d'un jeu de données. La génération de sa grille repose sur une méthode d'arrangement différente de celle utilisée par IsoMatch : elle utilise tout d'abord t-SNE pour projeter les données hautement dimensionnelles dans un espace bidimensionnel, et construit la grille comme un problème d'affectation linéaire. Ce problème est d'abord simplifié en construisant un graphe basé sur les k-voisins les plus proches de chaque élément, suivi d'un algorithme glouton pour imposer à chaque nœud du graphe d'être de degré k. Ce graphe est ensuite transformé en une grille en utilisant la méthode de Jonker-Volgenant [40]. La complexité globale du processus d'agencement de la grille est $O\left(k.N^2\right)$ avec une valeur recommandée de k à 100 pour un arrangement précis de 2000 éléments projetés.

Self-Sorting Map (SSM) [61] arrange directement les données à hautes dimensions sur une grille de telle sorte que la distance entre les éléments arrangés représente la dissimilarité entre eux. Les éléments sont d'abord disposés aléatoirement dans une grille, puis échangés intelligemment pour maximiser la corrélation entre les distances et une métrique de dissimilarité donnée. SSM divise progressivement la grille et parvient à produire rapidement un résultat grâce à une complexité donnée de $O\left(L\,(logN)^2\right)$ et peut être réduite en utilisant le calcul parallèle. Cette méthode est efficace, mais seule l'information de dissimilarité est utilisée pour organiser les données d'entrée.

Fast Linear Assignment Sorting (FLAS) [8] est une méthode créée pour arranger des images dans une grille, mais l'algo-

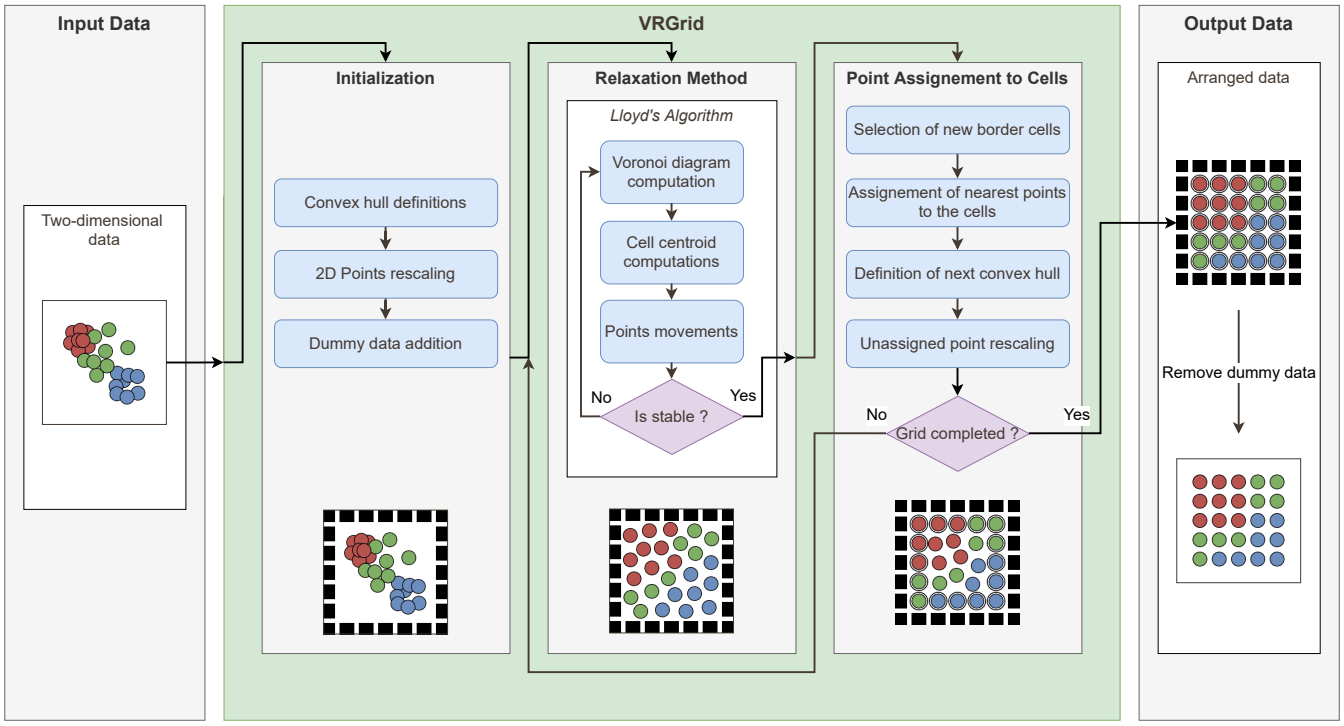


Figure 9.2. – Résumé de l'algorithme VRGRid [31] décrit en section 9.2.

rithme peut fonctionner avec n'importe quel type de données en entrée. D'après ses auteurs, cette méthode est inspirée de « Self Organizing Map (SOM) »[44] avec l'utilisation d'une grille contenant des données filtrées suivant leur voisinage et de SSM [61] avec la notion de permutation de cellules. Une version lente et exacte correspond à : la méthode affecte aléatoirement les données dans la grille, calcule une version filtrée (*c.-à-d.* moyenne selon un voisinage) de la grille, puis affecte à nouveau les données en minimisant leur distance avec les cellules filtrées de la grille (en utilisant Jonker-Volgenant [40] qui est extrêmement coûteux). Le processus est répété plusieurs fois en réduisant la taille du voisinage. La version rapide remplace l'affectation exacte en affectant un sous-ensemble de positions choisies aléatoirement.

D'autres alternatives cherchent plutôt à projeter les données dans un espace de coordonnées entières sans minimiser l'espace utilisé pour la représentation des données. Elles ne nous intéressent donc pas particulièrement, même si elles méritent d'être citées : DGrid [34] et HaGrid [15].

9.2. Projection de données dans une grille : Voronoi Relaxation Grid

L'approche que nous proposons [31] dans la thèse d'Adrien Halnaut (la plupart des illustrations proviennent de son manuscrit de thèse) se positionne en tant que méthode de post-traitement après une projection 2D dans le but de désambiguïser cette représentation qui contient beaucoup de superpositions. On peut considérer qu'elle est plus proche d'IsoMatch (qui a une étape de projection 2D définie par défaut comme étant IsoMap) que SSM (qui fait ses calculs directement avec les données dans l'espace de très haute dimension), tout en ayant une complexité intermédiaire. De façon itérative, elle va chercher à positionner les points de l'espace 2D dans la grille en construisant cette grille depuis ses bordures extérieures vers le centre. La méthode de relaxation du diagramme de Voronoi [51] permet de gérer le déplacement des points. Elle est capable de positionner les points dans une quasi-grille. L'approche est composée de trois étapes majeures : l'initialisation, suivi de la relaxation, et l'affectation appliquées de façon itérative à un sous-ensemble du jeu de données qui réduit au fur et à mesure des itérations (voir la figure 9.2).

9.2.1. Description de VRGRid

Initialisation

Soit un jeu de données P composé de n points 2D, G la grille dans laquelle positionner les points de P. G est une grille carrée composée de $(\lceil\sqrt{n}\rceil+2)\times(\lceil\sqrt{n}\rceil+2)$ cellules réparties en deux ensembles disjoints : les $\lceil\sqrt{n}\rceil\times\lceil\sqrt{n}\rceil$ cellules qui accueilleront les points de P et $4\times(\lceil\sqrt{n}\rceil+1)$ cellules

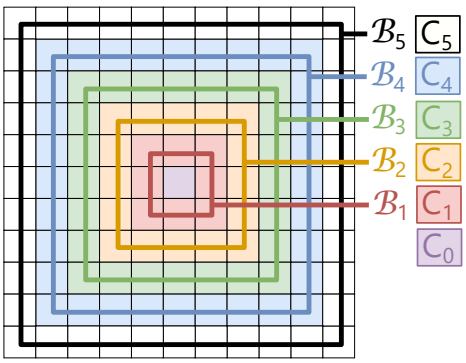


Figure 9.3. – Illustration des enveloppes convexes $\mathcal{B}_0$ (qui est inexistante par construction en réalité) à $\mathcal{B}5$ associées à leurs cellules de bordures C_0 à C_5 pour un jeu de données à projeter sur une grille 9×9.

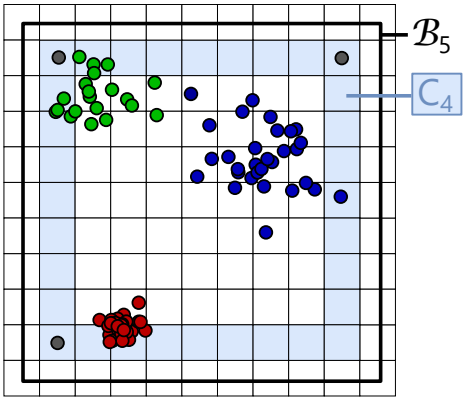


Figure 9.4. – Illustration de l'initialisation d'un jeu de données de 78 points. Il a été normalisé pour être contenu dans la bordure extérieure $\mathcal{B}5$. 3 points artificiels ont été ajoutés sur la prochaine bordure.

additionnelles pour faciliter la description de l'algorithme et qui composent une l'enveloppe convexe de la grille. On note $\mathcal{B}_0, \mathcal{B}_1, \cdots, \mathcal{B}_{\lceil\sqrt{n}/2\rceil+1}$ les enveloppes convexes centrées au milieu de la grille. Chaque enveloppe convexe $\mathcal{B}_i$ représente une bordure carrée de w_i cellules de large stockées dans C_i (voir la figure 9.3) :

$$w_i = \begin{cases} 2i - 1 & \text{si } \lceil\sqrt{n} + 1\rceil \text{ est impair} \\ 2(i+1) & \text{si } \lceil\sqrt{n} + 1\rceil \text{ est pair} \end{cases} \tag{9.1}$$

Si $\lceil\sqrt{n} + 1\rceil$ est impair, C_0 est composé de la cellule centrale de G. Le but de VRGRid est d'itérativement arranger les points de P sur les bordures en commençant par la bordure extérieure et en terminant sur C_0.

L'étape suivante consiste à normaliser les coordonnées des points de P pour être contenues par la bordure extérieure $\mathcal{B}_{\lceil\sqrt{n}/2\rceil+1}$.

Si $|P| < \lceil\sqrt{n}\rceil^2$, il y a plus de cellules dans la grille que de points à positionner, et les étapes suivantes ne fonctionnent pas comme attendu (il y aura un trou au milieu du dessin final). Des points artificiels sont ajoutés (mais non conservés dans le dessin final) et traités de la même façon que ceux de P dans le reste de la méthode. Il sont positionnés uniformément dans $\mathcal{B}_{\lceil\sqrt{n}/2\rceil}$.

La figure 9.4 illustre l'état des structures de données après l'initialisation d'un jeu de données de 78 points et l'ajout de 3 points artificiels ($78 + 3 = 9 \times 9$).

Relaxation

Les points de P sont itérativement affectés aux k bordures précédemment définies. À la i^{e} itération, l'espace courant est défini par l'enveloppe convexe $\mathcal{B}_{k-i}$, et les points à affecter correspondent à $P_i = P_{i-1} \setminus C_{k-i}$ (*c.-à-d.* les points de P sans ceux positionnés sur les bordures précédentes) avec $P_0 = P$.

L'algorithme de Lloyd [51] est utilisé pour distribuer les points P_i uniformément à l'intérieur de la surface définie par $\mathcal{B}_{k-i}$, avec pour objectif d'affecter certains d'entre eux à C_{k-i}. Sa convergence a été seulement prouvée en 1D [17] et en 2D [19] lorsque les données sont dans un espace borné convexe, ce qui est notre cas grâce aux bordures carrées. Son fonctionnement est relativement simple sur un jeu de donnée D en 2D :

1. Le diagramme de Vornoi V est calculé sur D. À chaque point $d \in D$ est associée une cellule de Voronoi $v_d \in V$.
2. Le centroïde $C(v_d)$ de chaque cellule v_d est calculé.
3. Chaque point d est déplacé à son $C(v_d)$ correspondant.

Ces trois étapes sont répétées jusqu'a stabilisation : la distance entre chaque point et son centroïde est inférieure à un seuil.

Dans notre cas, $D = P_i \bigcup C_{k-i}$ avec les points C_{k-i} non traités dans l'étape 3 ; ils servent à contraindre l'espace de relaxation.

La figure figure 9.5 page ci-contre illustre la relaxation de la première itération de l'exemple en figure 9.4. Étant donné que seule la position des points positionnés proche de la bordure extérieure nous intéresse dans l'étape suivante, il est possible d'accélérer VRGRid en stoppant l'étape de Lloyd avant stabilisation.

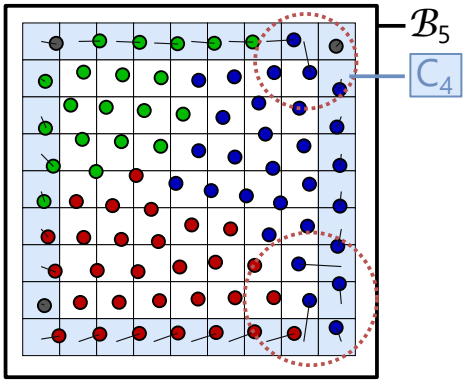


Figure 9.5. – Résultat de l'application de Lloyd sur la 1re itération de VRGRid. Les segments représentent les points de P_0 les plus proches des cellules de C_4.

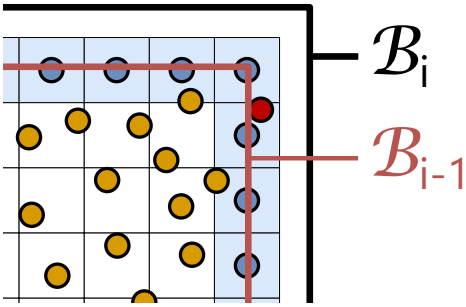


Figure 9.6. – Résultat de l'affectation sur une bordure, et illustration d'un point (le rouge) qui n'a pas été affecté sur la bordure courante et qui est positionné à l'extérieur.

Affectation

À la fin de l'étape de relaxation de l'itération courante, il est possible d'affecter des points à la bordure actuelle. Ainsi, pour chaque cellule $c \in C_{k-i-1}$ VRGRid calcule le point le plus proche dans $p \in P_i$ afin de l'affecter à la position de c.

VRGRid prépare ensuite l'itération $i+1$. L'enveloppe convexe $\mathcal{B}_{k-i-1}$ définie la zone pour la prochaine application de la méthode de relaxation et C_{k-i-2} la prochaine bordure sur laquelle affecter les points. Cependant, il n'y a pas de garantie que les points de P_{i+1} soient dans $\mathcal{B}_{k-i-1}$ après l'affectation sur C_{k-i-1}. En effet, comme la relaxation est appliquée sur l'espace délimité par $\mathcal{B}_{k-i-1}$, n'importe quel point de P_i peut être proche de la bordure sans être affecté à une cellule de C_{k-i-1} car suffisamment d'autres ont pu l'être. La figure 9.6 illustre un tel problème. Le contourner reste trivial : il suffit de normaliser les points de P_{i+1} en s'assurant qu'ils soient tous dans la prochaine enveloppe convexe.

Fin des itérations

Les étapes précédentes sont répétées avec pour objectif d'affecter à la bordure courante certains points de P_{i+1} dans l'itération $i+1$. La procédure se termine une fois que tous les points ont été affectés sur toutes les bordures.

Analyse de la complexité algorithmique

Le papier de VRGRid [31] détaille l'analyse de sa complexité algorithmique en détail. La complexité totale est de :

$$O\left(i \cdot n^{\frac{3}{2}} \cdot log(n)\right) \tag{9.2}$$

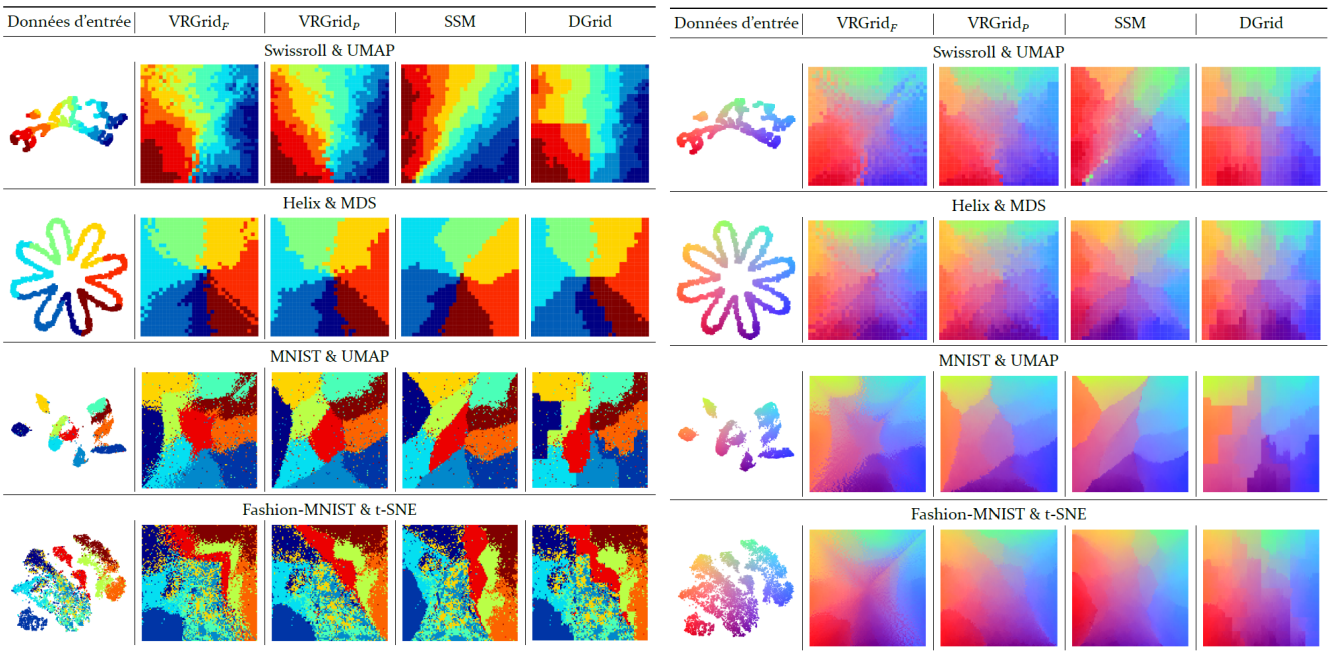
avec i le nombre d'itérations maximales de l'algorithme de Lloyd et n le nombre de cellules à positionner. Cette borne théorique ne semble cependant pas être atteinte en pratique.

9.2.2. Évaluation de VRGRid

Nous comparons VRGrid aux méthodes de l'état de l'art SSM et DGrid sur différents aspects : la préservation de la distance entre paire de points, la préservation du voisinage, le positionnement relatif par paire, le positionnement global et enfin le temps de calcul. La comparaison n'est pas faite avec OoDAnalyzer (car cette méthode effectue un filtrage des données et ne les affiche pas toutes) ou IsoMatch (car elle n'est pas capable de gérer des jeux de données aussi importants que ceux utilisés lors de notre évaluation).

La précision de VRGrid peut être contrôlée en modifiant ϵ et en fixant le nombre d'itérations i de LLoyd. Des valeurs ϵ plus petites donnent une représentation plus précise des données, mais un temps de traitement plus lent, alors que valeurs plus élevées permettent un temps de traitement plus rapide, mais un positionnement moins précis. De la même manière, des valeurs i plus élevées entraîneraient une meilleure représentation des données au détriment d'un traitement plus long. Nous testons deux configurations de VRGrid : VRGridP calcule l'algorithme de Lloyd jusqu'à la configuration stable avec une valeur ϵ fixée à 10^{-3}. VRGridF utilise un nombre fixe d'itérations i, ce qui permet d'obtenir des temps de traitement plus rapides au prix d'une dégradation de la qualité. Nous fixons i à 10 afin que l'algorithme de Lloyd puisse limiter la relaxation principalement sur les points les plus proches de la frontière de l'enveloppe convexe au lieu de la zone de points la plus dense.

La figure 9.7a page suivante illustre le fonctionnement de VRGRid en comparaison de SSM et DGRid sur quelques jeux de données en colorant les points suivant leur étiquette. La figure 9.7b page suivante fait de même en colorant les points selon leur positionnement initial. Nous observons que chaque méthode souffre de différents problèmes de distorsion. Les



(a) La couleur des points dépend de leur étiquette.

(b) La couleur des points dépend de leur position initiale.

Figure 9.7. – Comparaison de la projection VRGRid sur quelques jeux de données avec SSM et DGRid.

résultats de DGrid sont nettement en dessous des autres et SSM semble ne pas respecter le voisinage de façon exagérée.

Des résultats plus détaillés sont présents dans le manuscrit d'Adrien Halanaut [30]. On peut retenir globalement que VRGrid respecte mieux la préservation de distance que les autres méthodes. Contrairement à ce que les figures précédentes suggèrent, VRGRid est moins bon que SSM sur le respect de la préservation du voisinage (SSM fait de temps en temps de grosses erreurs, mais le respecte plutôt bien en général). VRGrid est également gagnant sur le positionnement relatif par rapport aux autres méthodes.

Chapitre 10.

Discussion sur la visualisation de masses de données

Cette partie a présenté quelques travaux sur de la Visualisation d'Information (VI) pour les masses de données auxquels j'ai contribué avec plusieurs collègues du LaBRI. J'ai pu ainsi monter en compétences en analyse et VI, notamment sur les aspects visualisation de très grands graphes, et visualisation orientée pixels ; compétences qui sont totalement à l'opposé de celles que j'ai acquises lors de mon doctorat et de mon post-doctorat. Ce chapitre effectue la synthèse de mes travaux sur ces thématiques. Il décrit également brièvement des travaux pertinents, mais non abordés, pour cette partie.

10.1. Dessin de grands graphes

Les graphes permettent de représenter des données où des entités sont reliées par différents types de relation. Il est possible de dessiner ce type d'objet mathématique en utilisant une vue nœud lien : chaque entité est représentée par un symbole (généralement un disque), et les relations pour un lien visuel (généralement un segment). De nombreuses méthodes existent pour calculer le positionnement des nœuds à l'écran, ainsi que les points de contrôle de arêtes lorsqu'elles sont représentées par des polylignes. Elles fonctionnent relativement bien sur des graphes de taille raisonnable, mais sont en échec sur de grands graphes (soit, car la complexité algorithmique est trop importante, soit car le rendu souffre de l'effet pelote de poils).

10.1.1. Plongement des nœuds

Nous avons proposé HCBL (présenté en section 8.2), une approche hiérarchique de placement de nœuds de grands graphes qui repose sur un partitionnement hiérarchique basé sur la valeur de dégénérescence et sur l'utilisation de techniques de dessin de sous-graphes spécifiques à leur topologie. Les résultats sont compétitifs avec les méthodes de l'état de l'art aussi bien en termes de temps de calcul que de qualité de rendu.

Il serait intéressant d'améliorer les performances de calcul d'une telle méthode en utilisant des algorithmes facilement implémentables dans une architecture de type « cloud computing » afin de permettre le dessin de graphes encore plus gros. Cependant, de telles approches restent majoritairement itératives, et la complexité algorithmique dépendra toujours du nombre d'itérations nécessaires pour obtenir un plongement stable. Il semble plus intéressant de mettre au point des approches efficaces qui ne sont pas itératives et sont capables d'effectuer directement le plongement des nœuds. Utiliser des réseaux de neurones est une approche prometteuse qui est étudiée dans le section 14.2.

10.1.2. Faisceautage des arêtes

Nous avons également proposé une méthode de calcul de points de contrôles sur les arêtes d'un graphe afin d'effectuer du faisceautage d'arêtes qui prend en compte les contraintes d'un partitionnement hiérarchique prédéfini et respecté dans le positionnement de ses nœuds (présenté en section 8.3). De cette façon, les arêtes ne traversent pas des zones du dessin sur lesquelles il n'y a pas de relation avec les nœuds de départ et d'arrivée, ainsi qu'avec leurs partitions hiérarchiques associées. Les résultats montrent que le rendu est moins bruité que les méthodes standard.

L'approche fonctionne convenablement lorsque les partitions peuvent être représentées facilement avec des diagrammes de confinement (comme les power graphes utilisés dans nos travaux actuels). Cependant, elle ne peut pas fonctionner lorsque le plongement des partitions ne peut pas être représenté dans des diagrammes de confinement : *p. ex.* des données où la position des nœuds ne reflète pas exactement le partitionnement hiérarchique, comme dans le cas d'un plongement 2D de vecteur d'attribution de données étiquetées de façon hiérarchique. Ce problème ouvert nécessite toujours d'être investigué (voir figure 10.1 page suivante).

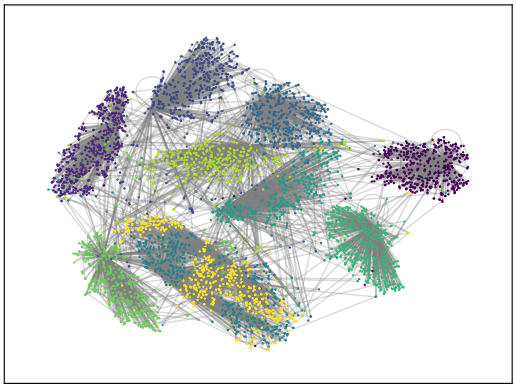


Figure 10.1. – Illustration d'un jeu de données sur lequel il n'est pas possible d'utiliser le faisceautage hiérarchique présenté dans cette partie. Ici les partitions (représentées par des couleurs différentes) sont éclatées dans le plongement (une même partition peut être à plusieurs endroits, une partition peut être visuellement incluse dans une autre).

10.2. Représentation de masses de données tabulaire dans une grille

Nous proposons VRGrid (voir la section 9.2) pour positionner des données projetées dans un espace 2D (qui peut être de taille conséquente) dans un espace grille (qui est plus compact). Elle utilise une approche qui fixe les points de façon itérative, bordure par bordure. Lors de certaines étapes de l'algorithme, des points peuvent être positionnés convenablement alors que la méthode continue d'appeler l'algorithme de LLoyd, ce qui demande des calculs inutiles et provoque des déplacements supplémentaires. Je pense qu'il est possible de mettre en place une stratégie qui ne cherche pas à fixer les points bordure par bordure, mais à les fixer en temps réel dès qu'ils sont suffisamment proches d'une cellule extérieure non fixée. Une évaluation est nécessaire afin de déterminer l'impact de cette approche sur la qualité du rendu final, car il sera nécessairement différent de celui de VRGRid.

Bien que VRGRid propose des résultats visuels intéressants, la méthode reste calculatoirement plus coûteuse que SSM [61]. Je pense qu'il faut revoir sa condition d'arrêt et trouver un moyen de créer une approche hybride qui combine les deux approches pour bénéficier de leurs avantages respectifs et diminuer au maximum leurs défauts.

10.3. Travaux non abordés

J'ai réalisé d'autres travaux sur la visualisation de masses de données. Certains ont été publiés, d'autres non.

Lors de mon postdoctorat sur le détection de fraudes dans les systèmes de paiement bancaire, j'ai mis en place des stratégies de dessins de graphes de transactions bancaires afin d'aider à comprendre certains motifs de transactions. Ces stratégies reposaient majoritairement sur de la définition de glyphes pour représenter les entités dans un graphe, ainsi que la modélisation de ce graph (*c.-à-d.* que représentent les nœuds et les arêtes). Le plongement de ce graphe étant délégué aux algorithmes de force de graphviz.

Dans ses travaux de thèse, Adrien Halnaut a mis au point un protocole d'évaluation de représentation de données haute dimension dans un nuage de points, ou dans une grille. Une préétude a montré que les approches de dessin en grille permettent de mieux représenter les informations de distances dans $\mathbb{R}^n$ que les nuages de points. Les utilisateurs n'étant pas habitués à une telle représentation, il leur est plus difficile de la prendre en main. Ces travaux doivent être poursuivis pour mieux appréhender les limites de telles approches et en créer une qui les corrige.

Chapitre 11.

Bibliographie de la visualisation de masses de données

[1] A. T. Adai, S. V. Date, S. Wieland et E. M. Marcotte. « LGL: creating a map of protein function with an algorithm for visualizing very large biological networks ». Dans : *Journal of molecular biology* 340.1 (2004), p. 179-190 (cf. p. 52).

[2] R. Alberich, J. Miro-Julia et F. Rosselló. « Marvel Universe looks almost like a real social network ». Dans : (2002). arXiv : cond-mat/0202174 (cf. p. 52).

[3] J. I. Alvarez-Hamelin, L. Dall'Asta, A. Barrat et A. Vespignani. « k-core decomposition: A tool for the visualization of large scale networks ». Dans : (2005). arXiv : cs/0504107 (cf. p. 49).

[4] F. J. Anscombe. « Graphs in statistical analysis ». Dans : *The american statistician* 27.1 (1973), p. 17-21 (cf. p. 44).

[5] D. Archambault, T. Munzner et D. Auber. « Smashing peacocks further: Drawing quasi-trees from biconnected components ». Dans : *IEEE Transactions on Visualization and Computer Graphics* 12.5 (2006), p. 813-820 (cf. p. 49, 51).

[6] D. Archambault, T. Munzner et D. Auber. « Topolayout: Multilevel graph layout by topological features ». Dans : *IEEE transactions on visualization and computer graphics* 13.2 (2007), p. 305-317 (cf. p. 49, 51).

[7] D. Auber, C. Huet, A. Lambert, B. Renoust, A. Sallaberry et A. Saulnier. « Gospermap: Using a gosper curve for laying out hierarchical data ». Dans : *IEEE transactions on visualization and computer graphics* 19.11 (2013), p. 1820-1832 (cf. p. 55).

[8] K. U. Barthel, N. Hezel, K. Jung et K. Schall. « Improved Evaluation and Generation Of Grid Layouts Using Distance Preservation Quality and Linear Assignment Sorting ». Dans : *Computer Graphics Forum* 42.1 (2023), p. 261-276. eprint : https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.14718 (cf. p. 56).

[9] V. Batagelj et M. Zaversnik. « An O (m) algorithm for cores decomposition of networks ». Dans : (2003). arXiv : cs/0310049 (cf. p. 51).

[10] G. D. Battista, P. Eades, R. Tamassia et I. G. Tollis. *Graph drawing: algorithms for the visualization of graphs*. Prentice Hall PTR, 1998 (cf. p. 47).

[11] M. Baur, U. Brandes, M. Gaertler et D. Wagner. « Drawing the AS graph in 2.5 dimensions ». Dans : *International Symposium on Graph Drawing*. Springer. 2004, p. 43-48 (cf. p. 49).

[12] R. Brockenauer et S. Cornelsen. « Drawing clusters and hierarchies ». Dans : *Drawing graphs*. Springer, 2001, p. 193-227 (cf. p. 47, 51).

[13] C. Chen, J. Yuan, Y. Lu, Y. Liu, H. Su, S. Yuan et S. Liu. « Oodanalyzer: Interactive analysis of out-of-distribution samples ». Dans : *IEEE transactions on visualization and computer graphics* 27.7 (2020), p. 3335-3349 (cf. p. 56).

[14] B. Cheswick, H. Burch et S. Branigan. « Mapping and visualizing the Internet. » Dans : *USENIX Annual Technical Conference, General Track*. Citeseer. 2000, p. 1-12 (cf. p. 52).

[15] R. Cutura, C. Morariu, Z. Cheng, Y. Wang, D. Weiskopf et M. Sedlmair. « Hagrid — Gridify Scatterplots with Hilbert and Gosper Curves ». Dans : *Proceedings of the 14th International Symposium on Visual Information Communication and Interaction*. VINCI '21. Association for Computing Machinery, 2021. isbn : 9781450386470 (cf. p. 57).

[16] W. Didimo et F. Montecchiani. « Fast layout computation of hierarchically clustered networks: Algorithmic advances and experimental analysis ». Dans : *2012 16th International Conference on Information Visualisation*. IEEE. 2012, p. 18-23 (cf. p. 49).

[17] Q. Du, M. Emelianenko et L. Ju. « Convergence of the Lloyd algorithm for computing centroidal Voronoi tessellations ». Dans : *SIAM journal on numerical analysis* 44.1 (2006), p. 102-119 (cf. p. 58).

[18] T. Dwyer, K. Marriott et P. J. Stuckey. « Fast node overlap removal ». Dans : *International Symposium on Graph Drawing*. Springer. 2005, p. 153-164 (cf. p. 52).

[19] M. Emelianenko, L. Ju et A. Rand. « Nondegeneracy and weak global convergence of the Lloyd algorithm in Rˆd ». Dans : *SIAM Journal on Numerical Analysis* 46.3 (2008), p. 1423-1441 (cf. p. 58).

[20] A. Frick, A. Ludwig et H. Mehldau. « A fast adaptive layout algorithm for undirected graphs (extended abstract and system demonstration) ». Dans : *International Symposium on Graph Drawing*. Springer. 1994, p. 388-403 (cf. p. 48).

[21] O. Fried, S. DiVerdi, M. Halber, E. Sizikova et A. Finkelstein. « IsoMatch: Creating informative grid layouts ». Dans : *Computer graphics forum*. T. 34. 2. Wiley Online Library. 2015, p. 155-166 (cf. p. 56).

[22] D. Fröhlich, M. Kammerhofer, S. Kogler et S. Stiboller. *Edge bundling*. Rapp. tech. Graz University of Technology, 2017 (cf. p. 50).

[23] T. M. Fruchterman et E. M. Reingold. « Graph drawing by force-directed placement ». Dans : *Software: Practice and experience* 21.11 (1991), p. 1129-1164 (cf. p. 48, 49).

[24] P. Gajer et S. G. Kobourov. « Grip: Graph drawing with intelligent placement ». Dans : *International Symposium on Graph Drawing*. Springer. 2000, p. 222-228 (cf. p. 48, 49).

[25] E. R. Gansner, Y. Hu, S. North et C. Scheidegger. « Multilevel agglomerative edge bundling for visualizing large graphs ». Dans : *2011 IEEE Pacific Visualization Symposium*. IEEE. 2011, p. 187-194 (cf. p. 50).

[26] H. Gibson, J. Faith et P. Vickers. « A survey of two-dimensional graph layout techniques for information visualisation ». Dans : *Information visualization* 12.3-4 (2013), p. 324-357 (cf. p. 49).

[27] R. **Giot**. « Evaluation of Biometric Authentication Systems through Visualisation of Partitioned and Bundled Powergraphs ». Dans : *Information Visualisation 2017 (IV 2017)*. 2017, p. 104-110 (cf. p. 53).

[28] R. **Giot** et R. Bourqui. « Fast Graph Drawing Algorithm Revealing Networks Cores ». Dans : *Information Visualisation 2015 (IV 2015)*. Conference Publishing Services, 2015, p. 259-264 (cf. p. 50, 52).

[29] S. Hachul et M. Jünger. « Drawing large graphs with a potential-field-based multilevel algorithm ». Dans : *International Symposium on Graph Drawing*. Springer. 2004, p. 285-295 (cf. p. 47-50, 52).

[30] A. Halnaut. « Méthodes et outils d'analyse visuelle pour la compréhension, l'optimisation et l'élaboration de modèles de réseaux de neurones profonds ». Thèse de doct. Université de Bordeaux, 2024. En prép. (cf. p. 60).

[31] A. Halnaut, R. **Giot**, R. Bourqui et D. Auber. « VRGrid: Efficient Transformation of 2D Data into Pixel Grid Layouts ». Dans : *Proceedings of the 26th International Conference Information Visualisation (IV 2022)*. 2022 (cf. p. 57, 59).

[32] D. Harel et Y. Koren. « Graph drawing by high-dimensional embedding ». Dans : *International symposium on graph drawing*. Springer. 2002, p. 207-219 (cf. p. 47, 49).

[33] H. Hauser et H. Schumann. « Visualization Pipeline ». Dans : *Encyclopedia of Database Systems*. Sous la dir. de L. LIU et M. T. ÖZSU. Springer US, 2009, p. 3414-3416. isbn : 978-0-387-39940-9 (cf. p. 44).

[34] G. M. Hilasaca et F. V. Paulovich. « A visual approach for user-guided feature fusion ». Dans : *Anais Estendidos do XXXII Conference on Graphics, Patterns and Images*. SBC. 2019, p. 133-139 (cf. p. 57).

[35] D. Hilbert et D. Hilbert. « Über die stetige Abbildung einer Linie auf ein Flächenstück ». Dans : *Dritter Band: Analysis· Grundlagen der Mathematik· Physik Verschiedenes: Nebst Einer Lebensgeschichte* (1935), p. 1-2 (cf. p. 55).

[36] D. Holten. « Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data ». Dans : *IEEE Transactions on visualization and computer graphics* 12.5 (2006), p. 741-748 (cf. p. 51, 52).

[37] D. Holten et J. J. Van Wijk. « Force-directed edge bundling for graph visualization ». Dans : *Computer graphics forum*. T. 28. 3. Wiley Online Library. 2009, p. 983-990 (cf. p. 50).

[38] C. Hurter, O. Ersoy et A. Telea. « Graph bundling by kernel density estimation ». Dans : *Computer graphics forum*. T. 31. 3pt1. Wiley Online Library. 2012, p. 865-874 (cf. p. 50).

[39] T. Itoh, C. Muelder, K.-L. Ma et J. Sese. « A hybrid space-filling and force-directed layout method for visualizing multiple-category graphs ». Dans : *2009 IEEE Pacific Visualization Symposium*. IEEE. 2009, p. 121-128 (cf. p. 49).

[40] R. Jonker et T. Volgenant. « A shortest augmenting path algorithm for dense and sparse linear assignment problems ». Dans : *DGOR/NSOR: Papers of the 16th Annual Meeting of DGOR in Cooperation with NSOR/Vorträge der 16. Jahrestagung der DGOR zusammen mit der NSOR*. Springer. 1988, p. 622-622 (cf. p. 56, 57).

[41] G. Kant. « Drawing planar graphs using the canonical ordering ». Dans : *Algorithmica* 16 (1996), p. 4-32 (cf. p. 47).

[42] D. A. Keim. « Designing pixel-oriented visualization techniques: Theory and applications ». Dans : *IEEE Transactions on visualization and computer graphics* 6.1 (2000), p. 59-78 (cf. p. 55).

[43] W. Kienreich, R. Wozelka, V. Sabol et C. Seifert. « Graph Visualization Using Hierarchical Edge Routing and Bundling. » Dans : *EuroVA@ EuroVis*. 2012 (cf. p. 50).

[44] T. Kohonen. « The self-organizing map ». Dans : *Proceedings of the IEEE* 78.9 (1990), p. 1464-1480 (cf. p. 57).

[45] Y. Koren, L. Carmel et D. Harel. « Ace: A fast multiscale eigenvectors computation for drawing huge graphs ». Dans : *IEEE Symposium on Information Visualization, 2002. INFOVIS 2002*. IEEE. 2002, p. 137-144 (cf. p. 49).

[46] J. B. Kruskal. *Multidimensional scaling*. 11. Sage, 1978 (cf. p. 55).

[47] H. W. Kuhn. « The Hungarian method for the assignment problem ». Dans : *Naval research logistics quarterly* 2.1-2 (1955), p. 83-97 (cf. p. 56).

[48] A. Lambert, R. Bourqui et D. Auber. « Winding roads: Routing edges into bundles ». Dans : *Computer graphics forum*. T. 29. 3. Wiley Online Library. 2010, p. 853-862 (cf. p. 50).

[49] J. Leskovec et A. Krevl. *SNAP Datasets: Stanford Large Network Dataset Collection*. http://snap.stanford.edu/data. 2014 (cf. p. 52).

[50] A. Lhuillier, C. Hurter et A. Telea. « State of the art in edge and trail bundling techniques ». Dans : *Computer Graphics Forum*. T. 36. 3. Wiley Online Library. 2017, p. 619-645 (cf. p. 49, 50).

[51] S. Lloyd. « Least squares quantization in PCM ». Dans : *IEEE transactions on information theory* 28.2 (1982), p. 129-137 (cf. p. 57, 58).

[52] J. Matejka et G. Fitzmaurice. « Same stats, different graphs: generating datasets with varied appearance and identical statistics through simulated annealing ». Dans : *Proceedings of the 2017 CHI conference on human factors in computing systems*. 2017, p. 1290-1294 (cf. p. 44).

[53] L. McInnes, J. Healy et J. Melville. « Umap: Uniform manifold approximation and projection for dimension reduction ». Dans : (2020). arXiv : 1802.03426v3 [stat] (cf. p. 55).

[54] G. A. Miller. « The magical number seven, plus or minus two: Some limits on our capacity for processing information. » Dans : *Psychological review* 63.2 (1956), p. 81 (cf. p. 43).

[55] S. Pupyrev, L. Nachmanson et M. Kaufmann. « Improving layered graph layouts with edge bundling ». Dans : *Graph Drawing: 18th International Symposium, GD 2010, Konstanz, Germany, September 21-24, 2010. Revised Selected Papers 18*. Springer. 2011, p. 329-340 (cf. p. 50).

[56] S. T. Roweis et L. K. Saul. « Nonlinear dimensionality reduction by locally linear embedding ». Dans : *science* 290.5500 (2000), p. 2323-2326 (cf. p. 55).

[57] L. Royer, M. Reimann, B. Andreopoulos et M. Schroeder. « Unraveling protein networks with power graph analysis ». Dans : *PLoS computational biology* 4.7 (2008), e1000108 (cf. p. 54).

[58] S. B. Seidman. « Network structure and minimum degree ». Dans : *Social networks* 5.3 (1983), p. 269-287 (cf. p. 48, 49).

[59] B. Shneiderman. « The eyes have it: A task by data type taxonomy for information visualizations ». Dans : *Proceedings 1996 IEEE symposium on visual languages*. IEEE. 1996, p. 336-343 (cf. p. 43).

[60] J. M. Six et I. G. Tollis. « A framework for circular drawings of networks ». Dans : *International Symposium on Graph Drawing*. Springer. 1999, p. 107-116 (cf. p. 49).

[61] G. Strong et M. Gong. « Self-sorting map: An efficient algorithm for presenting multimedia data in structured layouts ». Dans : *IEEE Transactions on Multimedia* 16.4 (2014), p. 1045-1058 (cf. p. 56, 57, 62).

[62] A. Telea et O. Ersoy. « Image-based edge bundles: Simplified visualization of large graphs ». Dans : *Computer Graphics Forum*. T. 29. 3. Wiley Online Library. 2010, p. 843-852 (cf. p. 50).

[63] J. B. Tenenbaum, V. De Silva et J. C. Langford. « A global geometric framework for nonlinear dimensionality reduction ». Dans : *science* 290.5500 (2000), p. 2319-2323 (cf. p. 55, 56).

[64] S. T. Teoh et M. Kwan-Liu. « RINGS: A technique for visualizing large hierarchies ». Dans : *International Symposium on Graph Drawing*. Springer. 2002, p. 268-275 (cf. p. 51).

[65] L. Van der Maaten et G. Hinton. « Visualizing data using t-SNE. » Dans : *Journal of machine learning research* 9.11 (2008) (cf. p. 55).

[66] C. Wetherell et A. Shannon. « Tidy drawings of trees ». Dans : *IEEE Transactions on software Engineering* 5 (1979), p. 514-520 (cf. p. 47).

[67] H. Zhou, X. Yuan, W. Cui, H. Qu et B. Chen. « Energy-based hierarchical edge clustering of graphs ». Dans : *2008 ieee pacific visualization symposium*. IEEE. 2008, p. 55-61 (cf. p. 50).

Quatrième partie

Symbiose de la visualisation et de l'apprentissage automatique

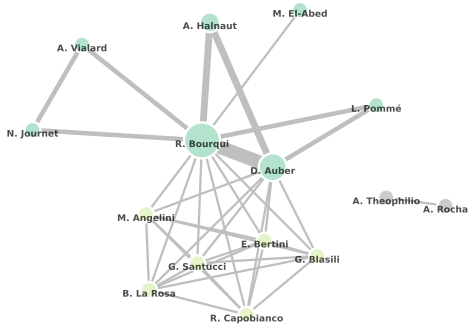
Chapitre 12. Introduction 71
Chapitre 13. Visualisation pour l'apprentissage 73
13.1. Visualisation pour l'authentification biométrique 73
13.1.1. Justification 73
13.1.2. Le Zoo Graph pour contourner les limitations du Zoo Plot 74
13.2. Visualisation pour l'analyse de base de documents étiquetés 76
13.2.1. Justification 76
13.2.2. ACID: une Application pour Classer des Images dans des bases de Données 77
13.3. Visualisation pour la compréhension des réseaux profonds 79
13.3.1. Justification 79
13.3.2. Représentation des flux de données dans un réseau 81
Chapitre 14. Apprentissage automatique pour la visualisation d'information 85
14.1. Analyse automatique de techniques de visualisation 85
14.1.1. Illustration avec la comparaison de dessin de graphes 85
14.2. Dessin de graphes : plongement dans une vue nœuds-liens 87
14.2.1. Introduction 87
14.2.2. Définition de réseaux sur graphe pour le dessin de graphe 87
14.2.3. Résultats 89
14.3. Suppression de chevauchements pour le dessin de graphes 90
14.3.1. Introduction 90
14.3.2. Présentation de Fast Overlap Removal By stochastic gradIent Descent for Graph Drawing 90
Chapitre 15. Discussion sur la symbiose de la visualisation d'information et l'apprentissage automatique 93
15.1. Visualisation pour l'apprentissage automatique 93
15.1.1. Application à l'authentification biométrique 93
15.1.2. Application à l'apprentissage supervisé 93
15.1.3. Application aux réseaux de neurones profonds 94
15.2. Apprentissage automatique pour la visualisation d'information 94
15.2.1. Analyse automatique de techniques de visualisation d'information 94
15.2.2. Dessin de graphes 95
15.2.3. Suppression de chevauchements 96
Chapitre 16. Bibliographie de la symbiose de la visualisation et de l'apprentissage automatique 97

Chapitre 12.

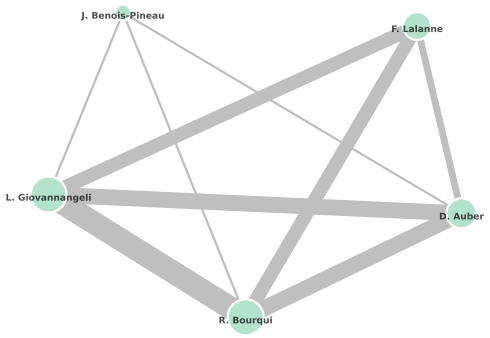
Introduction

Contexte

Pour rappel, les parties parties II et III ont présenté les deux thématiques de recherche auxquelles j'ai participé depuis mon arrivée au LaBRI : « l'Apprentissage Automatique (AA) » (*c.-à-d.* les compétences acquises lors de ma thèse) dans la partie II page 11 et la « Visualisation d'Information (VI) » (*c.-à-d.* les compétences acquises lors de mon arrivée au LaBRI) dans la partie III page 41.



(a) Graph ego centré de mes co-auteurs sur les travaux liés à la Visualisation d'Information (VI) pour l'Apprentissage Automatique (AA).



(b) Graph ego centré de mes co-auteurs sur les travaux liés à la l'Apprentissage Automatique (AA) pour la Visualisation d'Information (VI).

Figure 12.1. – Graphes égo centré de mes co-auteurs sur les travaux liés à la symbiose entre entre la Visualisation d'Information (VI) et l'Apprentissage Automatique (AA).

Cette partie se focalise sur la suite logique et novatrice liée à ces deux thématiques de recherche : mes travaux sur la symbiose entre la « VI » et l'« AA ». La combinaison « VI pour l'AA » est une nouvelle thématique que j'ai apportée au LaBRI : tout d'abord en travaillant sur des techniques de VI spécifiques à l'authentification biométrique, puis en travaillant sur des techniques ou systèmes plus généraux sur la VI pour les réseaux de neurones profonds, notamment avec la co-direction des travaux de thèse d'Adrien Halnaut, et la participation aux travaux de thèse de Luc-Étienne Pommé-Casseriou. Je vais donc insister plus sur cet aspect. L'association « AA pour la VI » est également une nouvelle thématique au LaBRI, à laquelle j'ai contribué, mais portée par Romain Bourqui, principalement avec la participation aux travaux de thèse de Loann Giovannangeli. Ces nouvelles thématiques novatrices associent toutes deux AA (ce qui correspond plutôt à ma zone de confort) et VI (ce qui correspond plutôt à une prise de risque) afin de faire bénéficier des apports d'une des thématiques dans l'autre et faire avancer l'état de l'art de la seconde.

Le graphe ego-centré de mes co-auteurs sur la symbiose est présenté en figure 12.1. La taille des nœuds et des arêtes est proportionnelle au nombre de publications communes avec les collègues concernés. Les couleurs dépendent d'un algorithme de partitionnement de graphe, mais reflètent globalement bien les origines des collaborations.

Organisation

Les différents concepts détaillés concernant cette thématique sont illustrés dans la figure 12.2 page suivante.

Ainsi le chapitre 13 présente quelques travaux majeurs liés à la VI pour l'AA. Trois projets sont présentés : la VI pour l'authentification biométrique, pour un système de vérification d'étiquetage d'une base de documents et pour l'Apprentissage

Profond (AP). Les contributions pour l'authentification biométrique en section 13.1 se situent principalement sur des techniques de visualisation de performance de système biométrique, et d'Analyse Visuelle et Interactive (AVI) intégrant de telles visualisations associées aux méthodes standards. Les contributions liées à l'analyse d'une base de données en section 13.2 permettent de guider l'utilisateur pour détecter des erreurs potentielles d'étiquetage. Les contributions liées à l'AP en section 13.3 concernent principalement des techniques d'explication globales en utilisant des méthodes de VI capables de tenir l'échelle du nombre de données. Les approches de type « attribution de caractéristiques » ne sont pas présentées dans ce chapitre, car l'aspect VI est minimal (*c.-à-d.* majoritairement affichage d'une carte de chaleur).

Le chapitre 14 présente mes travaux majeurs sur l'AA pour la VI. Ceux-ci sont abordés de façon plus succincte, car je ne suis pas le porteur principal de ces projets ; cependant ma contribution reste significative. Ces travaux concernent l'évaluation automatique de techniques de VI en section 14.1, le dessin de graphes à l'aide de Réseau de Neurones (RN) en section 14.2 et la suppression de chevauchement de données dans les nuages de points à l'aide d'optimisation d'une fonction spécifique avec une descente de gradient en section 14.3.

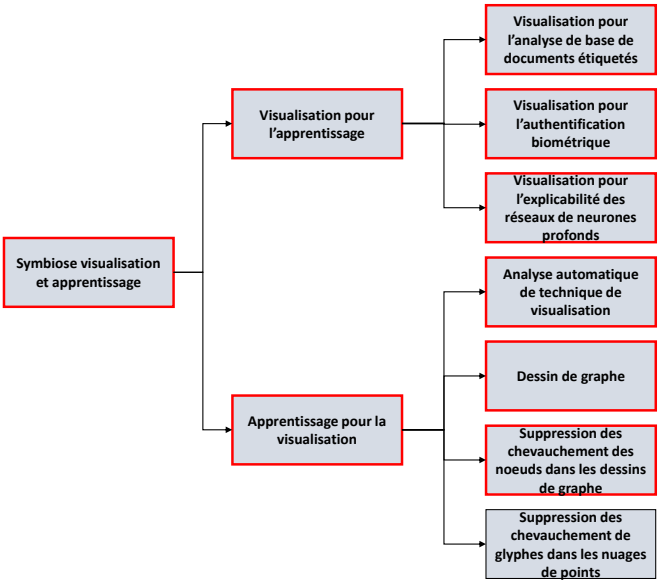


Figure 12.2. – Famille de travaux réalisés dans cette thématique. Seuls ceux encadrés en rouge sont présentés.

Chapitre 13.

Visualisation pour l'apprentissage

Ce chapitre présente mes travaux clés liés à la visualisation pour l'apprentissage.

La section 13.1 concerne la visualisation pour l'authentification biométrique : nous voulons identifier les individus problématiques d'un système d'authentification biométrique dans des travaux effectués conjointement avec Mohammad El-Abed de l'université de Rafik Hariri/Liban. Il s'agit donc d'un domaine d'application spécifique.

La section 13.2 présente une application de technique d'analyse et de visualisation de données pour détecter des erreurs d'étiquetage dans une base de données manuellement étiquetée.

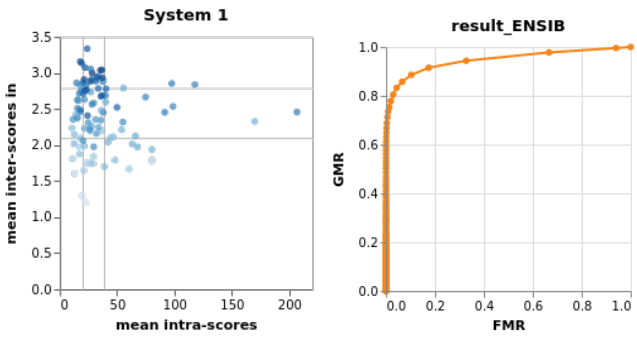
La section 13.3 concerne la visualisation pour l'Apprentissage Profond (AP) : nous sommes intéressés à identifier ce qui amène le réseau à prendre sa décision finale. Cette partie n'est pas dépendante d'un domaine d'application spécifique, bien que les travaux se limitent aux réseaux de convolution appliqués à de la classification d'images.

13.1. Visualisation pour l'authentification biométrique

13.1.1. Justification

Comme tout système d'apprentissage, les systèmes d'authentification biométriques (voir le chapitre 3) font des erreurs de reconnaissance (non-authentification d'un individu authentique, authentification erronée d'un imposteur, erreur lors de la capture de la donnée biométrique...). De façon générale, la littérature scientifique s'intéresse essentiellement aux erreurs effectuées par le module de vérification, et pas celles générées par les autres modules (acquisition, enrôlement...). Les systèmes sont évalués à l'aide de différentes métriques (*p. ex.* Taux de Fausse Acception (TFA), Taux de Faux Rejet (TFR), Taux d'Erreur Égale (TEE), aire sous la courbe d'opération caractéristique/AUC...) [46], voire de méthodes graphiques (*p. ex.* courbe ROC [12] (voir la figure 13.1b), Zoo Plot [71] (voir la figure 13.1a)...) communément acceptées dans la communauté.

Cependant, parmi ces outils, seul le Zoo Plot fournit des informations pour tous les utilisateurs du système ; les autres se contentent de résumés agrégés qui masquent l'information utile. Dans ce Zoo Plot (voir la figure 13.1a), un nuage de point est utilisé pour représenter tous les individus dans un plan 2D avec un point dont la coordonnée x représente le score moyen intraclasse et la coordonnée y représente le score moyen interclasse. Ces scores sont calculés grâce à des bases de test. Le 25e et 75e percentile de ces deux valeurs sont visuellement représentés avec des lignes horizontales et verticales afin d'identifier aisément la zone qui contient 50% des données. Nous visualisons la dispersion des scores, mais nous manquons d'informations telles que les relations de chaque individu avec la performance globale du système. De plus, en raison d'une distribution non uniforme des scores et de l'utilisation d'une échelle linéaire pour les axes, le Zoo Plot représente la majorité des individus dans un petit espace. Il est donc nécessaire de trouver des alternatives.



(a) Zoo plot. **(b)** Courbe ROC.

Figure 13.1. – Exemple de figures utilisées de façon standard d'évaluation d'un même système biométrique. Elles apportent des informations sur la performance, sans en expliquer la raison.

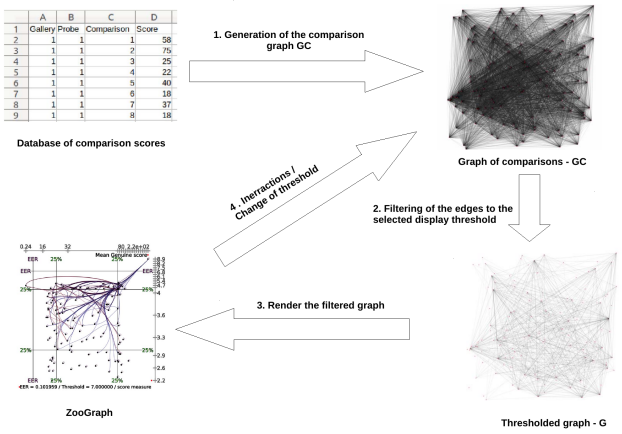


Figure 13.2. – Description du pipeline de construction du Zoo Graph depuis la base de scores de vérification biométrique, jusqu'à la visualisation interactive finale.

13.1.2. Le Zoo Graph pour contourner les limitations du Zoo Plot

Nous avons proposé le Zoo Graph [18] qui est une évolution du Zoo Plot et tente de limiter les aspects négatifs précédemment listés. Il s'agit du premier travail que j'ai réalisé qui fusionne les deux domaines de recherche. Il repose sur une vue nœuds-liens, plutôt qu'un nuage de points, associée à l'utilisation d'une échelle non linéaire pour améliorer la lisibilité de la vue. Cette vue nœuds-liens affiche le graphe $G = (V, E)$ construit à l'aide des étapes suivantes (voir la figure 13.2).

Modélisation d'un graphe de résultat biométrique

La première étape consiste à construire un graphe de comparaison $GC = (VC, EC)$ depuis un jeu de données de scores composés de tuples $t_i = \left(t_i^{requete}, t_i^{modele}, t_i^{comparaison}, t_i^{score}\right)$ interprétés de la façon suivante : t_i^{score} correspond au score de comparaison de la $t_i^{comparaison}$e comparaison de données biométriques de l'individu $t_i^{requete}$ au modèle de l'individu t_i^{modele}. Chaque nœud $vc_u \in VC$ représente un individu de la base de données $t_{\odot}^{requete} \bigcup t_{\odot}^{modele}$. Il y a une arête $\{vc_{u1}, vc_{u2}\} \in EC$ s'il existe au moins un tuple t_j tel que $t_j^{requete} = vc_{u2} \wedge t_j^{modele} = vc_{u1}$. Chaque arête $e \in EC$ est valuée par la liste des *scores* correspondants, ainsi que par leur moyenne *bioScore*. Les nœuds $vc \in VC$ sont également valués par $selfBioScore$ qui correspond à la moyenne des *scores* de $\{vc, vc\}$, $inBioScore$ qui correspond à la moyenne des *scores* de $\bigcup_{u \in voisins_in(vc)} \{u, vc\}$ (les u essaient de se faire passer pour vc), et $outBioScore$ qui correspond à la moyenne des *scores* de $\bigcup_{o \in voisins_out(vc)} \{vc, o\}$ (vc essaie de se faire passer pour les o).

La seconde étape consiste à seuiller GC pour construire G en filtrant toutes les arêtes avec un *bioScore* inférieur à un seuil de décision τ. Ainsi, s'il y a un arc orienté entre deux nœuds de G, cela signifie, qu'en moyenne, le score de reconnaissance entre les données biométriques de l'individu représenté par le nœud source et le modèle de l'individu représenté par le nœud destination est supérieur ou égal à τ. Autrement dit, source arrive souvent à usurper l'identité destination. En milieu opérationnel, les systèmes biométriques sont souvent paramétrés à l'aide d'un tel seuil.

Positionnement des nœuds et encodage visuel

Contrairement au Zoo Plot qui utilise un nuage de points pour représenter les individus (voir la figure 13.1a page précédente), le Zoo Graph utilise une vue nœuds-liens dans laquelle le positionnement des nœuds du graphe dépend à la fois de sa topologie, et des performances de reconnaissance du système biométrique étudié.

Le Zoo Plot utilise une échelle linéaire pour positionner les individus, ce qui peut amener une grande partie des individus à occuper une minorité de l'espace. Bien qu'une telle représentation ait du sens, car elle montre la distribution des valeurs, la superposition des points complexifie l'identification des individus et de leur voisinage. Pour cette raison, nous avons décidé d'appliquer une transformation non linéaire sur les coordonnées (telles qu'elles seraient dans le Zoo Plot) des individus pour permettre aux percentiles à 25% et 75% d'occuper chacun 25% de l'espace et de laisser 50% a de l'espace aux autre individus.

Pour réduire le bruit du à l'affichage des liens, nous utilisons une technique standard de faisceautage d'arêtes [42]. D'autres informations sont encodées et résumées dans le tableau 13.1.

Informations visuelles supplémentaires

Comme dans le Zoo Plot, nous affichons les premiers et derniers 25 percentiles ainsi qu'une graduation pour donner une idée de la dispersion des valeurs. La graduation est nécessaire étant donné que nous appliquons une transformation non linéaire. Une distribution des scores dense dans un petit intervalle d'origine sera représentée par quelques graduations alors qu'une distribution de scores creuse dans un grand intervalle de l'espace d'origine est représentée par plusieurs graduations. Comme le Zoo Graph est dépendant d'un seuil, le seuil de décision est représenté par un axe horizontal et vertical afin de délimiter visuellement les zones qui correspondent à l'acceptation et celles qui correspondent au rejet.

Table 13.1. – Liste des informations utilisées dans le Zoo Graph et comparaison avec le Zoo Plot.

Information	**Encodage de la proposition** (Zoo Graph)	**Encodage de la méthode de base** (Zoo Plot)
Score authentique moyen (bioScoreSelf)	Position X du nœud	Position X du nœud
Score d'attaque moyen (bioScoreOut)	Position Y du nœud	Position Y du nœud
Score d'usurpation moyen (bioScoreIn)	Couleur du nœud	-
Nombre de fois qu'un individu est usurpé (in degree)	Taille du nœud	-
Score de comparaison moyen entre deux individus (bioScore)	Couleur de l'arête (relative mapping)	-
Relation à τ	Présence d'arêtes	-
Exploration	Analyse interactive	-

Interaction

L'analyse d'une image statique d'un Zoo Graph peut ne pas être satisfaisante pour un seuil de vérification τ générant un taux de fausses correspondances élevé : le nombre d'arêtes à dessiner est important et encombre la visualisation, même en utilisant un mécanisme de faisceautage. Il est nécessaire d'utiliser une représentation interactive pour de tels cas.

Rappelez-vous que le Zoo Graph est dessiné comme un graphe dirigé où une arête encode un individu source qui peut souvent être reconnu comme étant l'individu cible. Pour comprendre les interactions d'un individu spécifique avec les autres, nous pouvons mettre l'accent sur le voisinage dirigé du nœud qui le représente [49]. En inspectant les nœuds des individus et leur voisinage (en entrée ou en sortie), l'expert peut identifier pour tout individu problématique, ceux qui peuvent usurper son identité, et au contraire les individus à qui il peut usurper les identités.

Les performances d'un système biométrique dépendent du seuil décision τ ; il est donc intéressant de faire varier sa valeur pour analyser son impact. Notre technique de visualisation peut supporter un filtrage rapide des arêtes, en effet l'étape de rendu de notre workflow s'exécute en temps linéaire. L'utilisateur peut donc modifier ce seuil à la volée et analyser l'apparition/disparition des arêtes en fonction de ce seuil.

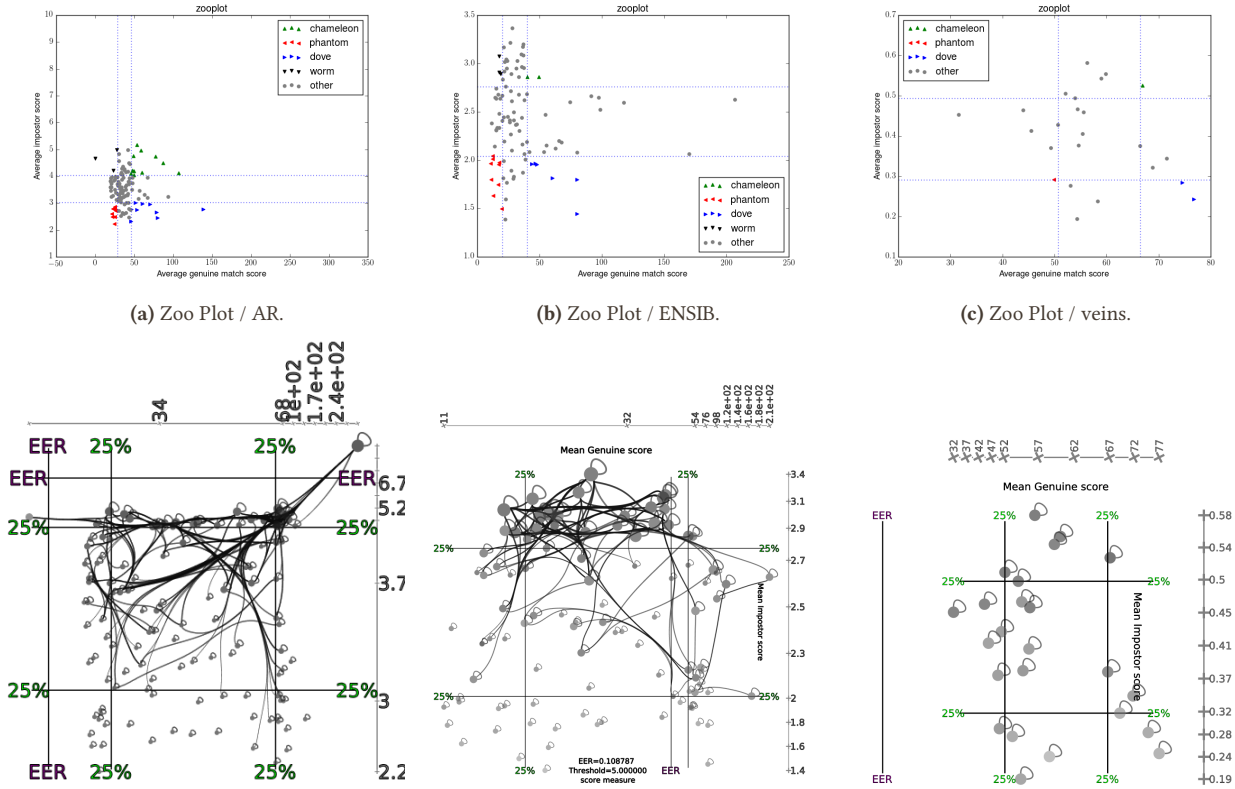
Résultats

La figure 13.3 page suivante illustre la différence de représentation entre le Zoo Plot et le Zoo Graph sur trois bases de données de l'état de l'art.

La projection non linéaire des scores dans l'espace-écran permet de réduire les chevauchements de nœuds ; si nous analysons les résultats du Zoo Graph sur la base AR, l'individu en haut à droite de figure 13.3d page suivante correspond à un individu caché par la légende de la figure 13.3a page suivante). La ligne de seuil (EER) est clairement visible et met en évidence l'individu en haut à gauche qui est incapable de se reconnaître (c'est le seul nœud qui n'a pas d'auto boucles). Il y a une majorité de nœuds qui n'ont pas d'arêtes (à l'exception de leur boucle). Des informations similaires sont obtenues à partir du Zoo Graph de l'ENSIB (figure 13.3e page suivante). Cela valide l'intérêt d'utiliser la projection non linéaire.

Les arêtes du Zoo Graph fournissent clairement des informations supplémentaires informations. Nous voyons rapidement les utilisateurs capables en moyenne de s'authentifier au système (*c.-à-d.* les nœuds avec une boucle) de ceux qui ne sont pas capables en moyenne de s'authentifier au système (*c.-à-d.* les nœuds sans boucle). Dans les systèmes où le FMR est limité (*c.-à-d.* pas plus de 10 %), le nombre d'arêtes (non-boucle) est modéré et nous pouvons voir quelles parties des individus sont responsables de ces erreurs (pour AR, figure 13.3d page suivante, et ENSIB, figure 13.3e page suivante, ces nœuds se trouvent plutôt sur la partie supérieure supérieure de l'image). Cette information n'est pas disponible avec le Zoo Plot. Le pouvoir de similarité entre deux utilisateurs est représenté par la transparence de l'arête ; nous pouvons voir que ce pouvoir est plus faible pour les nœuds plus éloignés de la partie supérieure des images. Ceci valide l'intérêt d'utiliser les arêtes pour montrer les relations entre les individus.

Si nous analysons la taille d'un nœud, nous accédons rapidement aux individus qui posent problème et qui usurpent souvent d'autres individus. Par exemple, dans AR, figure 13.3d page suivante, le plus mauvais individu se trouve en haut à droite de l'image, tandis que dans ENSIB, figure 13.3e page suivante, il y a plus d'individus avec un mauvais comportement (en haut de l'image et même au milieu de l'image). Cela valide également la nécessité de faire correspondre la taille d'un nœud à l'attractivité de son utilisateur.



(a) Zoo Plot / AR. **(b)** Zoo Plot / ENSIB. **(c)** Zoo Plot / veins.

(d) Zoo Graph / AR. **(e)** Zoo Graph / ENSIB. **(f)** Zoo Graph / veins.

Figure 13.3. – Illustration de la différence entre le Zoo Plot et le Zoo Graph pour trois bases de données biométriques.

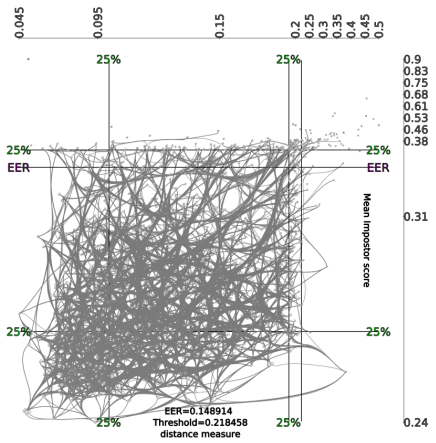


Figure 13.4. – La visualisation est encombrée en raison du grand nombre d'arêtes. Dans ce jeu de données, la mesure est une distance pas un score, donc les pires individus sont sous le EER.

Malgré ces bons points, le Zoo Graph présente quelques limites. La méthode peut échouer lorsqu'il y a trop de problèmes de reconnaissance. Par exemple, dans figure 13.4, le nombre d'arêtes est trop important, ce qui donne une visualisation encombrée. Dans ce cas, il est probablement préférable d'utiliser le Zoo Plot ou de naviguer de manière dans le graphique avec l'outil d'interaction de voisinage.

13.2. Visualisation pour l'analyse de base de documents étiquetés

13.2.1. Justification

L'étiquetage manuel de données est une opération longue et mentalement épuisante : il est à la fois difficile de ne pas faire d'erreurs, et primordial de les détecter au plus tôt. Ceci est d'autant plus vrai dans un cadre professionnel où la qualité de l'étiquetage est contractualisée.

C'est dans ce contexte que nous avons proposé ACID [19]. Nous travaillons sur l'évaluation de la qualité d'un étiquetage préexistant (réalisé dans un environnement industriel). Dans la plupart des cas pratiques, un opérateur étiquette manuellement une base d'images de documents en parcourant séquentiellement les vignettes correspondant aux images. Cette tâche

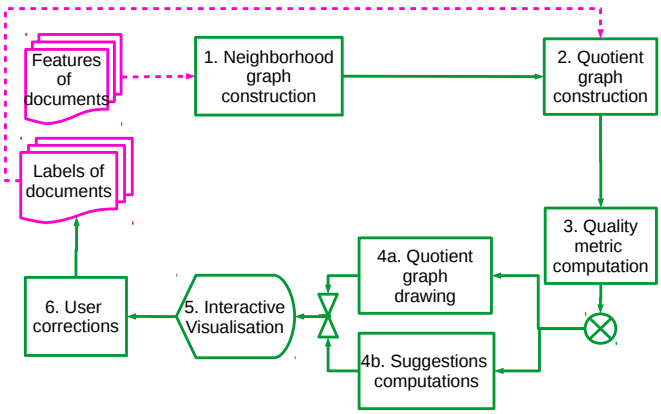


Figure 13.5. – Fonctionnement global d'ACID. Un vecteur de caractéristiques et une étiquette sont donnés pour chaque image qui représente un document. Notre logiciel construit un graphe où les documents sont rassemblés dans des classes en fonction de leur étiquette. Il permet à l'utilisateur de corriger les erreurs de classification. 1- Seules les caractéristiques sont utilisées pour créer le graphe de voisinage reliant les images en fonction de leurs similarités. 2- Le graphe quotient représentant les classes est créé en utilisant le graphe de voisinage et les étiquettes. 3- Une mesure globale est calculée pour indiquer la cohérence entre le graphe de voisinage et le graphe de quotient. 4- Le graphe quotient est affiché. Des corrections de classification sont proposées à l'utilisateur. 5- L'utilisateur peut interagir avec le logiciel pour explorer la base de données d'images. 6- L'utilisateur confirme un changement ; la base de données de documents étiquetés est mise à jour et le nouveau graphe quotient est calculé.

est très répétitive ; de plus, le plan de classement définissant les noms et le nombre de classes est souvent incomplet (*c.-à-d.* amené à évoluer), ce qui conduit à de nombreuses erreurs d'étiquetage. La question est alors d'évaluer si la qualité d'un lot étiqueté est suffisante pour qu'il soit accepté. Notre objectif est de faciliter et d'accélérer cette évaluation qui prend en pratique plus de 1,5 fois le temps nécessaire à l'étiquetage lui-même. Nous proposons un outil interactif pour visualiser les données sous la forme d'un graphe. Ce graphe met en évidence les similarités entre documents ainsi que la qualité de l'étiquetage. Nous définissons des critères sur le graphe qui caractérisent les trois types d'erreurs qu'un opérateur peut faire : une image est mal étiquetée, une classe devrait être découpée en sous-classes plus pertinentes, plusieurs classes devraient être fusionnées en une seule. Ceci nous permet de focaliser l'attention de l'utilisateur sur de potentielles erreurs. Il peut alors plus facilement compter les erreurs d'étiquetage et valider (ou pas) une qualité d'étiquetage conforme aux attentes.

13.2.2. ACID: une Application pour Classer des Images dans des bases de Données

Comme mentionné par Plant et al. [52], le principal avantage de l'utilisation d'une visualisation orientée graphe au lieu d'une navigation séquentielle d'images est d'améliorer la détection de similarité entre les documents. ACID est un outil interactif qui repose sur des techniques de visualisation de l'information appliquées à graphes afin de vérifier la qualité d'un étiquetage d'un jeu de données. Il repose sur une visualisation multiéchelle. Le premier niveau représente les similarités entre les classes de documents, tandis que le second niveau représente les similarités entre les documents d'une classe donnée. En concentrant l'attention de l'utilisateur sur la partie du graphe correspondant aux erreurs d'étiquetage, nous nous assurons que l'ensemble du processus de correction sera plus facile et plus rapide qu'avec la méthode standard.

Le jeu de données de documents est abstrait avec deux graphes : le graphe de voisinage et le graphe quotient [19].
Le graphe de voisinage modélise les données (*c.-à-d.* les documents et leurs similitudes). Pour faciliter sa visualisation et l'identification des erreurs d'étiquetage, il est abstrait sous la forme d'un graphe quotient [56] qui représente les classes de documents et leurs relations. La suite détaille comment ces graphes sont construits.
Le fonctionnement global de la méthode est décrit dans la figure 13.5. Le système a besoin de deux entrées : un vecteur de caractéristiques, et une étiquette pour chaque image (en rose sur la figure). Dans nos expériences, le vecteur de caractéristiques est construit à partir du texte et de l'image, mais il peut varier en fonction des spécificités des bases à utiliser.

Construction du graphe à représenter

Comme décrit dans la figure 13.6 page suivante, le graphe de voisinage G et son graphe quotient G_q sont construits en fonction du vecteur de caractéristiques et de l'étiquette de chaque image de document. Chaque nœud de G correspond à un document ; l'ensemble des arêtes correspond à une similarité orientée entre eux. Il existe une arête orientée entre un nœud n_a et un nœud n_b si n_b est dans les k plus proche voisin de n_a. Dans l'implémentation actuelle, la distance entre deux vecteurs de caractéristiques est la distance euclidienne.

Le graphe quotient G_q est construit à partir du graphe G de telle façon que chaque métanœud du graphe quotient représente une classe de documents ainsi que l'ensemble des nœuds de G appartenant à cette classe. Il existe une méta-arête orientée entre deux métanœuds si au moins un nœud (*c.-à-d.* un document) de la classe source a l'un de ses k plus proche voisin dans la classe cible.

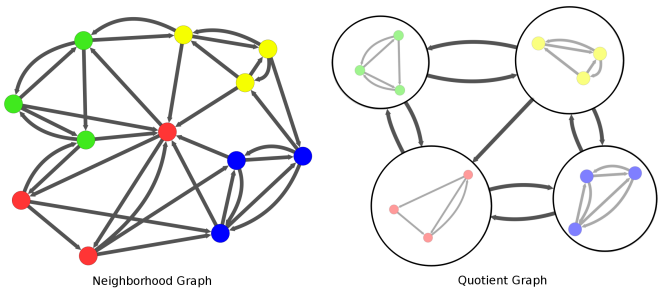


Figure 13.6. – À gauche : le graphe de voisinage modélisant la similarité entre les images de documents, les couleurs indiquent la classe de chaque document. À droite : le graphe quotient correspondant où chaque nœud représente une classe d'images de documents.

Estimation de qualité de l'étiquetage

Nous évaluons la qualité de l'étiquetage à l'aide de mesures qui quantifient la qualité de partitionnement d'un graphe. De telles mesures ont été initialement proposées par la communauté d'analyse des réseaux sociaux afin d'estimer la cohésion des communautés d'individus. Parmi les mesures de qualité disponibles, nous utilisons la mesure MQ [45], adaptée à la structure *k-nn* du graphe.

Nous évaluons la Cohésion Interne *CI* d'une classe en comparant son nombre d'arêtes au nombre maximal d'arêtes qu'elle pourrait avoir dans un tel graphe. Le même principe est utilisé pour évaluer la Cohésion Externe *CE* entre chaque paire de classes de documents.

Dans le meilleur des cas théoriques, toutes les valeurs *CI* sont égales à 1 et toutes les valeurs *CE* sont égales à 0, ce qui signifie que tous les k plus proches voisins d'un document appartiennent à la même classe. En d'autres termes, cela signifie que la classification d'entrée est cohérente avec les caractéristiques calculées.

À partir de ces métriques de qualité locales, nous sommes en mesure de calculer une métrique de Qualité Globale *QG* (plus elle est élevée, mieux c'est) qui est calculée comme la moyenne des cohésions externes soustraite à la cohésion interne moyenne. Le papier associé ACID à [19] décrit plus précisément ces métriques.

Visualisation du graphe et de ses propriétés

C'est le graphe quotient que nous voulons afficher. Nous utilisons FM^3 [29] qui fournit un bon compromis entre temps de calcul et esthétique pour calculer le plongement des métanœuds. Une bonne visualisation du graphe doit également éviter les chevauchements entre les nœuds, mais bien que FM^3 peut prendre en compte la taille des nœuds pour éviter de tels chevauchements, il ne fournit pas cette garantie. Pour supprimer les chevauchements restants, nous utilisons l'algorithme *Fast Overlap Removal* (*FOR*) [10]. Nous avons besoin de dessiner les clusters (*c.-à-d.* les groupes que nous avons détectés comme étant similaires), ainsi que les classes de documents (*c.-à-d.* la vérité terrain). Afin de prendre en compte la surface nécessaire pour dessiner chaque classe lors de la mise en page du métaniveau, nous utilisons une approche ascendante [3, 17]. Le graphe induit (nœuds de la même classe) représenté par chaque métanœud est donc dessiné en premier en utilisant FM^3 et *FOR*. La boîte de délimitation de chaque classe est ensuite utilisée pour définir la taille du métanœud correspondant. Enfin, le graphe quotient est dessiné avec la même combinaison d'algorithmes.

Rendu et interaction

Pour guider l'opérateur dans son processus d'exploration, nous rendons également la cohésion interne avec des couleurs. Comme elle est bornée entre 0 et 1, nous utilisons une correspondance de couleurs du rouge au gris où le gris est toujours associé à une bonne cohésion et le rouge à une mauvaise. L'utilisateur peut ainsi facilement visualiser quelles classes peuvent contenir des erreurs d'étiquetage.

L'épaisseur d'une arête dépend linéairement du nombre d'images de documents de la classe source qui sont liées à la classe destination classe dans le graphe de voisinage.

ACID fournit également un outil d'interaction *zoom et pan* [4, 13] ainsi qu'un outil permettant de mettre en évidence le graphe de voisinage d'un nœud nœud donné (inspiré de [49]). En cliquant sur un nœud, nous construisons un graphe quotient supplémentaire en ne considérant que l'image focalisée et son voisinage et le dessinons comme une superposition du dessin actuel. Le panneau gauche du logiciel affiche des informations détaillées détaillées sur l'image focalisée.

Tous ces outils d'interaction générale sont utiles pour visualiser une grande base de données d'images étiquettées. Cependant, pour faciliter la correction de la classification, l'opérateur doit être guidé davantage. Nous proposons de concentrer son attention sur les erreurs potentielles d'étiquetage.

Suggestion de correction d'erreurs

Pour une classe donnée (métanœud), nous calculons une liste d'instances (nœuds) qui pourraient être déplacées dans une autre classe. L'idée principale est d'identifier les nœuds qui sont fortement connectés à un ensemble de nœuds appartenant à une autre classe. Nous vérifions indépendamment tous les déplacements possibles d'un document de la classe d'intérêt vers les classes adjacentes et conservons la modification

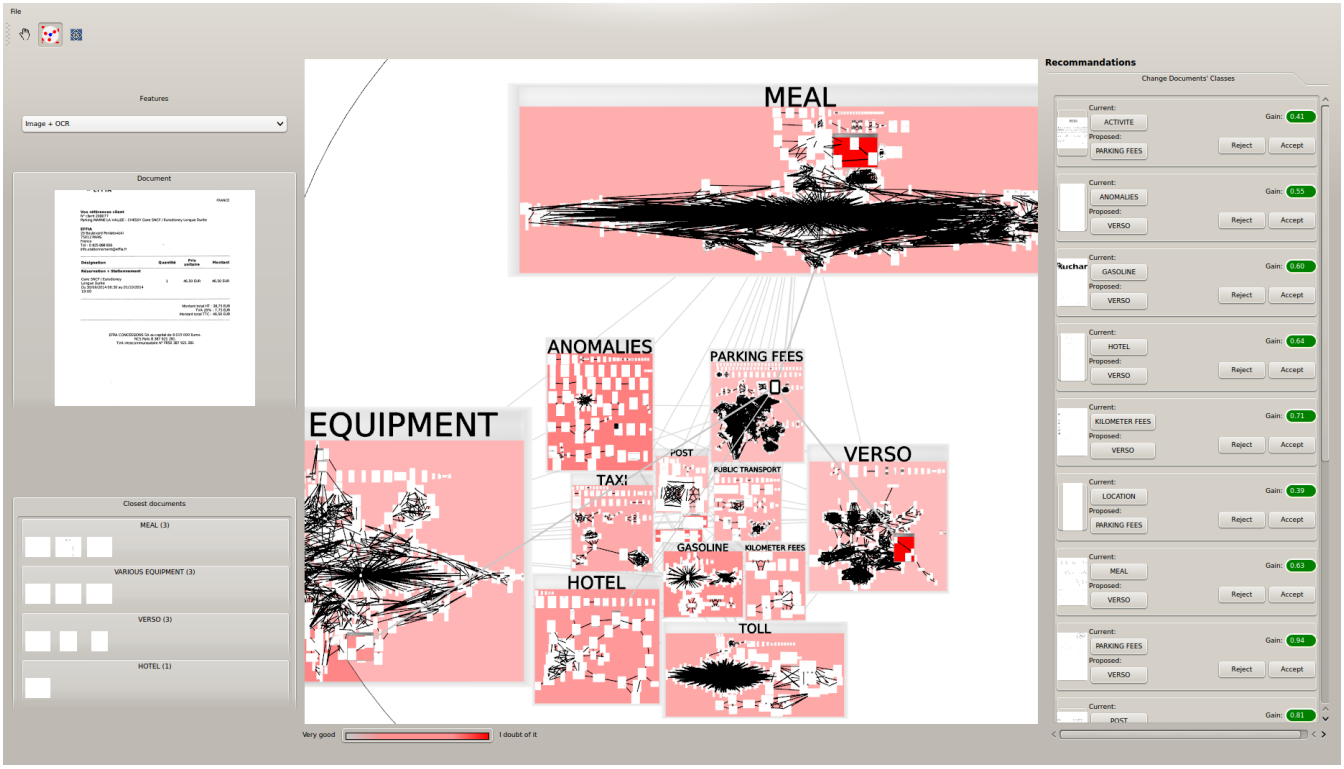


Figure 13.7. – Résultat visuel obtenu après la construction du graphe quotient. Le panneau central contient le graphe quotient. Les nœuds correspondent aux images et les arêtes relient les images similaires, les métanœuds contiennent tous les nœuds avec des étiquettes identiques et les méta-arêtes relient les métanœuds contenant des images liées. La couleur du métanœud indique sa cohésion interne. Par exemple, le métanœud rose clair « PARKING FEES » a une bonne cohésion (probablement aucune erreur de classification) alors que la couleur rouge du métanœud « ANOMALIES » montre qu'il pourrait contenir plusieurs erreurs d'étiquetage. Sur le panneau de droite, des suggestions de changements d'étiquettes sont proposées à l'utilisateur. Par exemple, ACID propose de déplacer une image du métanœud « ACTIVITIE » vers le métanœud « PARKING FEES ». ACID indique également que ce changement améliorera la cohésion du graphe global par un facteur de $0,41$.

qui fournit la meilleure valeur de qualité globale (QG) si nous l'appliquons.

L'interface utilisateur intègre un module de suggestion d'erreurs dans lequel les erreurs d'étiquetage potentielles et leurs corrections proposées sont listées. Il appartient à l'opérateur d'accepter ou de rejeter chaque proposition. Si la classification est effectivement erronée, il peut valider la proposition. Voir la figure 13.7, partie droite.

Résultats

L'évaluation est détaillée dans le papier correspondant [19]. On peut retenir que notre outil permet de ne parcourir que 20% à 30% de la base de données plutôt que 100% pour juger de la qualité de son étiquetage.

13.3. Visualisation pour la compréhension des réseaux profonds

13.3.1. Justification

Malgré son grand succès, l'AP souffre d'un problème important : la complexité de ces réseaux fait qu'il est difficile de comprendre comment ils prennent des décisions et pourquoi ils échouent. Ces dernières années, ce problème a donné naissance à l'IA Explicable (IAX) [28], ou « XAI pour Explainable Artificial Intelligence » dans la littérature anglophone. Elle a d'abord visé l'Apprentissage Automatique (AA) dans son ensemble, puis a abouti à l'essor de l'Apprentissage Profond Explicable (APX) [70], ou « XDL pour Explainable Deep Learning » dans la littérature anglophone. Dans le même temps, des solutions destinées à favoriser l'explicabilité et l'interprétabilité des systèmes à base d'AP ont été élaborées pour répondre

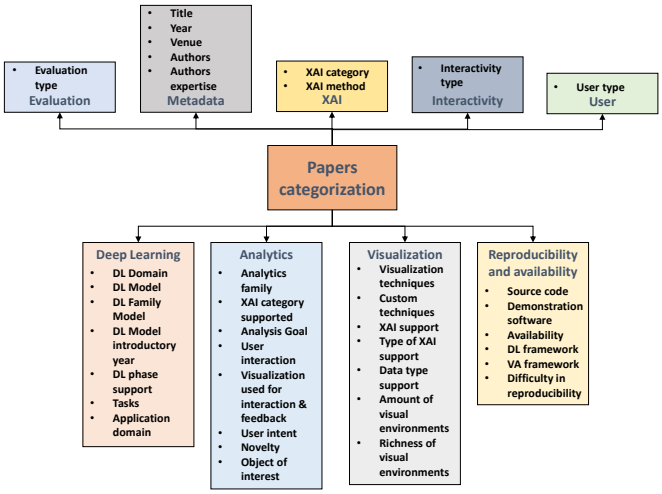


Figure 13.8. – Les 38 dimensions utilisées pour caractériser chaque article sont regroupées au sein de 9 groupes. Elles nous ont permis d'extraire correctement les informations d'intérêt pour chaque article et de les organiser.

aux besoins des différentes parties prenantes. Ces contributions ont permis d'identifier la visualisation et l'Analyse Visuelle et Interactive (AVI) comme des disciplines bien adaptées pour soutenir les chercheurs, les développeurs et les utilisateurs de solutions l'AP.

Avec une équipe de collègues de l'université de Rome et de la Northeastern University, nous avons catégorisé les travaux de la littérature liés à l'explicabilité des réseaux de neurones à l'aide méthodes interactives [41]. L'ensemble des dimensions utilisées pour effectuer la catégorisation des travaux est disponible dans la figure 13.8. Nous nous sommes intéressés aux méthodes *posthoc* (*c.-à-d.* après que le réseau ait effectué sa décision) plutôt qu'aux méthodes intrinsèques (le réseau fournit l'explication en même temps que la décision).

Plusieurs informations peuvent être collectées depuis un système d'AP (voir la figure 13.9). On peut lister les données brutes et les étiquettes issues des bases de données, ainsi que la sortie du Réseau de Neurones Profond (RNP) ou ses poids. Mais on peut également collecter des informations intermédiaires générées par le réseau lors de l'inférence telles que les cartes d'activations ou les cartes d'attentions. On peut en collecter d'autres lors de l'apprentissage comme les gradients ou les différentes fonctions de coût. Lors de l'évaluation, les métriques d'évaluation constituent également des données disponibles.

Notre analyse a montré que ces différentes données peuvent être utilisées pour générer des explications réparties dans 4 familles majeures (voir la figure 13.10) déjà présentées en section 4.1 : l'attribution de caractéristiques (*p. ex.* nos travaux du chapitre 4), l'explication par exemple, la présentation d'exemples contre-factuel, la détection des caractéristiques apprises et le comportement du modèle. C'est cette dernière catégorie qui est résoluble quasiment uniquement avec des approches AVI et qui nous intéresse dans ce chapitre.

L'analyse de la littérature [41] a également montré que la plupart des applications interactives ne ciblent pas un domaine métier spécifique et traitent majoritairement les modèles utilisés en vision. La plupart des travaux ciblent des utilisateurs qui ont une expertise en AP, mais utilisent des architectures existantes. Un peu moins concernent les architectes et peu concernent des non-experts en AP.

Le chapitre 4 a présenté la méthode d'explication Feature Explanation Method (FEM), ainsi que son évolution Multi-Layer Feature Explanation Method (MLFEM). Il s'agit d'approches orientées instance qui ne reposent pas sur des aspects visuels importants (mis à part l'utilisation d'une carte de chaleur pour

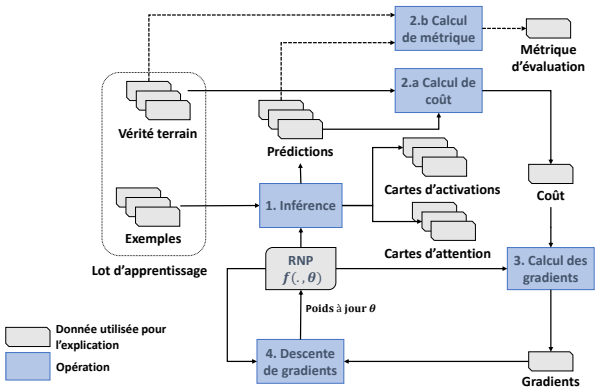


Figure 13.9. – Illustration de l'inférence et de la descente de gradient sur un lot pour un modèle d'AP. Le processus génère plusieurs données : prédictions, cartes d'activations, perte, métriques et gradients. Chacune d'entre elles peut être utilisée pour calculer des explications.

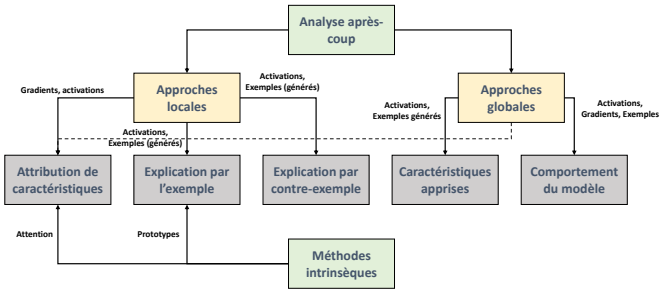


Figure 13.10. – Catégorisation des méthodes d'explication. Nous pouvons faire la distinction entre les méthodes qui utilisent des moyens externes (post-hoc) et les méthodes qui utilisent les internes du modèle (intrinsèques), et entre les méthodes qui fournissent des explications uniquement pour l'entrée actuelle (locales) et celles pour le comportement général du modèle (globales).

leur représentation) et ne sont donc pas traités dans cette partie. À la place, nous présentons une des méthodes visuelles pour l'explication que nous avons développées ; elles sont toutes basées sur des approches orientées jeu de données.

13.3.2. Représentation des flux de données dans un réseau

Cette section présente une application interactive qui aide à comprendre le flux de traitement des données dans un RNP. Il s'agit d'une approche globale de type comportement de modèle [32] qui peut aider à répondre aux questions suivantes :

- Q1 : Quelle est la performance du modèle ?
- Q2 : Devons-nous ajouter ou supprimer des couches dans le modèle ?
- Q3 : Devons-nous changer le nombre de filtres de cette couche convolutive ?
- Q4 : Notre ensemble de données d'entraînement est-il de qualité suffisante pour entraîner notre modèle ?

En comparaison aux méthodes de l'état de l'art, son originalité repose sur :

- sa polyvalence, car l'approche pas spécifique à la nature des données traitées par le modèle (*p. ex.* il n'existe aucune contrainte liée à l'utilisation d'images ou de mots en entrée du réseau à expliquer) ;
- son indépendance vis-à-vis de l'architecture du modèle et des connexions internes des neurones ;
- sa compatibilité avec tout type d'information qui est calculable au niveau de la couche du modèle, telle que l'activation des neurones, les gradients, les méthodes d'analyse (comme l'analyse de contribution) ou la saillance.

Généralités sur l'approche

Cette section présente l'ensemble des défis et des exigences identifiées afin de répondre aux questions Q1 à Q4. La figure 13.11 page suivante décrit le processus complet, allant de la collecte de données liées à l'évaluation d'un modèle jusqu'à sa visualisation.

Un réseau est composé de couches connectées qui agissent comme une succession de transformations de données de l'entrée vers la sortie. Les valeurs intermédiaires produites par chaque couche sont appelées « cartes d'activations » et sont généralement vues comme des matrices multidimensionnelles. Il convient de noter que les cartes d'activations sont des tenseurs avec de nombreux axes en hautes dimensions (*p. ex.* les cartes d'activations de VGG16 atteignent des tailles de 28 × 28 × 256 sur le jeu de données MNIST constitué de petites images en niveau de gris). Elles peuvent être techniquement difficiles à manipuler par un ordinateur classique lorsqu'elles sont nombreuses (R1) : la méthode doit s'adapter à des gigaoctets de données.

La méthode d'analyse choisie utilise l'ensemble des cartes d'activations calculées pour chaque échantillon d'entrée d'un jeu de données. De plus, pour comparer les bonnes prédictions de modèle aux mauvaises (R2) : chaque carte d'activations doit être collectée pour effectuer une analyse de données multidimensionnelle en fonction de sa vérité terrain et prédiction finale.

La méthode de visualisation doit être robuste pour répondre aux questions Q1 à Q4. Q1 est généralement exprimée par une métrique telle que le taux de reconnaissance et/ou la fonction de coût. Cette métrique est facilement interprétable par l'utilisateur, et (R3) : les performances de reconnaissance doivent être clairement identifiables sur la visualisation.

Les performances de classification d'un modèle sont généralement liées à la qualité de l'extraction des caractéristiques à travers ses couches : les premières couches détectent celles de bas niveau (*p. ex.* les formes dans les images) et les couches suivantes établissent des corrélations entre elles. Pour aider à répondre aux questions Q2 à Q4 et en suivant ce concept, (R4) le processus de raffinement doit être perceptible sur la visualisation.

Afin d'aider au débogage du modèle, la méthode donne des réponses plus précises aux questions Q2 et Q3 en présentant comment le processus d'extraction des caractéristiques évolue à travers le réseau. La comparaison des performances d'extraction de caractéristiques de chaque couche avec les autres permet de savoir où le modèle doit être amélioré. Cette comparaison doit être encodée dans la visualisation (R5) : la contribution de chaque couche au processus de classification doit être comparable à celle des autres.

Pour répondre à R1 et R2, il est possible de tirer parti d'une infrastructure de calcul distribué. Les systèmes modernes sont capables de traiter des pétaoctets de données à un coût raisonnable, ce qui est approprié pour analyser des réseaux complexes, puisque la charge de calcul peut être répartie étant donné que l'évaluation d'un seul exemple ne nécessite pas d'informations provenant des autres.

Le traitement de grandes quantités de données en hautes dimensions implique une bonne « scalabilité » de perception : dans le cas de R2, il est fréquent d'effectuer une réduction de dimensions (et de visualiser) en utilisant diverses méthodes linéaires (*p. ex.* PCA) et non linéaires (*p. ex.* t-SNE) de projection de données. Pour interpréter efficacement les résultats de la méthode d'analyse, des partitions sont calculées, pour chaque couche L^I, comme suit : étant donné L^I_x la carte d'activations calculée par L^I pour les données d'entrée x et f la fonction d'analyse choisie, les exemples a et b sont dans la même partition si et seulement si $f\left(L^I_a\right) \approx f\left(L^I_b\right)$. f peut être

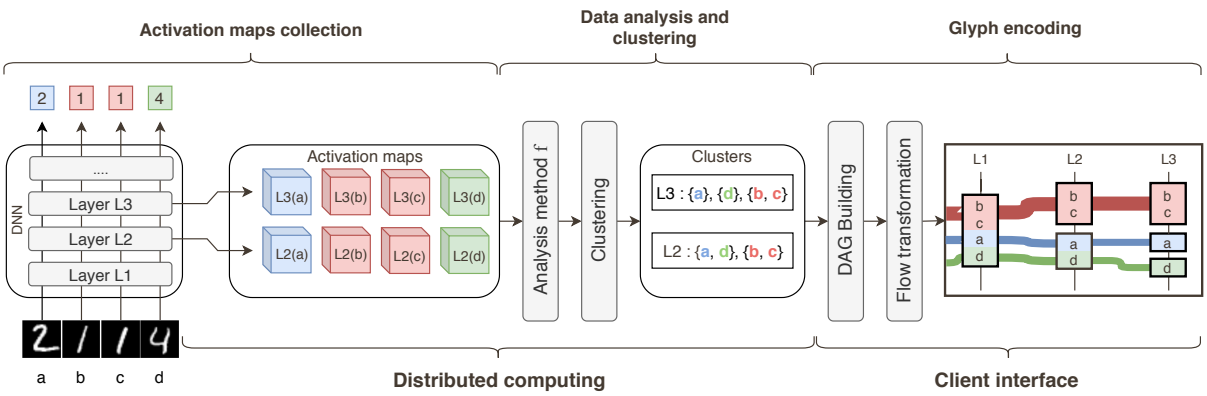


Figure 13.11. – Workflow de la méthode de visualisation d'activations par flux.

la fonction d'identité si nous voulons manipuler directement les données d'activation.

R3 est résolu en évaluant la composition des partitions calculées sur la dernière couche du modèle : une classification parfaite implique que chaque élément d'une même partition a la même vérité terrain que les autres, alors qu'une mauvaise classification conduira à obtenir des partitions composées d'exemples ayant une forte hétérogénéité de vérité terrain.

En suivant l'approche du partitionnement, la réponse à R4 a déjà été étudiée précédemment pour représenter une histoire [62], ou pour visualiser l'évolution de composants logiciels [63]. Les choix de conception sont principalement basés sur ces résultats précédents. Les clusters de chaque couche sont représentés à l'aide un diagramme de confinement, composé de leurs exemples respectifs. Puisque la méthode considère une grande quantité d'exemples, il a été décidé de ne pas les représenter individuellement. À la place, la taille de la représentation de chaque partition est proportionnelle à sa cardinalité, puis segmentée proportionnellement à chaque vérité terrain des exemples concernés. Cette approche permet de répondre à R3 et R5, mais R4 est plus délicat à traiter. Pour tracer chaque exemple au cours du processus d'extraction des caractéristiques, la même métaphore qu'une storyline est utilisée : elle est principalement inspirée du diagramme de coordonnées parallèles qui consiste à dessiner des connexions entre les partitions des couches consécutives qui contiennent le même ensemble d'exemples.

Détails de l'approche

Le processus global peut être divisé en deux parties : une première procédure qui se concentre sur la collecte de données, l'application de la méthode d'analyse et de son partitionnement, tandis que la seconde génère la visualisation de ces données.

Choix d'implémentation. Chaque motif (*c.-à-d.* le résultat du calcul précédent pour un exemple donné dans une couche donnée) est représenté par un nœud d'un graphe, et sa mesure de similarité avec les autres comme ses arêtes pondérées. L'algorithme de partitionnement de Markov (ou MCL) [64] est appliqué au graphe pour générer des partitions de motifs similaires. Cet algorithme est efficace pour identifier des communautés denses de nœuds dans un graphe en simulant une marche aléatoire, influencée par le poids des bords, sur celui-ci.

Analyse de données et partitionnement. Comme mentionné précédemment, en adressant R1 et R2, il est nécessaire d'effectuer un calcul distribué pour analyser les données. La procédure effectue deux itérations : la première sur l'ensemble de données d'évaluation pour collecter les cartes d'activations calculées par le modèle et effectuer leur analyse, l'autre sur chaque couche pour préparer ces résultats dans des encodages spécifiques.

La première étape consiste à collecter les cartes d'activations de chaque couche du modèle pour l'ensemble du jeu de données. Cette collecte peut être effectuée de diverses manières, mais il est nécessaire de conserver le lien entre les cartes et leur couche de calcul (R1) .

La méthode d'analyse (*c.-à-d.* prétraitement des activations) choisie est appliquée. Cet algorithme va transformer les valeurs des cartes d'activations tout en conservant l'aspect multidimensionnel des matrices (calcul de LRP dans notre expérimentation). Nous appelons motif le résultat de la transformation d'une carte d'un exemple (R2).

Les prédictions du modèle sont collectées pour étiqueter les flux de données pendant l'encodage de la visualisation (R3).

Les motifs d'une même couche sont vectorisés pour faciliter leur comparaison. Ces vecteurs peuvent être comparés à l'aide d'une métrique de similarité afin d'établir une matrice

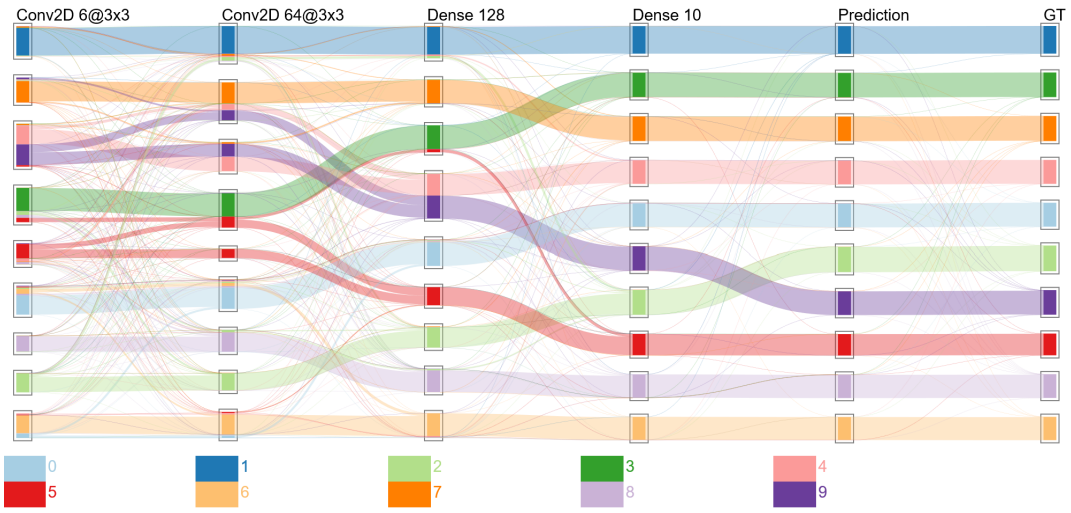


Figure 13.12. – Illustration de l'approche sur le réseau LeNet5 et la base de données MNIST. Le modèle est performant et convenablement dimensionné.

de similarité. La similarité Cosinus a été utilisée dans notre évaluation afin de mettre l'accent sur la diversité des modèles plutôt que la magnitude de la valeur des neurones calculée par la similarité euclidienne. La construction de ces matrices de similarité est une opération extrêmement coûteuse en raison du nombre de comparaisons à effectuer ($O\left(n^2\right)$) et de la taille des vecteurs comparés (R3, R4).

La matrice de similarité construite précédemment correspond également à une matrice d'adjacence d'un graphe d'arêtes pondérées. Cependant, ce graphe est très dense, car la similitude entre tous les éléments a été calculée. Pour obtenir un graphe moins dense et adéquat pour le partitionnement, on génère un graphe en connectant chaque nœud à ses k plus proches voisins. Le facteur k a été empiriquement fixé à D/N (D le nombre de données, N le nombre de classes) (R3, R4).

MCL est appliqué sur chaque graphe filtré afin de collecter leurs partitions de motifs. Chaque partition représente des exemples fortement similaires sur la couche donnée (R3, R4, R5).

Génération de la visualisation. La procédure de rendu est moins coûteuse que la précédente, principalement grâce à la taille des données à traiter, et peut être réalisée sur un seul ordinateur classique.

Pour afficher l'ensemble des prédictions globales par rapport aux vérités terrain (*c.-à-d.* simuler une matrice de confusion) de la même manière que les contributions des couches à la classification, deux groupes de clusters supplémentaires sont créés en utilisant la prédiction du modèle pour l'un et la vérité terrain pour l'autre (R3).

Comme pour la visualisation par coordonnées parallèles, cette visualisation peut être représentée par un graphe acyclique dirigé, ou DAG. Chaque partition est représentée par un sommet, et chaque arête signifie qu'il y a qu'il y a au moins un exemple dans les deux partitions concernées. Une contrainte est appliquée sur les coordonnées de chaque sommet (x, y). x est lié à la position de la couche du modèle concernée par la partition. y sera calculé de telle façon que les glyphes représentant les sommets ne se superposent pas (R3, R4).

La lisibilité du DAG peut être améliorée en réduisant le nombre de croisements d'arêtes en ordonnant chaque sommet selon sa coordonnée y. Ici, l'ordre barycentrique, décrit dans les travaux de Sugiyama [61], est appliqué principalement en raison de sa facilité de mise en œuvre et de son efficacité (R4).

Les flux (représentés par les arêtes entre les nœuds) doivent représenter une quantité de données (les exemples appartenant à la même partition) qui partagent des informations similaires. Ici, l'information affichée est la vérité terrain des échantillons concernés, ainsi que les partitions connectées (des partitions issues de deux couches successives du modèle). Les arêtes des exemples qui partagent la même vérité terrain et les mêmes partitions sont fusionnées. Le résultat de la fusion est un DAG avec les mêmes sommets, mais des arêtes pondérées reflétant la quantité de données d'une même vérité terrain dans l'intersection des partitions source et destination (R4, R5).

Les données sont encodées dans un graphe dans lequel chaque sommet décrit une partition, sa composition, et sa position dans la visualisation, et chaque arête décrit un flux, sa position de départ et destination, et sa composition, dans laquelle chaque exemple partage la même vérité terrain (R3, R4, R5).

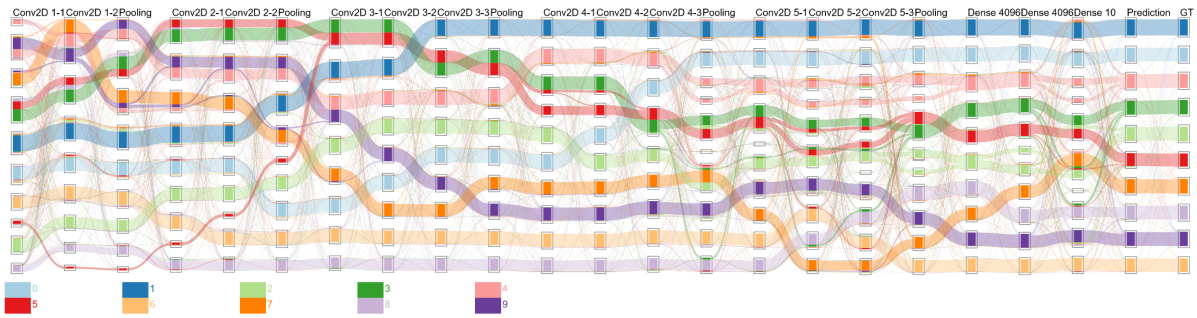


Figure 13.13. – Illustration de l'approche sur le réseau VGG16 et la base de données MNIST. Le modèle est performant et surdimensionné.

Dans la continuité de ce qui a été abordé pour R4 et R5, une couleur est attribuée à chaque vérité terrain. Les partitions sont représentées par des « boîtes » contenants des rectangles. Chaque rectangle représente une vérité terrain, est coloré en fonction de cette vérité terrain et sa hauteur dépend la proportion d'exemples appartenant à cette vérité terrain dans cette partition. Les flux sont représentés avec des Courbes de Bézier transparentes pour faciliter leur lecture. De plus, chaque axe est étiqueté par le type de couche responsable du regroupement, et une légende est en cours ajouté pour comprendre comment chaque classe de données est représentée.

Résultats obtenus

L'approche a été validée en utilisant plusieurs réseaux de neurones et jeux de données. Plus de détails sont présents dans le papier associé [32] ou le manuscrit de thèse d'Adrien Halnaut [31] Ce que l'on peut retenir de l'approche est qu'elle peut permettre (a) à des non-experts de comprendre comment les données sont perçues par le réseau tout au long de son exécution, et (b) à des experts d'identifier des zones superflues dans le réseau qui pourraient mériter d'être ablatées.

La figure 13.12 page précédente illustre la méthode sur le jeu de données MNIST appliqué au modèle LeNet5 (ce réseau obtient 98.5% de performance de reconnaissance sur le jeu de test). Peu de flux divergent entre la colonne `Prediction` (prédiction) et GT (vérité terrain) ce qui montre la bonne performance du modèle. On observe également la progression de la classification du réseau : quelques classes sont détectées rapidement (*p. ex.* 1 et 7), tandis que d'autres classes nécessitent de plus de traitements (*p. ex.* 0 et 6).

La figure 13.13 illustre la méthode sur le jeu de données MNIST appliqué au modèle VGG16 (ce réseau obtient 97,1% de performance de reconnaissance sur le jeu de test). Les classes sont principalement identifiées par la couche `Conv2D 3-2`. Cependant, la progression de classification stagne sur les convolutions suivantes, voire régresse aux couches de pooling. Nous attribuons ce comportement aux faits que le modèle est surdimensionné par rapport à la tâche à résoudre et mérite d'être élagué.

Chapitre 14.

Apprentissage automatique pour la visualisation d'information

Ce chapitre présente les travaux clés auxquels j'ai participé sur l'Apprentissage Automatique (AA) au service de la Visualisation d'Information (VI). Cette thématique de recherche a récemment émergé dans la communauté VI et tend à utiliser des techniques d'AA pour : automatiser certaines tâches en remplaçant des humains ou remplacer des algorithmes finement décrits par des humains par des méthodes automatiquement construites grâce à l'apprentissage.

La section 14.1 décrit la mise en place d'une méthodologie d'analyse automatique de technique de visualisation. Elle a été appliquée à l'évaluation de techniques de visualisation de graphes et à l'analyse de limite de perception.

Le section 14.2 montre comment utiliser un Réseau de Neurones Profond (RNP) sur graphe afin de calculer le plongement des nœuds d'un graphe en entrée pour le dessiner.

Le section 14.3 montre l'utilisation d'une fonction de coût optimisée par une simulation de descente de gradient afin de supprimer de façon efficace le recouvrement de glyphes dans un nuage de points.

14.1. Analyse automatique de techniques de visualisation

Il est souvent nécessaire de visualiser les jeux de données afin d'aider à appréhender leur contenu. Comme nous l'avons vu précédemment , utiliser uniquement des résumés statistiques peut amener à une mauvaise interprétation (pour rappel, voir la figure 7.2 page 44). Cependant, un même jeu de données peut être représenté par de nombreux encodages. Chaque encodage dispose de ses propres forces et faiblesses, ce qui le rend plus adapté à la réalisation de certaines tâches que d'autres. Ainsi, un utilisateur lambda réalisera plus facilement ou rapidement certaines tâches avec des encodages particuliers, et d'autres tâches avec d'autres encodages.

Pour réaliser des applications interactives pertinentes, il est nécessaire de connaitre la performance de réalisation d'une tâche en fonction de l'encodage pour choisir le plus approprié. Cette connaissance est acquise à l'aide d'une évaluation utilisateur [55] qui cherche à connaitre à quel point ils réalisent efficacement les tâches en fonction des encodages. Pour une tâche donnée, un nombre important de cobayes doit intervenir afin de réaliser la tâche sur différents encodages et différentes données : la performance de réalisation d'une tâche peut dépendre de l'encodage ou de propriétés de la donnée.

Le nombre de combinaisons (*c.-à-d.* donnée/encodage/tâches) possibles peut être trop important pour qu'un cobaye soit capable de toutes les tester. Il est nécessaire de faire des choix, pour sélectionner les combinaisons pertinentes : ce choix impacte nécessairement la qualité de l'évaluation. Il est généralement fait manuellement en s'aidant de résultats d'une pré-évaluation.

Afin de diminuer les biais humains, et d'améliorer la qualité de l'évaluation, nous proposons une méthode automatique qui fait office de pré-évaluation et permet de sélectionner les combinaisons intéressantes à proposer à des humains. L'approche est générique, mais la section 14.1.1 illustre cette approche sur des techniques de dessin de graphes [21] ; notre hypothèse étant que cette sélection est de meilleure qualité que ce que ferait un humain.

14.1.1. Illustration avec la comparaison de dessin de graphes

Principe de l'approche

Le framework que nous proposons est illustré en figure 14.1 page suivante. Il nécessite de choisir une tâche à résoudre et deux techniques de visualisation à comparer. Une base de données d'informations à visualiser associée à leur vérité terrain (*c.-à-d.* la réponse à la tâche) doit être constituée. Chaque technique de visualisation est utilisée pour représenter chaque

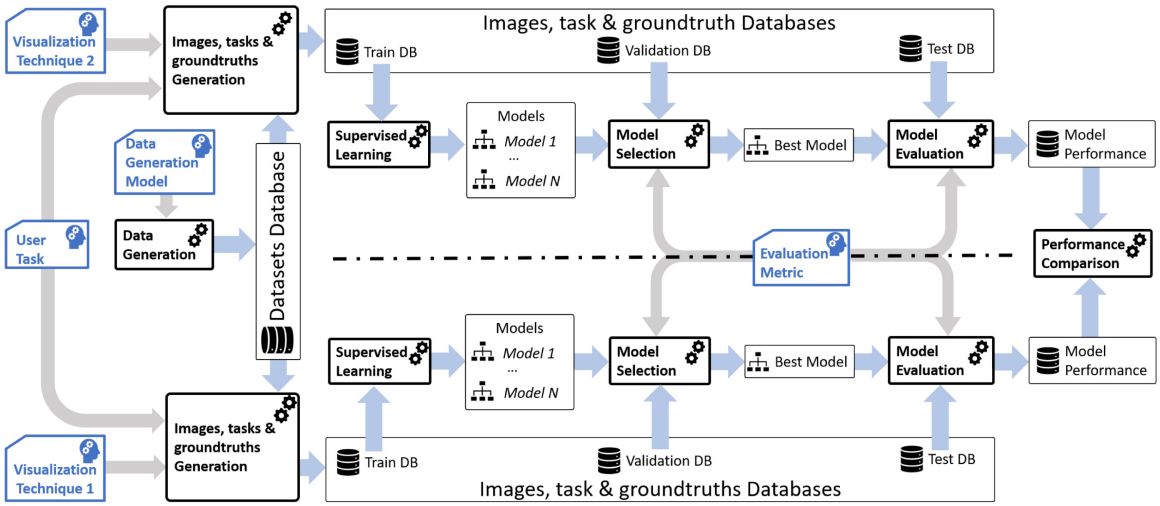


Figure 14.1. – Framework de comparaison de représentation visuelle.

élément de la base de données ; l'image générée est stockée. Les bases d'images des deux techniques sont partagées en ensembles d'apprentissage/validation/test. Pour chaque technique, différents modèles sont appris sur la base d'apprentissage, celui qui fait le moins d'erreurs sur la base de validation est retenu. La base de test est quant à elle utilisée pour comparer les performances des deux modèles sélectionnés sur leur visualisation. L'efficacité d'une représentation sur une tâche peut être mesurée grâce à la vitesse de convergence lors de l'apprentissage, et le nombre d'erreurs de reconnaissances sur la base de test.

Application à la représentation de graphes

Différents encodages peuvent être utilisés pour visualiser des graphes. Dans cette étude, nous nous intéressons à la matrice d'adjacence (voir la figure 14.2a page suivante) et la vue nœuds-liens (voir la figure 14.2b page ci-contre, et déjà abordé au chapitre 8). La matrice d'adjacence représente le graphe sous la forme d'une grille carrée dont la largeur correspond au nombre de nœuds et les cellules correspondent à la présence d'une arête entre le nœud représenté par la ligne et celui de la colonne. Il y a donc un double encodage pour chaque nœud. La vue nœuds-liens représente chaque nœud avec un glyphe, souvent un disque, et les arêtes sont dessinées en traçant une ligne entre entre le glyphe représentant le nœud source et celui représentant le nœud destination.

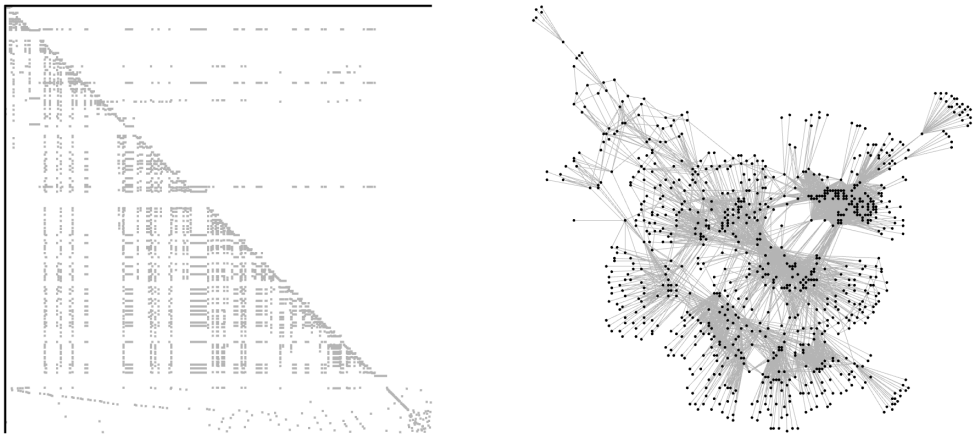
Il est connu que ces deux encodages n'ont pas les mêmes performances en fonction de la tâche cible, et nous vérifions dans ce travail que les résultats d'évaluation obtenus par la machine sont similaires à ceux obtenus par les humains [15, 51].

Résultats

Le papier associé à cette partie [21] présente en détail la génération du jeu de données et ses contraintes, la configuration des techniques de visualisation, ainsi que le protocole expérimental et les résultats. Nous avons testé, de façon automatique, 3 des tâches (bas niveau) de Ghoniem et al. [15] et 1 (haut niveau, tache 10 qui consiste à trouver la taille du plus court chemin entre deux nœuds) de Okoe *et al.* [51] qui cherchent à comparer les performances d'une vue nœuds-liens à celle d'une matrice d'adjacence. Nous pouvons trouver des résultats divergents dans les reproductions de ces études [21].

Les résultats que nous obtenons montrent les points suivants. La matrice de confusion est plus appropriée que la vue nœuds-liens pour les tâches suivantes : estimation du nombre de nœuds, estimation du nombre d'arêtes, estimation du degré maximum du graphe. Les modèles appris pour la matrice d'adjacence ont convergé plus rapidement que ceux appris pour la vue nœuds-liens : nous considérons que la tâche est plus facile avec la matrice d'adjacence.

Les résultats sont plus mitigés sur la détection du plus court chemin entre deux nœuds. La plupart des combinaisons architecture de réseau et technique de visualisation n'ont pas réussi à apprendre un modèle pour répondre au problème. Le couple LeNEt et vue nœuds-liens a cependant eu de bonnes performances de reconnaissance.



(a) Matrice d'adjacence

(b) Vue nœuds-liens

Figure 14.2. – Représentation du même graphe en utilisant deux techniques de représentation différentes : la matrice d'adjacence en 14.2a, et la vue nœuds-liens en 14.2b.

14.2. Dessin de graphes : plongement dans une vue nœuds-liens

14.2.1. Introduction

Le chapitre 8 a présenté quelques techniques de dessin de graphe en vue nœuds-liens. Ce qu'il faut retenir, c'est qu'une façon usuelle de calculer les coordonnées des nœuds d'un graphe est d'employer une méthode itérative de placement. Les nœuds sont déplacés jusqu'à satisfaire un critère à optimiser ; les techniques varient sur la stratégie à utiliser pour déplacer les nœuds ou la fonction d'optimisation. Le nombre d'opérations à réaliser est donc relativement important et dépend du nombre d'itérations nécessaire pour converger.

Les réseaux de neurones sont considérés comme étant des fonctions d'approximation universelle [7, 38] : on peut trouver des architectures qui permettent d'approximer n'importe quelle fonction (dans les faits, les théorèmes ne le prouvent que sur des fonctions continues, mais on prendra la liberté de considérer c'est également vrai dans notre cas). Il est donc intéressant de ce demander s'il est possible de trouver un réseau qui approxime une fonction de dessin graphe (*c.-à-d.* la sortie du réseau correspond aux coordonnées $\mathbb{R}^2$ des nœuds). De cette façon, au lieu d'utiliser une méthode itérative, et lente, de dessin de graphe, nous pouvons utiliser une méthode non itérative, et rapide. Le coût est déplacé au moment de l'apprentissage du réseau, plutôt qu'au moment du dessin du graphe.

Des approches orientées apprentissage supervisé existent déjà. À partir d'une base de données de graphes pour lesquels le plongement est connu, on sait faire apprendre au réseau à reproduire le plongement depuis la topologie du graphe [67]. De telles méthodes fonctionnent plutôt bien, mais restent relativement coûteuses : il faut calculer le plongement d'un nombre considérable de graphes pour construire la base d'apprentissage ; la fonction de coût peut également être calculatoirement complexe à calculer pour prendre en compte d'éventuelles rotations ou symétries dans le résultat. Le réseau se limite à reproduire ce qu'il connait et ne pourra pas produire de meilleurs résultats que ceux présents dans la base d'apprentissage. Des réseaux non supervisés ont également été utilisés dans le cas de dessin de graphes. Cependant, ils nécessitent un apprentissage spécifique pour chaque graphe à dessiner. GraphTSNE [43] utilise un réseau sur graphe pour faire le plongement d'un graphe, dont les nœuds disposent d'attributs, en 2D. La fonction de coût prend en compte à la fois la topologie et l'attribution des nœuds.

Nous avons fait le choix d'utiliser une approche non supervisée qui ne nécessite pas une optimisation spécifique pour chaque graphe à dessiner : il n'est pas nécessaire de disposer d'une base de graphes déjà dessinés, et le calcul du plongement est rapide, car il n'implique pas d'étapes d'optimisation.

14.2.2. Définition de réseaux sur graphe pour le dessin de graphe

L'approche retenue s'inspire des techniques de calcul de plongement multidimensionnel, ou « embedding » en anglais, qui calculent des représentations vectorielles de nœuds d'un graphe à partir de sa topologie et de ses attributs. Les détails et justifications sont présentés dans le papier correspondant [22].

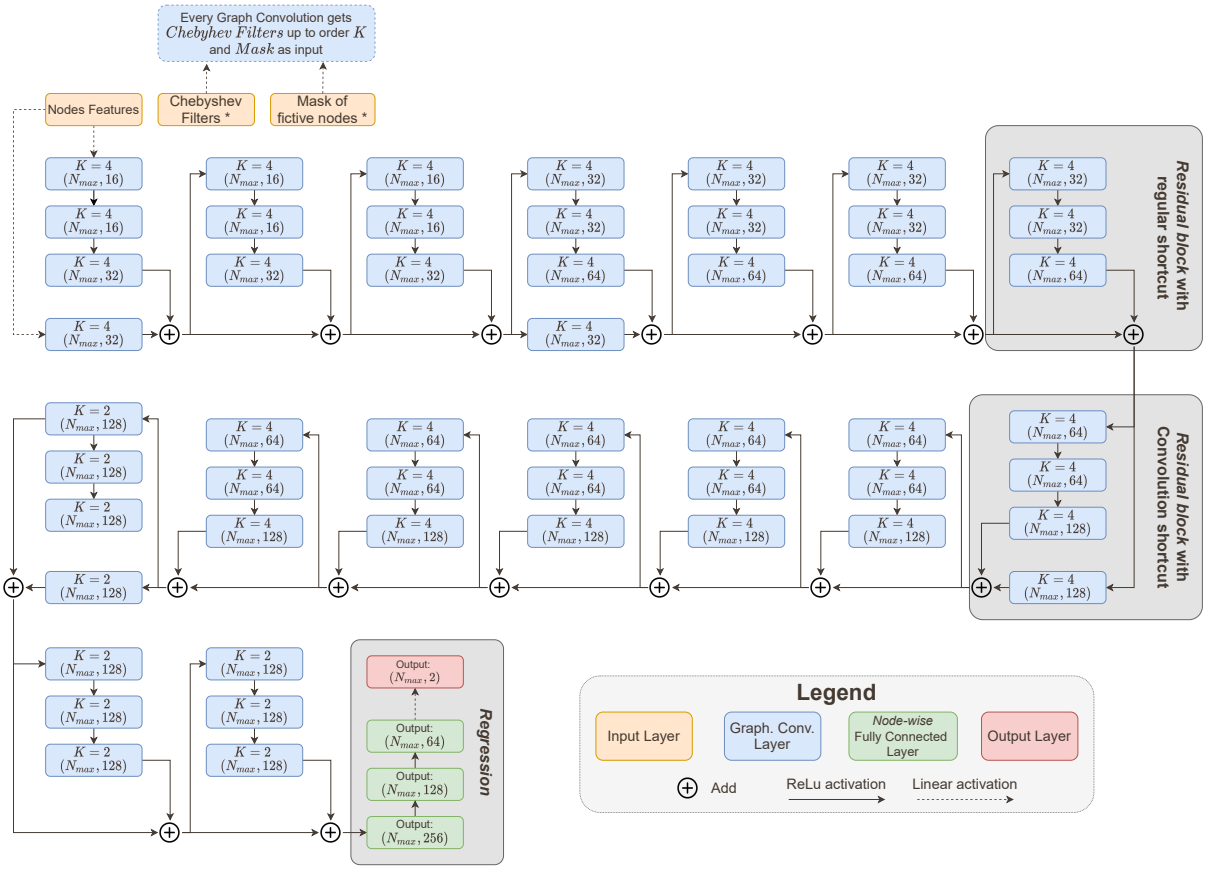


Figure 14.3. – Schéma de l'architecture de réseau choisie pour calculer le plongement d'un graphe.

Ici, nous nous reposons sur l'utilisation de réseaux de neurones sur graphes pour calculer de tels plongements en hautes dimensions, suivi d'un réseau multicouche qui réduit ce nombre de dimensions à 2 (*c.-à-d.* x et y) (voir la figure 14.3 pour la description de l'architecture). La fonction de coût est définie de telle façon à produire un plongement qui satisfait des critères esthétiques.

La partie du réseau qui se charge de collecter toutes les informations de voisinage, nécessaires au calcul d'un plongement pertinent, est basée sur l'architecture ResNet50 [37]. Ce réseau, initialement créé pour faire de la classification d'images, est composé de 16 blocs relativement similaires qui associent convolutions multiples sur le graphe et agrégation de l'entrée du bloc afin d'éviter des problèmes calculatoires lors de la descente de gradient. Les blocs ne possèdent pas d'opération de réduction spatiale de données (*c.-à-d.* « pooling » habituellement utilisé dans ce genre d'architecture). Plusieurs familles de méthodes permettent de manipuler des convolutions dans un graphe [69] : les méthodes spatiales à base de transfert de message, et les méthodes spectrales manipulant le Laplacien du graphe. Nous avons choisi d'implémenter une approche de la deuxième famille de méthodes qui repose sur l'utilisation de polynômes de Chebychev pour approximer les convolutions, et être calculatoirement plus efficace qu'un calcul exact [8]. Pour éviter que des nœuds avec une topologie de voisinage similaire soient plongés au même endroit, des attributs sont ajoutés aux nœuds : un identifiant (pour simuler l'ordre de parcours des nœuds d'un algorithme standard de dessin de graphe), et une valeur aléatoire (pour simuler des choix aléatoires comme dans la plupart des algorithmes de dessin de graphe par modèle de force). Une variante est également utilisée : elle ajoute aux attributs les coordonnées calculées par PivotMDS [5], le réseau cherchant à améliorer ce plongement initial.

Après le sous-réseau ResNet50, un réseau de 4 couches est utilisé pour calculer le plongement 2D des nœuds du graphe. La dernière a donc une sortie de taille 2. La représentation de chaque nœud du graphe est traitée de façon identique (*c.-à-d.* partage des poids) par cette partie de réduction de dimension.

L'approche étant non supervisée, la fonction de coût utilisée pour optimiser le réseau ne peut pas se baser sur une vérité terrain. À la place, elle utilise des informations liées à la topologie du réseau :

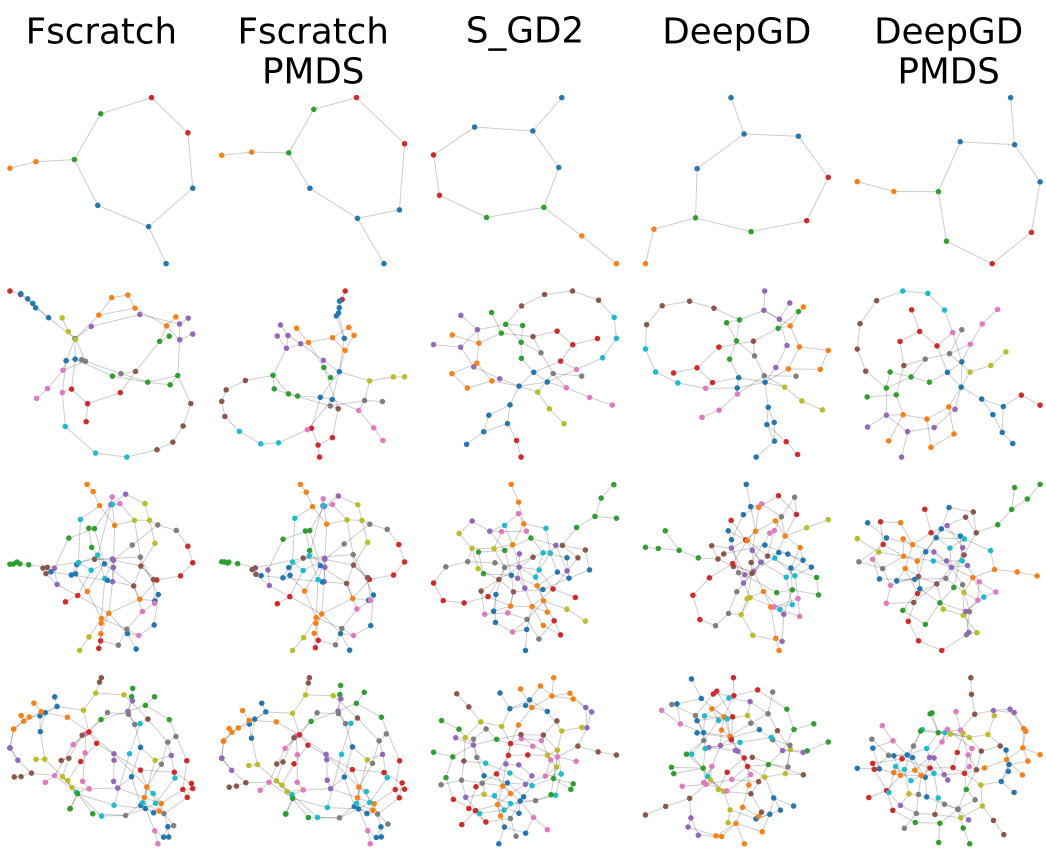


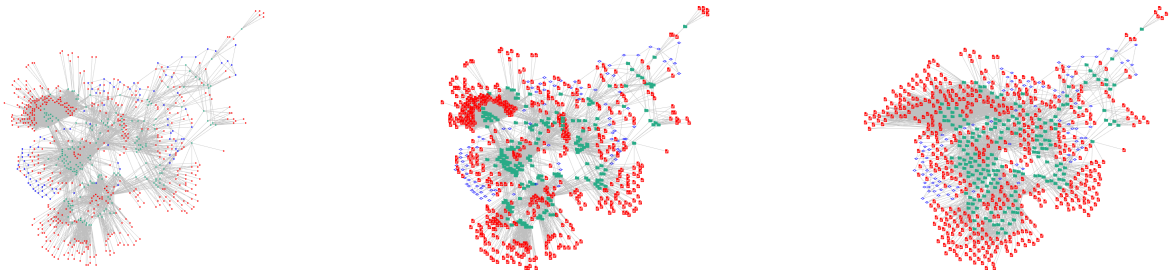
Figure 14.4. – Représentation de quelques graphes issus de la base Rome $N \approx \{10, 50, 75, 100\}$. Fscratch, Fscratch PMDS sont deux variantes de notre approche. Les autres sont des approches concurrentes.

$$C_{KL} = \sum_{i,j \in V, i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{14.1}$$

Cette fonction de coût correspond à la partie optimisation de la divergence de Kullback-Leibler de tsNet [40] qui cherche à rendre comparable la distribution des distances dans $\mathbb{R}$ des nœuds projetés (p_{ij}) par rapport à la distribution des distances dans le graphe (q_{ij}). Les composants de tsNet liés à la compression et la répulsion ne sont pas inclus, car nous avons expérimentalement observé qu'ils avaient un impact négatif. Nous n'utilisons pas non plus la fonction de stress.

14.2.3. Résultats

Les résultats détaillés de l'évaluation de l'approche sont détaillés dans l'article correspondant [22] et quelques images sont disponibles dans la figure 14.4. Ce que l'on peut retenir est que l'approche est pertinente, car elle permet d'obtenir très rapidement la projection 2D d'un graphe (au prix d'un apprentissage coûteux). La qualité de la base de données utilisée pour apprendre les poids du réseau est primordiale : moins de variabilité, ou un mauvais équilibrage des propriétés des graphes impliquent une moins bonne performance. Comme dans les autres domaines d'application, de meilleures performances sont obtenues en faisant un préapprentissage sur une base de données générale composée de graphes aléatoires, puis en la spécialisant sur une plus petite base de données spécifique (*p. ex.* des familles précises de graphes) pour améliorer les performances sur ce type de graphes. Ajouter aux nœuds des caractéristiques qui correspondent à un premier plongement effectué avec une méthode rapide, mais de qualité moyenne permet d'améliorer la qualité du plongement calculé par le réseau. Malgré l'utilisation de caractéristiques de position et d'aléatoire, les rendus peuvent souffrir de repliement lorsque le graphe est grand, et que la topologie des nœuds est similaire : on peut penser que le réseau n'est pas suffisamment profond pour identifier ces nœuds différents avec une topologie similaire.



(a) Plongement initial : une petite taille a été définie pour représenter les nœuds, mais ils sont considérés comme étant de taille nulle dans le calcul du plongement.

(b) Plongement initial : les nœuds sont affichés avec la bonne taille, il y a plusieurs chevauchements de glyphes les représentant.

(c) Application d'un algorithme (VPSC [11]) de suppression de chevauchements : les nœuds ne se superposent pas, mais la déformation de la représentation est fortement perceptible.

Figure 14.5. – Illustration du concept de chevauchements sur le graphe du dépôt git de ce manuscrit.

14.3. Suppression de chevauchements pour le dessin de graphes

14.3.1. Introduction

La section 14.2 présente une méthode de calcul de plongement des nœuds d'un graphe dans un espace 2D. Tout comme la plupart des méthodes de la littérature (bien qu'il existe des exceptions, *p. ex.* [35] ou les travaux en section 8.1.2), elle ne prend pas en compte la taille ou la forme des nœuds lors du calcul de la projection. La figure 14.5 illustre ce problème.

Il est fortement probable que la représentation du graphe contienne des chevauchements de nœuds qu'il faut supprimer avec un post-traitement de suppression de chevauchements (« Overlap Removal » en anglais) [6]. De telles méthodes utilisent la taille, forme et position des nœuds et les déplacent jusqu'à obtenir une représentation sans chevauchements. Elles cherchent communément à optimiser deux critères [6] : la compacité et la préservation du plongement initial. Le critère de compacité nécessite de créer une représentation sans chevauchements qui agrandit le moins possible la surface de dessin nécessaire (*c.-à-d.* qui affiche à l'écran les éléments avec une taille plus petite). En effet, une méthode efficace de suppression de chevauchements consiste à agrandir la zone de dessin en multipliant les coordonnées par un facteur commun à toutes les formes tel qu'il n'y ait plus de chevauchements (visuellement, cela revient à réduire la taille des nœuds tout en les affichant au même endroit dans l'espace-écran). Le critère de préservation du plongement initial cherche à ne pas déformer visuellement la représentation : les nœuds proches doivent rester proches, le voisinage doit être conservé, tout comme l'ordre en x et y. Différentes métriques, non listées pour des raisons évidentes de place, existent pour évaluer le respect de ces critères [6].

La suppression de chevauchements d'arêtes nécessite de détecter d'abord leur existence. Une recherche naïve se fait en $O(N^2)$ avec N nœuds, mais Dwyer et al. proposent une version de scan-line qui les détecte en $O(Nlog(N))$ [11]. Cette stratégie est souvent un élément clé dans les algorithmes de la littérature ; PFS [48], PFS' [36], FTA [39], RWordle-L [60] et uniform scaling [6] reposent sur lui. VPSC [11] et DIAMOND [47] utilisent un système de résolution de contrainte. PRISM [14] et GTREE [50] utilisent un système d'optimisation de stress (dans un graphe proxy créé à partir d'une triangulation de Delauney du graphe initial pour PRISME et un arbre couvrant minimal pour GTREE).

La méthode que nous proposons, FORBID [23], repose sur une approximation de descente de gradient, et ne traite que de nœuds rectangulaires de taille variable.

14.3.2. Présentation de Fast Overlap Removal By stochastic gradIent Descent for Graph Drawing

L'originalité de FORBID repose sur l'optimisation de deux critères : la recherche d'un facteur d'échelle optimal, ainsi que la quantité de déplacements minimale des nœuds pour un facteur donné. Le facteur d'échelle optimal s'obtient avec une recherche dichotomique, le facteur maximal correspondant à celui d'« uniform scaling ». Le déplacement des nœuds est lié à la définition d'une fonction de coût dédiée à la suppression de chevauchements : l'utilisation de la méthode d'optimisation SGD^2 [72] initialement développée pour faire du dessin de graphe. Le fonctionnement général de la méthode est présenté par la figure 14.6 page ci-contre ; deux variantes ont été testées : FORBID utilise le dernier « Agencement optimisé » à chaque nouvelle étape de « Mise à l'échelle du dessin », tandis que FORBID' utilise le plongement initial. Dans les deux variantes, le stress est modélisé de façon standard:

$$\sigma(X) = \sum_{i,j \in V} W_{ij} \left(||X_i - X_j|| - \delta_{ij} \right)^2 \tag{14.2}$$

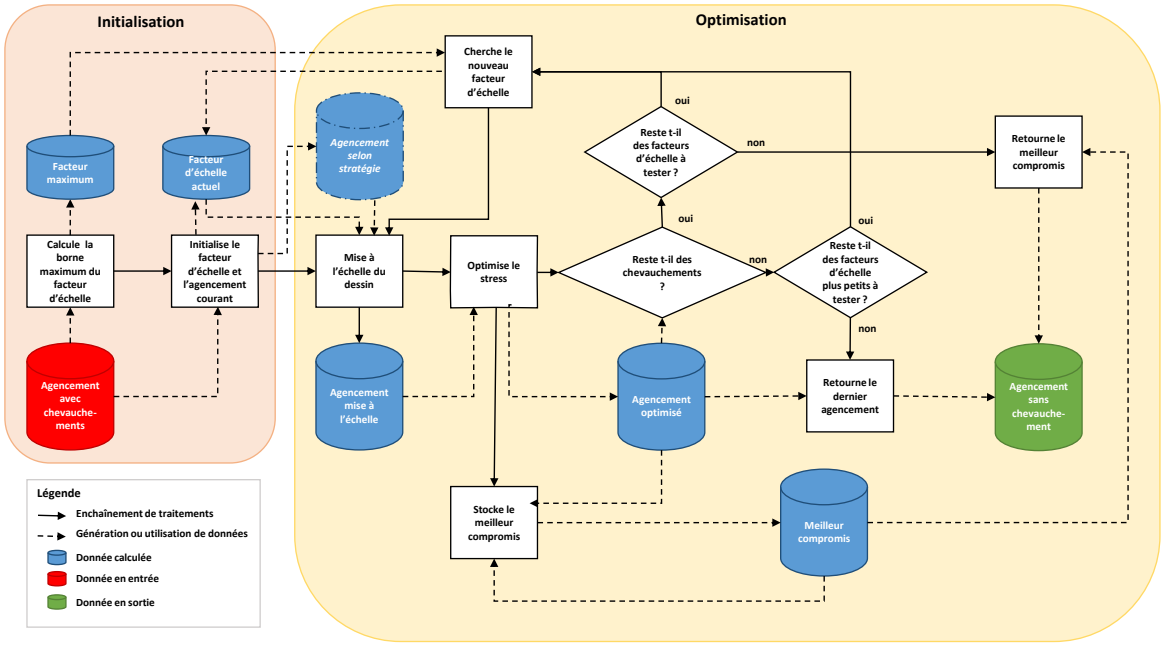


Figure 14.6. – Fonctionnement général du l'algorithme FORBID qui cherche à minimiser la quantité déplacements des nœuds et le changement d'échelle du dessin. La variante FORBID utilise le dernier « Agencement optimisé » à chaque nouvelle étape de « Mise à l'échelle du dessin », tandis que FORBID' utilise le plongement initial.

avec X_i le plongement du nœud i après transformation. Classiquement, δ_{ij} correspond à la distance des nœuds i et j dans l'espace initial, et W_{ij} est un facteur d'échelle souvent égal à δ_{ij}^{-1}. Nous nous servons de δ_{ij} pour encoder l'information indiquant que deux nœuds doivent bouger en cas de superposition, ou rester fixe s'il n'y en a pas. δ_{ij} est définie de telle façon à interdire les déplacements en l'absence de chevauchements ou le déplacement minimal s'il y en a un :

$$\delta_{ij} = \begin{cases} \sqrt{\left(\frac{larg_i + larg_j}{2}\right)^2 + \left(\frac{haut_i + haut_j}{2}\right)^2}, \text{si } (n_i, n_j) \in \mathbb{O} \\ ||X_i - X_j||, \text{sinon} \end{cases} \tag{14.3}$$

avec $larg_i$ et $haut_i$ respectivement la largeur et la hauteur du nœud i et $\mathbb{O}$ l'ensemble des paires de nœuds avec chevauchements. Le facteur de normalisation W_{ij} est défini de la façon suivante :

$$W_{ij} = \begin{cases} \delta_{ij}^{k\alpha}, \text{si } (n_i, n_j) \in \mathbb{O} \\ \delta_{ij}^{\alpha} \end{cases} \tag{14.4}$$

avec α généralement configuré à -2 et $k \in \mathbb{R}$ pour guider l'algorithme à converger dans une position qui ne détruit pas la forme générale de la projection cible.

Le nombre d'appels de la fonction d'optimisation présentée dans l'équation (14.2) page précédente dépend du nombre d'appels maximum de la fonction dichotomique de recherche du meilleur facteur d'échelle et est borné par $\log\left(\frac{\text{facteur maximal} - 1}{\text{précision du pas}}\right)$.

Résultats

FORBID a été évalué en suivant un protocole de l'état de l'art [6] et ses résultats sont décrits dans sa publication [23]. Ce que l'on peut retenir est que FORBID et sa variante FORBID' sont compétitifs ou meilleurs sur la majorité des métriques par rapport aux méthodes de l'état de l'art sur différentes catégories de graphes de complexité variable. Les temps d'exécution sont systématiquement supérieurs à ceux de PFS', mais restent meilleurs que ceux de GTREE et PRISM. La figure 14.7 page suivante présente visuellement les résultats obtenus sur 3 graphes de notre méthode et de celles de l'état de l'art. On peut constater que FORBID et FORBID' génèrent des plongements qui optimisent à la fois la compacité de la représentation et la préservation du plongement initial.

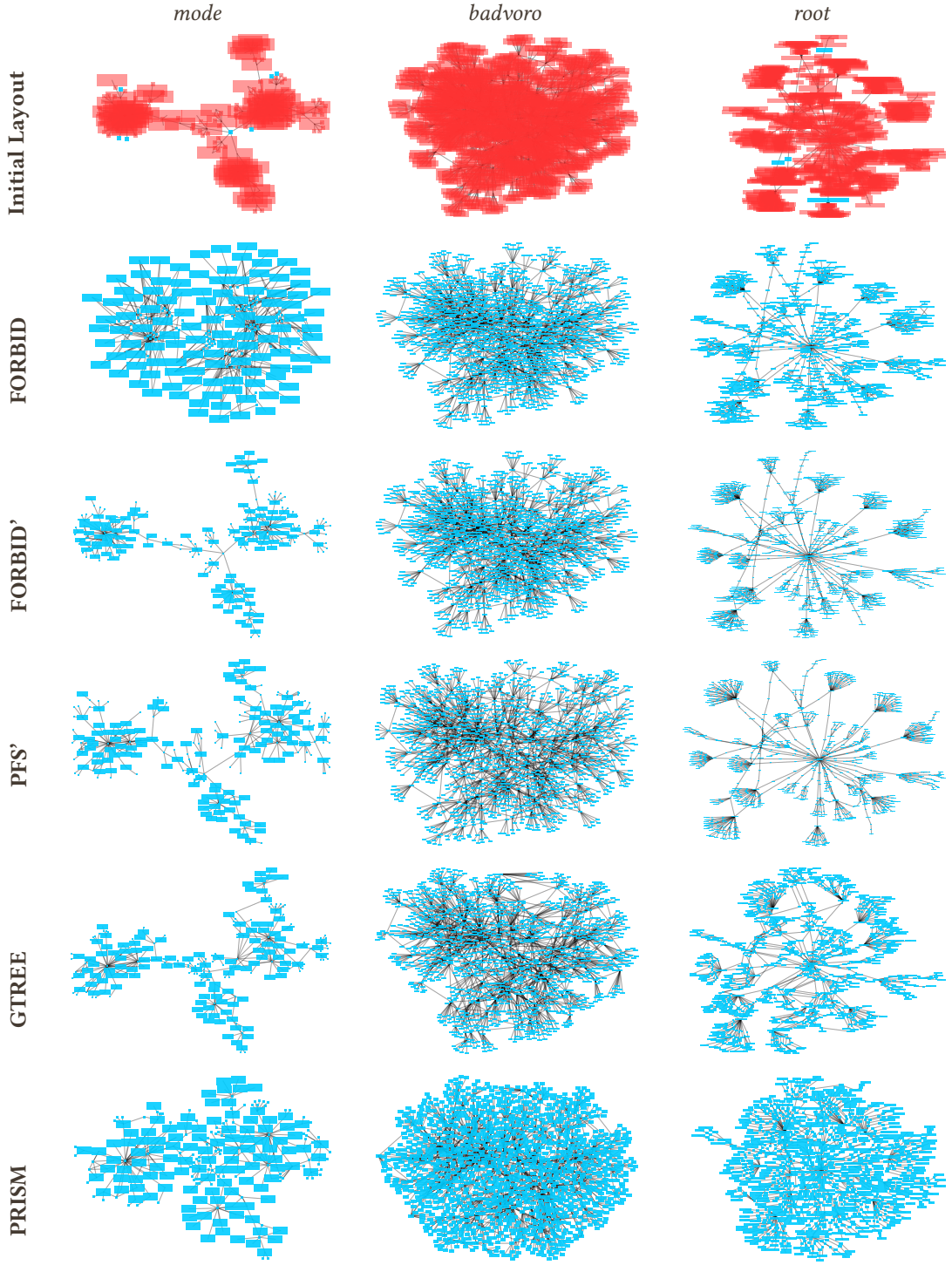


Figure 14.7. – Visualisation de graphes issus du jeu de données Graphviz avec une suppression des chevauchements calculées par FORBID, FORBID', PFS' GTREE et PRISM. Les nœuds sont colorés en rouge transparent si ils souffrent de chevauchements et bleu autrement.

Chapitre 15.

Discussion sur la symbiose de la visualisation d'information et l'apprentissage automatique

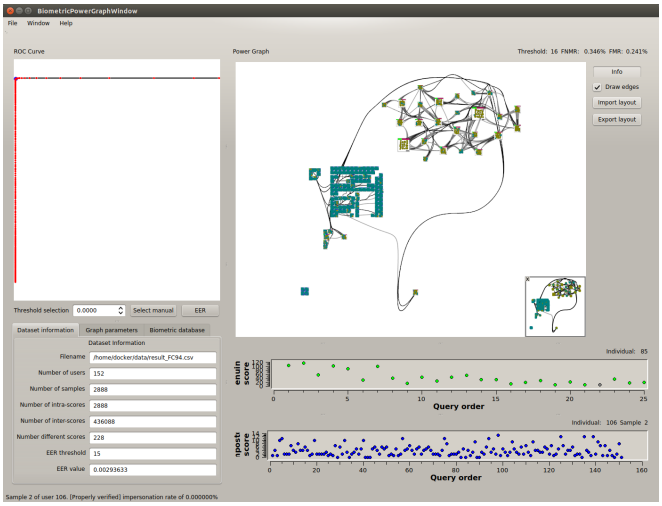


Figure 15.1. – Illustration de l'application d'Analyse Visuelle et Interactive (AVI) utilisant un PowerGraph pour évaluer un système biométrique.

Cette partie a présenté plusieurs travaux en Visualisation d'Information (VI) pour l'Apprentissage Automatique (AA), ainsi qu'en AA pour la VI. Certains d'entre eux sont toujours affectés de limitations qui mériteraient d'être traitées dans de prochaines études, et sont listées dans la suite de ce chapitre. D'autres n'ont pas été présentés pour des raisons évidentes de place, et sont rapidement décrits ici.

15.1. Visualisation pour l'apprentissage automatique

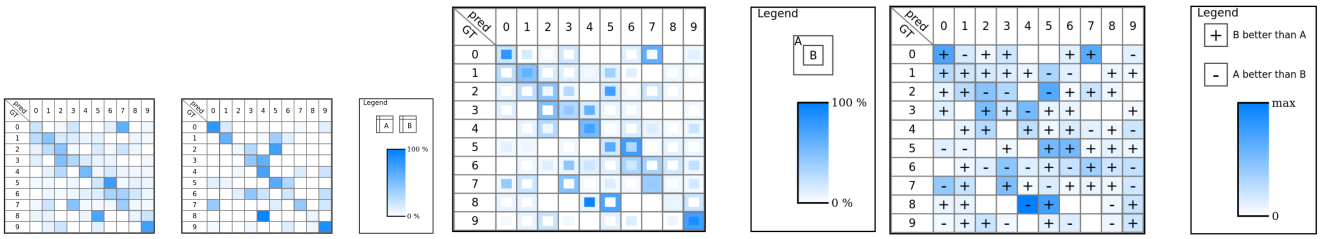
15.1.1. Application à l'authentification biométrique

L'utilisation de techniques de l'état de l'art en VI a permis d'apporter de nouvelles méthodes d'évaluation de systèmes biométrique. On peut regretter que les avoir publiées dans des conférences en VI n'a pas permis de les faire connaître à la communauté biométrique. Nous avons amélioré le Zoo Graph en utilisant un encodage différent, où les utilisateurs sont représentés dans un graphe avec leurs données biométriques en étant représentés sous la forme d'un PowerGraph [16] (voir la figure 15.1 et la section 8.3). La technique permet de représenter à la fois tous les utilisateurs, et tous les exemples de la base de test du système biométrique. L'utilisateur identifie rapidement les individus et les exemples problématiques du système. Ce travail et a été indirectement et partiellement présenté en section 8.3 sur la gestion du dessin des arêtes. Une évaluation à large échelle des méthodes proposées était prévue avec l'université ABC dans l'état de São Paulo/Brésil ; cependant, la contractualisation n'a pas été possible du fait de l'absence d'un service éthique et d'interlocuteurs privilégiés à l'université de Bordeaux à ce moment.

15.1.2. Application à l'apprentissage supervisé

ACID (voir la section 13.2) a cherché a estimer la qualité de l'étiquetage d'une base de données en effectuant, entre autres, un partitionnement. Cette approche montre ses limites lorsque la base de données à manipuler est trop grosse ; ce qui est le cas lors de l'apprentissage de modèles profonds. Des approches hiérarchiques semblent pertinentes pour passer à l'échelle [58].

Certains des travaux de thèse de Luc-Étienne Pommé (sous la direction de David Auber) auquel j'ai participé concernent l'évaluation de techniques de représentation pour comparer des matrices de confusion [54]. Elles sont souvent utilisées pour représenter les résultats d'un système de classification à plusieurs classes. Il n'existe cependant pas d'approche standard dans la littérature pour les comparer. Une première étude a permis d'identifier le meilleur encodage pour comparer deux



(a) Juxtaposition : affichage côte à côte des deux matrices

(b) Superposition : Combinaison des deux matrices [2]

(c) Encodage spécifique : Création d'une matrice relative

Figure 15.2. – Représentation des trois méthodes de comparaison de paires de matrices de confusion [54].

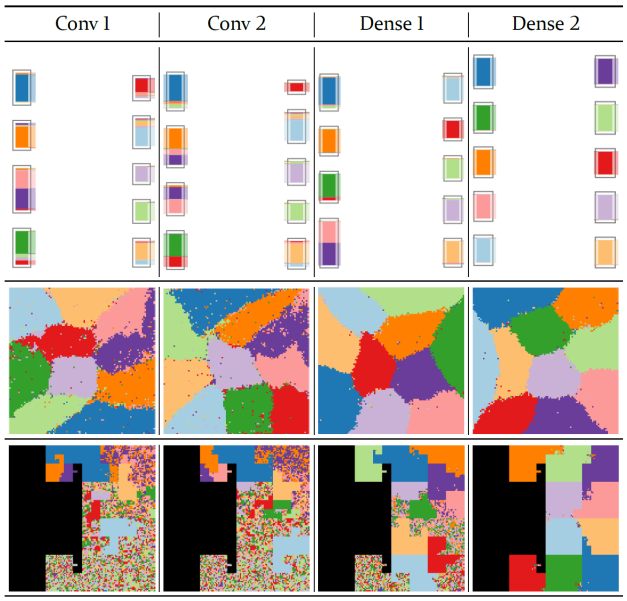


Figure 15.3. – Comparaison des encodages utilisés dans nos essais d'explication d'un couple données/réseau : les boites de confinement de la vue par flux, la représentation orientée pixels et les courbes fractales. Ici différentes couches de LeNet5 en utilisant la base MNIST sont utilisées pour illustrer les concepts [31].

systèmes en utilisant des techniques de juxtaposition, superposition et d'encodage [26] ; la figure 15.2 illustre ces trois approches. Une expérience utilisateur a montré que notre proposition (l'encodage spécifique) est plus performante que les autres, mais nécessite un apprentissage plus long. Nous envisageons de travailler sur son amélioration pour la simplifier afin de ce passer d'une période d'apprentissage pour savoir l'utiliser. Il est également intéressant de mettre au point de nouvelles stratégies pour passer à l'échelle en comparant plus de deux systèmes.

15.1.3. Application aux réseaux de neurones profonds

La représentation par flux (voir la section 13.3.2) pour expliquer un RNP et son jeu de données propose une vue globale qui ne peut pas passer l'échelle lorsque l'architecture du réseau n'est pas linéaire. En effet chaque flux n'a qu'une seule source et une seule destination. Malheureusement, la plupart des réseaux ne sont pas linéaires et proposent des « skip connection » (*p. ex.* ResNet, mais on peut s'en sortir en affichant que les fins de blocs) ou des branches (*p. ex.* les réseaux siamois, et là c'est plus compliqué à contourner) : certains flux peuvent avoir plusieurs sources ou destinations. Afin de prendre cet aspect en compte, nous avons proposé de nouveaux encodages basés sur des courbes fractales [33] ou des méthodes de représentation orientée pixel [34] qui permettent de s'affranchir de telles contraintes en n'affichant plus ces flux (voir figure 15.3 pour la différence de glyphe, et figure 15.4 page ci-contre pour la différence de systèmes). De futurs travaux les associeront à des approches locales, et des interactions dédiées afin de combiner le meilleur des deux approches.

L'évaluation de ces techniques de VI a montré qu'elles peuvent également être utilisées dans un contexte de simplification de réseau. Nous avons donc mis en place des vues spécifiques à cette problématique, et proposé un outil dédié à cet aspect [53] (figure 15.5 page suivante). L'évaluation a montré que les interactions permettent à l'utilisateur de mieux comprendre son modèle tout en le simplifiant ; les gains de réduction du nombre de paramètres peuvent atteindre 90% sur un modèle tel que VGG16 avec un impact minimal sur la réduction des performances de reconnaissance.

15.2. Apprentissage automatique pour la visualisation d'information

15.2.1. Analyse automatique de techniques de visualisation d'information

La méthodologie de travail utilisée pour évaluer les techniques de visualisation de graphes s'est avérée être pertinente et no-

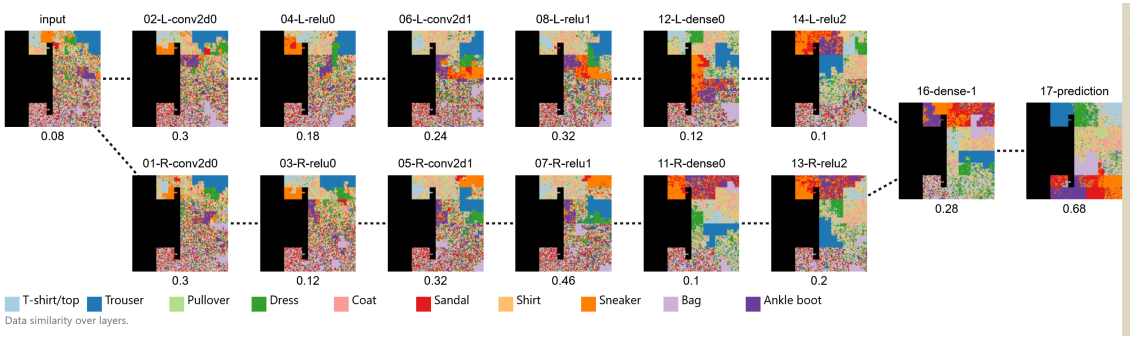


Figure 15.4. – Autre type de représentation permettant une visualisation sur des réseaux plus complexes.

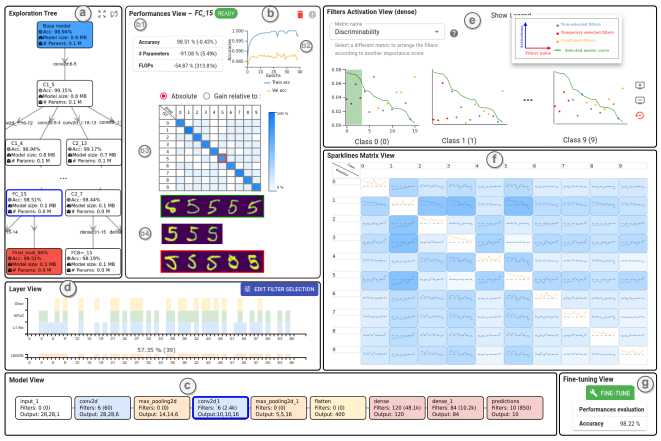


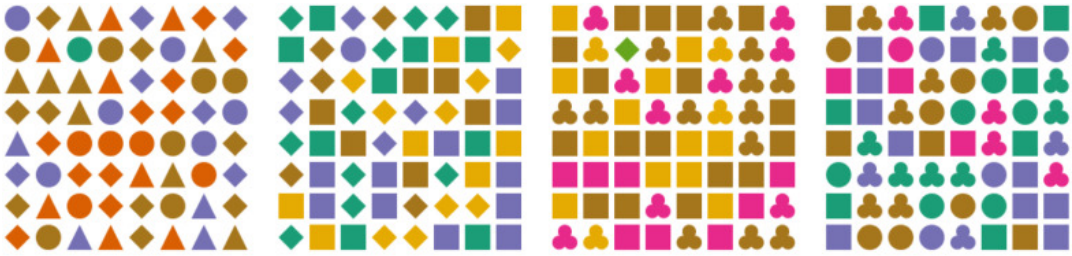
Figure 15.5. – Netprune : application de simplification de Réseau de Neurones Profond (RNP), reposant sur différentes représentations interactives dédiées.

vatrice. Depuis, on observe dans la littérature de plus en plus de travaux sur cette thématique (souvent appelée AI4VIS [68] ou ML4VIS[66]) ont été produit. Certains sont liés à la génération de dashboards [9], l'évaluation de techniques de visualisation [30], d'autres à la lecture d'information dans des visualisations [59].

Nous avons également adapté notre méthodologie de travail pour nous aider à établir une évaluation utilisateur sur les limites des perceptions visuelles d'un humain [20]. Plusieurs cobayes ont identifié des distracteurs dans une grille 8×8 de stimulus (voir figure 15.6 page suivante). Ces distracteurs pouvaient varier en fonction de leur forme, couleur ou combinaison de formes et couleurs. Ici les modèles ont permis d'identifier des conditions perçues de façon similaires afin de ne pas les proposer aux cobayes lors de l'évaluation et de grandement simplifier celle-ci. Les résultats montrent que la couleur est plus efficace que la forme pour encoder la donnée. Ils montrent également que les limites de capacités de perception des couleurs et formes ne sont pas aussi importantes que ce qui est communément admis.

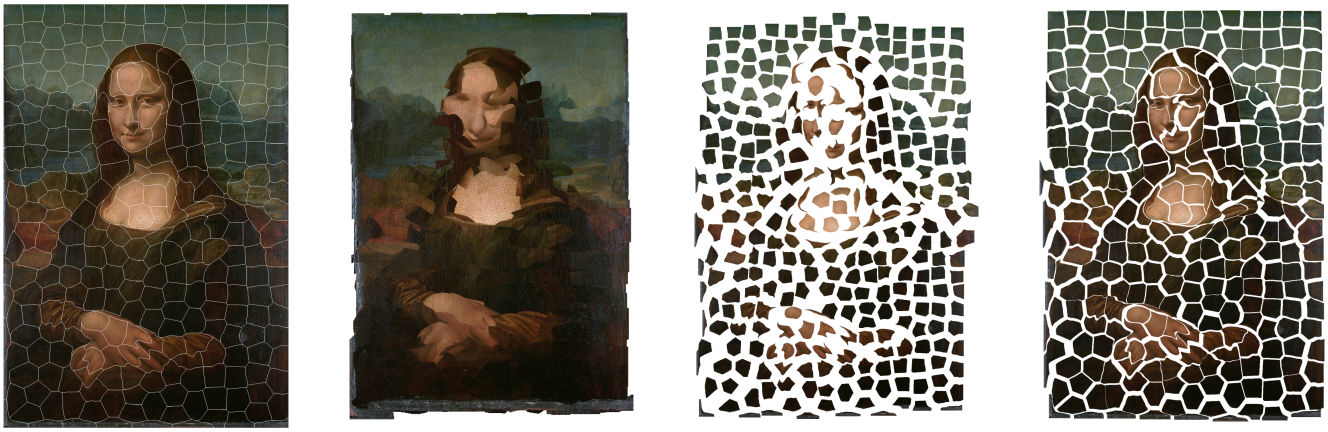
15.2.2. Dessin de graphes

Nous avons montré qu'il est pertinent d'utiliser un RNP, appris de façon non supervisée, pour calculer le plongement 2D de graphes. Cependant, il reste encore de nombreuses marges de progression pour améliorer de telles approches. En effet, il est relativement coûteux d'apprendre un tel réseau (même s'il n'est pas nécessaire de générer de vérité terrain) et les graphes manipulés restent de taille raisonnable. La qualité de la fonction de coût est primordial dans la qualité des résultats. Nous avons privilégié de tester des fonctions qui cherchent à préserver le voisinage avec des fonctions de coût proche de TSNE (*c.-à-d.* les voisins dans le graphe doivent être voisins dans l'espace 2D) plutôt que des fonctions qui cherchent à préserver la distance comme le stress (c'est à dire, des nœuds proches dans le graphe sont proches dans le dessin). Les définitions sont proches, mais les fonctions à optimiser différentes. Nous avons vu avec les techniques de suppression de chevauchements que le stress est une fonction de coût intéressante, c'est peut-être également le cas dans notre contexte de dessin de graphe. Cela dit, l'espace des fonctions de coût d'intérêt est relativement grand et nécessite d'être exploré. Je suis également convaincu que changer l'architecture du réseau pourrait avoir un impact non négligeable sur la qualité de la projection. Les transformeurs qui ont été créés initialement pour des séries temporelles [65] ont été adaptés avec succès dans le contexte du traitement d'images [44] et même des graphes [57]. De telles architectures pourraient aider à collecter des informations sur la topologie du graphe à une distance plus lointaine (voir entre tous les nœuds, deux à deux, dans une approche naïve et coûteuse), et participer efficacement à la réduction des repliements qui apparaissent dans les grandes grilles. Utiliser le système d'encodage des index est également une possibilité intéressante pour représenter l'identifiant des nœuds ou les nombres aléatoires. La génération automatique de caractéristiques pertinente, en amont du RNP, est également une piste intéressante. Il est possible de s'inspirer de méthodes



(a) *color* type; 3 shapes; 4 colors (b) *shape* type; 3 shapes; 4 colors (c) *red.* type; 3 shapes; 4 colors (d) *conj.* type; 3 shapes; 4 colors

Figure 15.6. – Illustration des différentes familles de distracteurs étudiés dans l'étude de perception visuelle [20].



(a) Génération artificielle de polygones représentant des nœuds.

(b) Altération pour créer du chevauchement.

(c) Suppression de chevauchements avec une méthode utilisant la boite englobante des nœuds (FORBID).

(d) Suppression de chevauchements avec une méthode utilisant les polygones concaves des nœuds (SORDID).

Figure 15.7. – Cette image illustre l'intérêt d'utiliser une méthode de suppression du chevauchement qui n'a pas d'aprioris sur la forme des nœuds à déplacer. (a) La Joconde découpée en super-pixels non convexes avec SLIC [1]. (b) Version altérée de (a) où la taille de chaque polygone est doublée. (c) Résultat de FORBID (qui considère des glyphes rectangulaires). (d) Résultat de SORDID (qui prend en compte la forme des polygones).

de création de caractéristiques à l'aide de pivots comme dans PivotMDS [5] ou de marche aléatoire comme node2vec [27].

15.2.3. Suppression de chevauchements

FORBID permet de gérer de façon efficace la suppression de chevauchements sur un graphe où les nœuds sont rectangulaires. On peut naturellement l'utiliser sur un graphe où les nœuds ont des formes plus complexes en utilisant leur boite englobante pour les représenter ; cependant beaucoup de place peut être perdue. Nous avons proposé SORDID [25], une version dépendante de la forme du glyphe : elle prend en compte la forme du glyphe décrite à l'aide d'un polygone. La méthode est capable de diminuer l'espace-écran utilisé, ainsi que la déformation du plongement initial en permettant à des formes concaves de s'imbriquer sans souffrir de chevauchement. Une illustration est fournie par la figure 15.7.

L'utilisation de la descente de gradient et de fonction de coût spécifique a également été généralisée dans GIST [24] afin de faire de la suppression de chevauchements dans le cadre plus général des nuages de points. Ici, nous travaillons également dans l'espace de représentation, plutôt que l'espace géométrique comme FORBID, afin de modéliser des tolérances de superposition en quantité de pixels dans la représentation. Il est ainsi possible de générer des représentations plus compactes pour lesquelles les chevauchements peuvent être à peine visibles.

Chapitre 16.

Bibliographie de la symbiose de la visualisation et de l'apprentissage automatique

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua et S. Süsstrunk. « SLIC superpixels compared to state-of-the-art superpixel methods ». Dans : *IEEE transactions on pattern analysis and machine intelligence* 34.11 (2012), p. 2274-2282 (cf. p. 96).

[2] B. Alper, B. Bach, N. Henry Riche, T. Isenberg et J.-D. Fekete. « Weighted graph comparison techniques for brain connectivity analysis ». Dans : *Proceedings of the SIGCHI conference on human factors in computing systems*. 2013, p. 483-492 (cf. p. 94).

[3] D. Archambault, T. Munzner et D. Auber. « Topolayout: Multilevel graph layout by topological features ». Dans : *IEEE transactions on visualization and computer graphics* 13.2 (2007), p. 305-317 (cf. p. 78).

[4] B. B. Bederson, L. Stead et J. D. Hollan. « Pad++: Advances in Multiscale Interfaces ». Dans : *Conference Companion on Human Factors in Computing Systems*. CHI '94. ACM, 1994, p. 315-316. isbn : 0-89791-651-4 (cf. p. 78).

[5] U. Brandes et C. Pich. « Eigensolver methods for progressive multidimensional scaling of large data ». Dans : *International Symposium on Graph Drawing*. Springer. 2006, p. 42-53 (cf. p. 88, 96).

[6] F. Chen, L. Piccinini, P. Poncelet et A. Sallaberry. « Node overlap removal algorithms: an extended comparative study ». Dans : *Journal of Graph Algorithms and Applications* (2020) (cf. p. 90, 91).

[7] G. Cybenko. « Approximation by superpositions of a sigmoidal function ». Dans : *Mathematics of control, signals and systems* 2.4 (1989), p. 303-314 (cf. p. 87).

[8] M. Defferrard, X. Bresson et P. Vandergheynst. « Convolutional neural networks on graphs with fast localized spectral filtering ». Dans : *Advances in neural information processing systems* 29 (2016) (cf. p. 88).

[9] D. Deng, A. Wu, H. Qu et Y. Wu. « Dashbot: Insight-driven dashboard generation based on deep reinforcement learning ». Dans : *IEEE Transactions on Visualization and Computer Graphics* 29.1 (2022), p. 690-700 (cf. p. 95).

[10] T. Dwyer, K. Marriott et P. Stuckey. « Fast Node Overlap Removal ». Dans : *Proc. Graph Drawing 2005 (GD'05)*. 2005, p. 153-164 (cf. p. 78).

[11] T. Dwyer, K. Marriott et P. J. Stuckey. « Fast node overlap removal ». Dans : *International Symposium on Graph Drawing*. Springer. 2005, p. 153-164 (cf. p. 90).

[12] T. Fawcett. « An introduction to ROC analysis ». Dans : *Pattern recognition letters* 27.8 (2006), p. 861-874 (cf. p. 73).

[13] G. W. Furnas et B. B. Bederson. « Space-scale Diagrams: Understanding Multiscale Interfaces ». Dans : *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '95. ACM Press/Addison-Wesley Publishing Co., 1995, p. 234-241. isbn : 0-201-84705-1 (cf. p. 78).

[14] E. Gansner et Y. Hu. « Efficient, proximity-preserving node overlap removal. » Dans : *Journal of Graph Algorithms and Applications* 14.1 (2010), p. 53-74 (cf. p. 90).

[15] M. Ghoniem, J.-D. Fekete et P. Castagliola. « On the readability of graphs using node-link and matrix-based representations: a controlled experiment and statistical analysis ». Dans : *Information visualization* 4.2 (2005), p. 114-135 (cf. p. 86).

[16] R. **Giot**. « Evaluation of Biometric Authentication Systems through Visualisation of Partitioned and Bundled Powergraphs ». Dans : *Information Visualisation 2017 (IV 2017)*. 2017, p. 104-110 (cf. p. 93).

[17] R. **Giot** et R. Bourqui. « Fast Graph Drawing Algorithm Revealing Networks Cores ». Dans : *Information Visualisation 2015 (IV 2015)*. Conference Publishing Services, 2015, p. 259-264 (cf. p. 78).

[18] R. **Giot**, R. Bourqui et M. El-Abed. « Zoo Graph: a New Visualisation for Biometric System Evaluation ». Dans : *Information Visualisation 2016 (IV 2016)*. 2016, p. 190-195 (cf. p. 74).

[19] R. **Giot**, R. Bourqui, N. Journet et A. Vialard. « Visual Graph Analysis for Quality Assessment of Manually Labelled Documents Image Database ». Dans : *13th International Conference on Document Analysis and Recognition (ICDAR 2015)*. IAPR. IEEE Computer Society, 2015, p. 1136-1140 (cf. p. 76-79).

[20] L. Giovannangeli, R. Bourqui, R. **Giot** et D. Auber. « Color and Shape efficiency for outlier detection from automated to user evaluations ». Dans : *Visual Informatics (VI)* (2022). issn : 2468-502X (cf. p. 95, 96).

[21] L. Giovannangeli, R. Bourqui, R. **Giot** et D. Auber. « Toward automatic comparison of visualization techniques: Application to graph visualization ». Dans : *Visual Informatics (VI)* 4.2 (2020), p. 86-98. issn : 2468-502X (cf. p. 85, 86).

[22] L. Giovannangeli, F. Lalanne, D. Auber, R. **Giot** et R. Bourqui. « Toward efficient Deep Learning for Graph Drawing (DL4GD) ». Dans : *IEEE Transaction on Visualization and Computer Graphics (TVCG)* (2022), p. 16 (cf. p. 87, 89).

[23] L. Giovannangeli, F. Lalanne, R. **Giot** et R. Bourqui. « FORBID: Fast Overlap Removal By Stochastic Gradient Descent for Graph Drawing ». Dans : *Proceeding of the 30th International Symposium on Graph Drawing and Network Visualization (GD 22)*. 2022 (cf. p. 90, 91).

[24] L. Giovannangeli, F. Lalanne, R. **Giot** et R. Bourqui. « Guaranteed Visibility in Scatterplots with Tolerance ». Dans : *Proceedings of IEEE VIS 2023*. 2023, p. 11 (cf. p. 96).

[25] L. Giovannangeli, F. Lalanne, R. **Giot** et R. Bourqui. « Overlap Removal by Stochastic Gradient Descent with(out) Shape Awareness ». Dans : *IEEE Transactions on Visualization and Computer Graphics (TVCG)* (2024) (cf. p. 96).

[26] M. Gleicher. « Considerations for visualizing comparison ». Dans : *IEEE transactions on visualization and computer graphics* 24.1 (2017), p. 413-423 (cf. p. 94).

[27] A. Grover et J. Leskovec. « node2vec: Scalable feature learning for networks ». Dans : *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. 2016, p. 855-864 (cf. p. 96).

[28] D. Gunning, M. Stefik, J. Choi, T. Miller, S. Stumpf et G.-Z. Yang. « XAI—Explainable artificial intelligence ». Dans : *Science Robotics* 4.37 (2019) (cf. p. 79).

[29] S. Hachul et M. Jünger. « Drawing large graphs with a potential-field-based multilevel algorithm ». Dans : *Graph Drawing*. 2005, p. 285-295 (cf. p. 78).

[30] H. Haleem, Y. Wang, A. Puri, S. Wadhwa et H. Qu. « Evaluating the readability of force directed graph layouts: A deep learning approach ». Dans : *IEEE computer graphics and applications* 39.4 (2019), p. 40-53 (cf. p. 95).

[31] A. Halnaut. « Méthodes et outils d'analyse visuelle pour la compréhension, l'optimisation et l'élaboration de modèles de réseaux de neurones profonds ». Thèse de doct. Université de Bordeaux, 2024. En prép. (cf. p. 84, 94).

[32] A. Halnaut, R. **Giot**, R. Bourqui et D. Auber. « Deep Dive into Deep Neural Networks with Flows ». Dans : *Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2020): IVAPP*. T. 3. 2020, p. 231-239 (cf. p. 81, 84).

[33] A. Halnaut, R. **Giot**, R. Bourqui et D. Auber. « Samples Classification Analysis Accross DNN Layers with Fractal Curves ». Dans : *Proceedings of the workshops of the 25th International Conference on Pattern Recognition (EDL-AI/ICPR 2020). Pattern Recognition. ICPR International Workshops and Challenges: Virtual Event, January 10–15, 2021, Proceedings, Part III. (ICPRW)*. Springer International Publishing, 2021, p. 47-61 (cf. p. 94).

[34] A. Halnaut, R. **Giot**, R. Bourqui et D. Auber. « VRGrid: Efficient Transformation of 2D Data into Pixel Grid Layouts ». Dans : *Proceedings of the 26th International Conference Information Visualisation (IV 2022)*. 2022 (cf. p. 94).

[35] D. Harel et Y. Koren. « Drawing graphs with non-uniform vertices ». Dans : *Proceedings of the Working Conference on Advanced Visual Interfaces*. 2002, p. 157-166 (cf. p. 90).

[36] K. Hayashi, M. Inoue, T. Masuzawa et H. Fujiwara. « A layout adjustment problem for disjoint rectangles preserving orthogonal order ». Dans : *Graph Drawing: 6th International Symposium, GD'98 Montréal, Canada, August 13–15, 1998 Proceedings 6*. Springer. 1998, p. 183-197 (cf. p. 90).

[37] K. He, X. Zhang, S. Ren et J. Sun. « Deep residual learning for image recognition ». Dans : *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, p. 770-778 (cf. p. 88).

[38] K. Hornik. « Approximation capabilities of multilayer feedforward networks ». Dans : *Neural Networks* 4.2 (1991), p. 251-257. issn : 0893-6080 (cf. p. 87).

[39] X. Huang, W. Lai, A. Sajeev et J. Gao. « A new algorithm for removing node overlapping in graph visualization ». Dans : *Information Sciences* 177.14 (2007), p. 2821-2844 (cf. p. 90).

[40] J. F. Kruiger, P. E. Rauber, R. M. Martins, A. Kerren, S. Kobourov et A. C. Telea. « Graph Layouts by t-SNE ». Dans : *Computer graphics forum*. T. 36. 3. Wiley Online Library. 2017, p. 283-294 (cf. p. 89).

[41] B. La Rosa, G. Blasili, R. Bourqui, D. Auber, G. Santucci, R. Capobianco, E. Bertini, R. **Giot** et M. Angelini. « State of the Art of Visual Analytics for eXplainable Deep Learning ». Dans : *Computer Graphics forum (CGF)* 42.1 (2023), p. 319-355 (cf. p. 80).

[42] A. Lambert, R. Bourqui et D. Auber. « Winding roads: Routing edges into bundles ». Dans : *Computer graphics forum*. T. 29. 3. Wiley Online Library. 2010, p. 853-862 (cf. p. 74).

[43] Y. Y. Leow, T. Laurent et X. Bresson. « GraphTSNE: A Visualization Technique for Graph-Structured Data ». Dans : *ICLR Workshop on Representation Learning on Graphs and Manifolds*. 2019 (cf. p. 87).

[44] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin et B. Guo. « Swin transformer: Hierarchical vision transformer using shifted windows ». Dans : *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, p. 10012-10022 (cf. p. 95).

[45] S. Mancoridis, B. S. Mitchell, C. Rorres, Y. Chen et E. R. Gansner. « Using Automatic Clustering to Produce High-Level System Organizations of Source Code ». Dans : *IEEE Proc. Int. Workshop on Program Understanding (IWPC'98)*. 1998, p. 45-53 (cf. p. 78).

[46] A. Mansfield. « Information technology–biometric performance testing and reporting–part 1: Principles and framework ». Dans : *ISO/IEC* (2006), p. 19795-1 (cf. p. 73).

[47] W. Meulemans. « Efficient optimal overlap removal: Algorithms and experiments ». Dans : *Computer Graphics Forum*. T. 38. 3. Wiley Online Library. 2019, p. 713-723 (cf. p. 90).

[48] K. Misue, P. Eades, W. Lai et K. Sugiyama. « Layout adjustment and the mental map ». Dans : *Journal of Visual Languages & Computing* 6.2 (1995), p. 183-210 (cf. p. 90).

[49] T. Moscovich, F. Chevalier, N. Henry, E. Pietriga et J.-D. Fekete. « Topology-Aware Navigation in Large Networks ». Dans : *SIGCHI conference on Human Factors in computing systems*. Sous la dir. d'A. Press. 2009, p. 2319-2328 (cf. p. 75, 78).

[50] L. Nachmanson, A. Nocaj, S. Bereg, L. Zhang et A. Holroyd. « Node overlap removal by growing a tree ». Dans : *Graph Drawing and Network Visualization: 24th International Symposium, GD 2016, Athens, Greece, September 19-21, 2016, Revised Selected Papers 24*. Springer. 2016, p. 33-43 (cf. p. 90).

[51] M. Okoe, R. Jianu et S. Kobourov. « Node-link or adjacency matrices: Old question, new insights ». Dans : *IEEE transactions on visualization and computer graphics* 25.10 (2018), p. 2940-2952 (cf. p. 86).

[52] W. Plant et G. Schaefer. « Visualisation and browsing of image databases ». Dans : *Multimedia Analysis, Processing and Communications*. Springer, 2011, p. 3-57 (cf. p. 77).

[53] L.-E. Pommé, R. Bourqui, R. **Giot** et D. Auber. « NetPrune: A Sparklines Visualization for Network Pruning ». Dans : *Visual Informatics (VI)* (2023). issn : 2468-502X. À para. (cf. p. 94).

[54] L.-E. Pommé, R. Bourqui, R. **Giot** et D. Auber. « Relative Confusion Matrix: Efficient Comparison of Decision Models ». Dans : *Proceedings of the 26th International Conference Information Visualisation (IV 2022)*. 2022 (cf. p. 93, 94).

[55] H. C. Purchase. *Experimental Human-Computer Interaction: A Practical Guide with Visual Examples*. Cambridge University Press, 2012 (cf. p. 85).

[56] S. C. R. Brockenauer. « Drawing Clusters and Hierarchies ». Dans : *Drawing Graphs: Methods and Models*. Sous la dir. de M. Kaufmann et D. Wagner. 2025. Springer-Verlag, 2001, p. 193-227. isbn : 3-540-42062-2 (cf. p. 77).

[57] L. Rampášek, M. Galkin, V. P. Dwivedi, A. T. Luu, G. Wolf et D. Beaini. « Recipe for a general, powerful, scalable graph transformer ». Dans : *Advances in Neural Information Processing Systems* 35 (2022), p. 14501-14515 (cf. p. 95).

[58] F. Rayar, S. Barrat, F. Bouali et G. Venturini. « Incremental hierarchical indexing and visualisation of large image collections ». Dans : *24th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*. 2016 (cf. p. 93).

[59] S. Song, C. Li, Y. Sun et C. Wang. « Vividgraph: Learning to extract and redesign network graphs from visualization images ». Dans : *IEEE Transactions on Visualization and Computer Graphics* (2022) (cf. p. 95).

[60] H. Strobelt, M. Spicker, A. Stoffel, D. Keim et O. Deussen. « Rolled-out wordles: A heuristic method for overlap removal of 2d data representatives ». Dans : *Computer Graphics Forum*. T. 31. 3pt3. Wiley Online Library. 2012, p. 1135-1144 (cf. p. 90).

[61] K. Sugiyama, S. Tagawa et M. Toda. « Methods for visual understanding of hierarchical system structures ». Dans : *IEEE Transactions on Systems, Man, and Cybernetics* 11.2 (1981), p. 109-125 (cf. p. 83).

[62] Y. Tanahashi, C.-H. Hsueh et K.-L. Ma. « An efficient framework for generating storyline visualizations from streaming data ». Dans : *IEEE transactions on visualization and computer graphics* 21.6 (2015), p. 730-742 (cf. p. 82).

[63] A. Telea et D. Auber. « Code flows: Visualizing structural evolution of source code ». Dans : *Computer Graphics Forum*. T. 27. 3. Wiley Online Library. 2008, p. 831-838 (cf. p. 82).

[64] S. M. Van Dongen. « Graph clustering by flow simulation ». Thèse de doct. 2000 (cf. p. 82).

[65] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser et I. Polosukhin. « Attention is all you need ». Dans : *Advances in neural information processing systems* 30 (2017) (cf. p. 95).

[66] Q. Wang, Z. Chen, Y. Wang et H. Qu. « A survey on ML4VIS: Applying machine learning advances to data visualization ». Dans : *IEEE transactions on visualization and computer graphics* 28.12 (2021), p. 5134-5153 (cf. p. 95).

[67] Y. Wang, Z. Jin, Q. Wang, W. Cui, T. Ma et H. Qu. « Deepdrawing: A deep learning approach to graph drawing ». Dans : *IEEE Transactions on Visualization and Computer Graphics* 26.1 (2019), p. 676-686 (cf. p. 87).

[68] A. Wu, Y. Wang, X. Shu, D. Moritz, W. Cui, H. Zhang, D. Zhang et H. Qu. « Ai4vis: Survey on artificial intelligence approaches for data visualization ». Dans : *IEEE Transactions on Visualization and Computer Graphics* (2021) (cf. p. 95).

[69] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang et S. Y. Philip. « A comprehensive survey on graph neural networks ». Dans : *IEEE transactions on neural networks and learning systems* 32.1 (2020), p. 4-24 (cf. p. 88).

[70] N. Xie, G. Ras, M. van Gerven et D. Doran. « Explainable deep learning: A field guide for the uninitiated ». Dans : *arXiv preprint arXiv:2004.14545* (2020) (cf. p. 79).

[71] N. Yager et T. Dunstone. « The biometric menagerie ». Dans : *IEEE transactions on pattern analysis and machine intelligence* 32.2 (2008), p. 220-230 (cf. p. 73).

[72] J. X. Zheng, S. Pawar et D. F. Goodman. « Graph drawing by stochastic gradient descent ». Dans : *IEEE transactions on visualization and computer graphics* 25.9 (2018), p. 2738-2748 (cf. p. 90).

Cinquième partie

Perspectives de recherche

Chapitre 17. Introduction aux perspectives de recherche 105
Chapitre 18. Visualisation pour l'apprentissage automatique 107
18.1. Rendre les systèmes de prédiction plus acceptables grâce à l'analyse visuelle et interactive . . 107
18.1.1. L'évaluation des explications 107
18.1.2. La confiance apportée aux explications 108
18.1.3. L'inclusion de conseils dans les interfaces 108
18.1.4. Inclusion de sémantique 108
18.2. Impliquer la société 109
18.2.1. L'alignement avec la société 109
18.2.2. Lien avec l'informatique écoresponsable 110
18.3. Rendre les explications plus versatiles 110
18.3.1. Plus de diversité de modèles et de données d'entrée 110
18.3.2. Plus de diversité dans les techniques d'explications 110
18.3.3. Plus de diversité dans les médiums de visualisation 111
18.4. Combler le fossé entre les communauté apprentissage, visualisation et explication 111
18.4.1. Sensibiliser mutuellement les différentes communautés 111
18.4.2. Collaboration entre les communautés visualisation, apprentissage et explication 112
18.4.3. Cadres logiciels 112
18.4.4. Alignement des domaines de recherche 112
Chapitre 19. Apprentissage pour la visualisation 113
19.1. Améliorer les techniques de plongement de données avec de l'apprentissage profond 113
19.1.1. Application au dessin de graphes 113
19.1.2. Application aux données multidimensionnelles 114
19.2. Remplacer des métriques d'évaluations coûteuses par des réseaux 114
19.3. Systématiser l'évaluation de techniques de visualisation avec de l'apprentissage profond . . 115
19.4. Faire le lien avec la communauté analyse de documents 115
Chapitre 20. Bibliographie des perspectives de recherche 117

Chapitre 17.

Introduction aux perspectives de recherche

Ce chapitre présente les perspectives de recherche qui guideront mon travail pour les cinq prochaines années. En raison de la dynamique de mes travaux sur la symbiose entre la Visualisation d'Information (VI) et l'Apprentissage Automatique (AA) (voir la partie IV), je compte principalement m'investir sur cette thématique. Le chapitre 18 présente les perspectives pour les aspects *visualisation pour l'apprentissage*, tandis que le chapitre 19 présente les aspects *apprentissage pour la visualisation*. De nombreux problèmes ouverts restent à investiguer dans les deux cas.

Le chapitre 18 liste des challenges à résoudre dans le cas de la VI pour l'AA, et plus particulièrement l'Apprentissage Profond (AP). Les challenges sont liés à différents critères exposés plus en détail : l'augmentation de l'acceptabilité des systèmes d'explication, l'implication du grand public dans les systèmes d'apprentissage grâce aux explications, la prise en compte de plus de systèmes d'explications, et l'amélioration des échanges entre les communautés VI et AA.

Le chapitre 19 liste des challenges à résoudre dans le cas de l'AA pour la VI. Il est fréquent d'utiliser des techniques de plongement de données multidimensionnelles dans un espace 2D. Toutes les méthodes de l'état de l'art ont leurs avantages et limites. Nous sommes convaincus qu'il est possible de réduire certaines de ces limites en utilisant des approches d'AP. Il est commun d'utiliser des fonctions de coût pour optimiser une visualisation ou l'évaluer. Cependant, celles-ci peuvent être calculatoirement coûteuses à utiliser (voir inutilisables lorsque le coût est polynomial ou exponentiel par rapport à la quantité de données). Dans ce cas, il est intéressant de les remplacer par les modèles qui miment leur comportement pour un coût d'utilisation négligeable, au détriment d'une étape d'apprentissage coûteuse. Les évaluations des systèmes de visualisation sont excessivement coûteuses et pourraient être facilitées grâce à de l'AP. Celui-ci pourrait intervenir pour comparer des méthodes (voir la partie précédente) ou pour remplacer des métriques d'évaluation coûteuses.

Les différents concepts détaillés dans cette partie sont illustrés dans la figure 17.1.

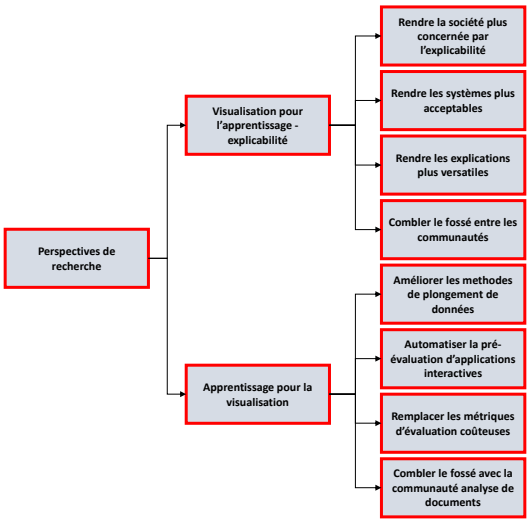


Figure 17.1. – Concepts présentés dans cette partie.

Chapitre 18.

Visualisation pour l'apprentissage automatique

Mes perspectives sont biaisées sur les aspects explicabilité des modèles ou des systèmes car mes travaux sur la Visualisation d'Information (VI) pour l'Apprentissage Automatique (AA) concernent ces aspects [32]. Les aspects à considérer cherchent à rendre : les systèmes d'explication plus acceptables grâce à l'Analyse Visuelle et Interactive (AVI), la société concernée par l'explicabilité grâce à l'AVI, les explications plus versatiles et combler le fossé entre les communauté AA, VI et explication.

18.1. Rendre les systèmes de prédiction plus acceptables grâce à l'analyse visuelle et interactive

L'acceptabilité d'un système est fortement liée à la confiance apportée par ses utilisateurs. Cette confiance a plusieurs aspects [49] tels que la confiance accordée à la compagnie qui a créé le système, la confiance accordée aux données utilisées pour le créer, ou sa capacité à être explicable. À noter que mes travaux précédents en explicabilité n'ont pas réellement pris en compte les aspects liés à la confiance.

Les points importants liés à l'acceptabilité concernent : l'amélioration de l'évaluation des explications, la création d'explications à laquelle on accorde directement plus de confiance, à l'ajout d'aide à l'utilisation du système interactif et de sémantique dans les explications (figure 18.1).

18.1.1. L'évaluation des explications

Bien qu'il existe dans la littérature en Intelligence Artificielle (IA) plusieurs travaux sur l'évaluation de méthodes d'explication, celles-ci sont principalement orientées sur les approches de type attribution de caractéristiques visualisées à l'aide de technique de représentation de carte de saillance [2, 27, 67, 69]. Ces évaluations considèrent des images statiques et ne prennent pas en compte les aspects liés à l'AVI. Comme il n'existe pas à ce jour de framework d'évaluation des explications dans un contexte d'AVI : il est nécessaire de mettre au point des procédures générales d'évaluation. J'insiste bien sur le terme général, car il semble compliqué de proposer une méthodologie d'évaluation trop précise étant donné que les systèmes d'explications interactifs cherchent à résoudre une tâche précise en utilisant différents types d'interactions et représentations.

Les travaux de la littérature se focalisent majoritairement sur la qualité du modèle. Cependant, bien qu'il existe quelques outils comme Stackatab [9] qui se focalise sur les jeux de données stockés dans un fichier CSV pour détecter des erreurs ou des données manquantes, ou FairVis [7] qui détecte des biais de distribution dans les jeux de données, l'aspect qualité des données est rarement mis en avant dans les études. C'est problématique, car la qualité de la base de données d'apprentissage est primordiale pour obtenir un modèle de qualité.

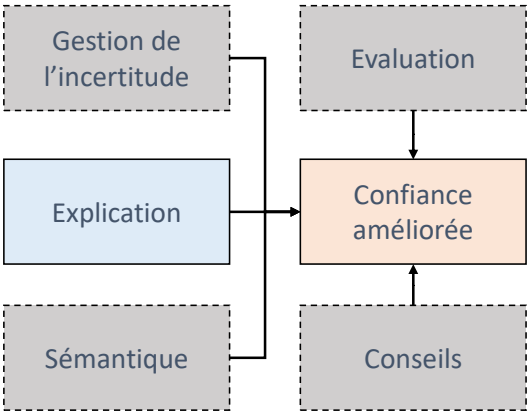


Figure 18.1. – L'acceptation des systèmes d'Apprentissage Profond (AP) peut se faire grâce à l'augmentation de la confiance apportée aux explications.

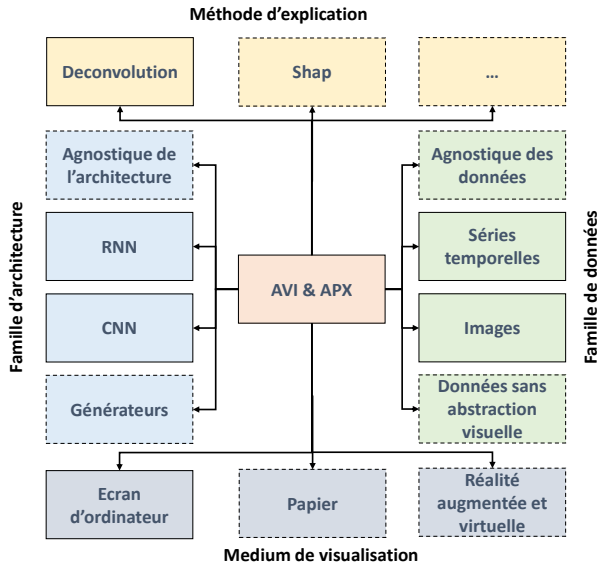


Figure 18.2. – Les outils d'AVI pour l'explicabilité devraient aider la société à mieux appréhender les systèmes reposants sur l'IA.

Les travaux futurs devraient prendre cet aspect en considération.

Un premier pas consiste à définir des concours sur des couples de taches/jeux de données en associant un protocole d'évaluation clair, précis et public (qui prend en compte les aspects d'interaction) sur chacun de ces couples. Les participants du concours ayant connaissance du protocole d'évaluation prendront conscience des limites des approches actuelles, ainsi que de leur proposition et chercheront à les contourner. S'inspirer des travaux en IA est naturellement intéressant. Par exemple, Meske et al. [38] décrivent diffèrent critères de qualité pour quantifier l'efficacité d'une explication : la *fidélité*, la *générabilité*, la *force d'explication*, la *compréhension*, l'*effort de vraisemblance*, et le *respect de la vie privée et de l'équité*. Le protocole d'évaluation devra définir quelles métriques [16, 56, 61] de la littérature peuvent être utilisées pour répondre à ces aspects, ou en définir de nouvelles le cas échéant.

18.1.2. La confiance apportée aux explications

Générer une représentation visuelle (éventuellement interactive) pour expliquer un résultat n'est pas suffisant ; il est nécessaire de faire en sorte qu'elle renforce la confiance des utilisateurs au système. Peu de travaux ont été effectués sur cet aspect. Une première piste pourrait être d'incorporer des informations sémantiques avec les explications (voir points suivants). Chatzimparmpas et al. [15] présentent un état de l'art sur les méthodes qui peuvent aider à améliorer la confiance. Un autre état de l'art [14] met également en avant les problèmes liés aux incertitudes : les modèles produisent des réponses incertaines. À noter que ce concept d'incertitude n'est pas directement lié à la réponse probabiliste du réseau [50]. Il n'existe pas de consensus clair sur la façon de représenter les incertitudes [29] dans le contexte de l'explicabilité, notamment pour les réseaux de neurones profonds qui souffrent de différentes sources d'incertitudes [23]. Cependant, une telle information est importante pour l'utilisateur, et donc sa confiance à accorder au système.

18.1.3. L'inclusion de conseils dans les interfaces

La plupart des systèmes d'explication génèrent des images et/ou des visualisations interactives qui aident à comprendre ce qui se passe. Il est difficile d'être plus précis dans la description, car les auteurs des méthodes ne savent pas toujours les tâches exactes qui doivent être résolues par les outils.

Il n'empêche que c'est souvent (pour ne pas dire toujours) à l'utilisateur de faire le travail intellectuel de compréhension, et d'inférer la connaissance à partir de cette information générée. Cette tâche n'est pas aisée. Par exemple, les méthodes mettant en avant les caractéristiques d'entrée importantes génèrent une carte de chaleur des pixels d'intérêt en superposition à l'image d'entrée : l'utilisateur identifie rapidement les pixels ayant un poids important sur la décision finale. Cependant, il n'a pas plus d'information que cette localisation. Il infère lui-même la raison de cette importance, sans savoir si le réseau c'est réellement focalisé sur cette partie pour la même raison. Les systèmes d'AVI peuvent guider plus l'utilisateur en proposant des aides spécifiques [12] et en lui laissant le vérifier à l'aide d'interactions ou d'expériences supplémentaires, afin de faciliter le travail de compréhension. Un exemple simple est la suggestion d'interactions de type « que ce passe-t-il si ? » [63] afin de modifier des paramètres ou des données et observer la différence de comportement du modèle. un autre exemple consiste à suggérer à l'utilisateur, à l'aide d'explications, comment il peut modifier son système d'apprentissage créé à l'aide d'un cadre logiciel visuel [22], afin de le rendre plus performant. Des exemples plus complexes restent à déterminer, probablement en utilisant des techniques d'interface homme-machine moderne.

18.1.4. Inclusion de sémantique

Tout comme l'inclusion de conseils dans les interfaces, nous pensons que les explications doivent inclure des informations sémantiques. Des travaux préliminaires de Panigutti et al. [44] utilisent des ontologies qui étiquettent les données d'entrée. Ils s'attendent à ce que cet étiquetage ne soit plus nécessaire dans le futur. Poli et al. [47] présentent un système de génération de phrases pour la segmentation d'image. Une

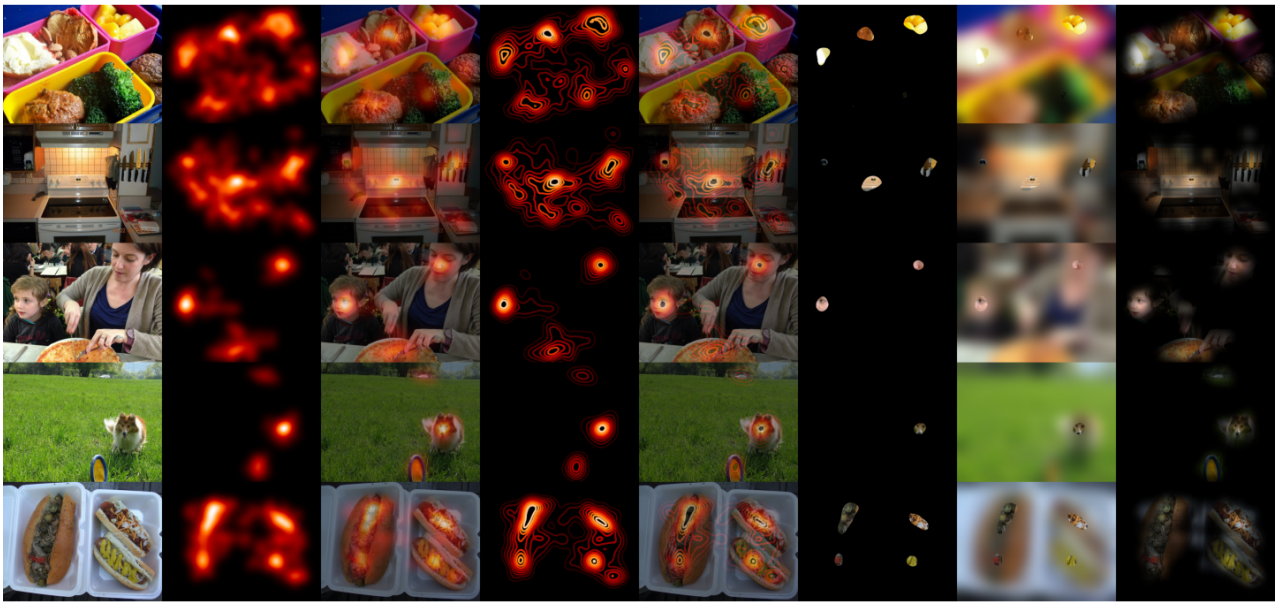


Figure 18.3. – Illustration de différentes méthodes de représentation de cartes de saillances recensées dans la littérature. La première colonne représente l'image classée et la seconde colonne une carte de saillance. La troisième colonne correspond à la technique de représentation majoritairement utilisée. Les autres colonnes sont peu utilisées, mais probablement plus efficaces.

telle approche pourrait être adaptée à des contextes similaires utilisant des systèmes d'AP. Ainsi pour en revenir à l'éternel exemple de la carte de saillance, nous espérons obtenir une information supplémentaire : cette zone de pixels est importante pour le réseau, car elle représente cette notion sémantique qui est importante pour cette classe.

Nous pensons que pour obtenir de tels résultats, il faut utiliser des approches hybrides qui mixent méthodes intrinsèques (modèles interprétables) et après-coup (modèles explicables). De telles approches commencent à émerger avec la gestion de concepts. La difficulté restant à définir ce qu'est un concept dans des systèmes manipulant des données autres que des images [30] ou des données tabulaires [65].

La verbalisation [53] des explications peut aider à apporter ces informations de sémantique. Par exemple, Pierrard et al. proposent un système qui justifie des résultat de segmentation d'organes en fonction de son positionnement relatif aux autres organes segmentés [46].

18.2. Impliquer la société

Le grand public doit se sentir concerné par l'AP, et plus généralement l'IA car elle repose énormément sur ces technologies (voir la figure 18.2 page ci-contre). L'explicabilité est la clé pour lui permettre de comprendre les avantages et les limites de tels systèmes. L'AVI est la clé pour rendre les explications divertissantes, compréhensibles et adoptables par la société. L'explicabilité peut également être utilisée dans un objectif de compréhension des dépenses énergétiques des modèles.

18.2.1. L'alignement avec la société

La plupart des outils d'AVI pour l'AP sont conçus pour des utilisateurs experts en AP ou dans un domaine d'application spécifique. Bien que les explications devraient également être compréhensibles par des non-experts, peu de travaux s'intéressent au grand public [24], alors qu'il s'agit d'une cible très importante. Si le grand public ne peut pas comprendre les avantages ou inconvénients d'un système d'AP, il ne peut pas lui faire confiance, ou prendre des décisions appropriées basées sur son utilisation dans leur vie quotidienne. Le même raisonnement peut être appliqué aux décideurs politiques ; d'autant plus que de nombreux gouvernements augmentent les obligations de transparence vis-à-vis des décisions prises par de tels systèmes. La CNIL propose des éclaircissements

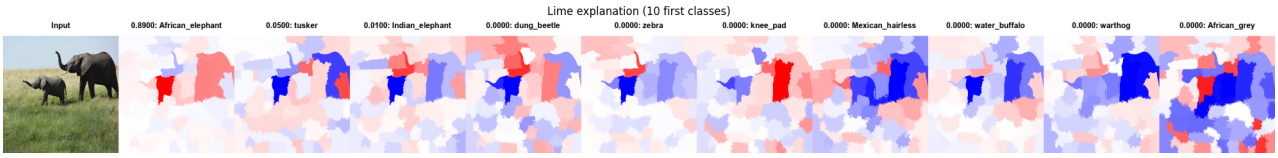


Figure 18.4. – Représentation « multiple petits » d'explications de type cartes de saillance calculées avec LIME pour les 10 premières classes prédites sur l'image de test.

à ce sujet (https://www.cnil.fr/fr/intelligence-artificielle/ia-comment-etre-en-conformite-avec-le-rgpd).

Il est donc primordial de proposer des méthodes d'explications et des outils pour les représenter qui ne sont pas dédiés aux experts.

18.2.2. Lien avec l'informatique écoresponsable

À notre connaissance, il n'existe pas d'outil d'AVI dédié à l'explication de l'empreinte carbone d'un modèle en plus de son explication. Un tel système serait un apport important, et intéressant, par la communauté d'IA afin d'aider à comprendre l'impact des modèles sur l'environnement. Cet impact doit être analysable sur l'intégralité de la chaîne de traitement : collecte des données, nettoyage, étiquetage, apprentissage, inférence ... Ce point est d'autant plus important que plusieurs gouvernements demandent à leurs citoyens, ainsi qu'aux laboratoires de recherche [36] de réduire leur émission de carbone. Le grand public a également accès, directement ou indirectement, à différents modèles de l'état de l'art (*p. ex.* ChatGPT [42] ou DALL-E [43]), utilisés principalement à des buts récréatifs, sans avoir la moindre idée du coût écologique de l'apprentissage ou de l'utilisation de tels systèmes.

18.3. Rendre les explications plus versatiles

La plupart des systèmes d'AVI ciblent des modèles de type réseau de convolution utilisés dans un contexte de classification avec peu de classes, sans nécessairement utiliser des méthodes d'explication de l'état de l'art. Ces systèmes devraient être plus versatiles en s'intéressant à une plus grande variété de familles de modèles, en utilisant des méthodes d'explications plus modernes, et en allant au-delà de l'écran standard d'un ordinateur.

18.3.1. Plus de diversité de modèles et de données d'entrée

Bien qu'il existe une grande diversité de modèles d'AP, seulement quelques familles ont été étudiées dans la littérature (principalement des CNN, des transformeurs, et, à moindre échelle, des RNN). Quelques familles, telles que les réseaux générateurs, ou les réseaux sur graphes, ont été peu étudiées. La plupart des travaux traitent de données de type image, car elles sont trivialement représentables. Il est nécessaire de se focaliser sur une famille plus large de modèles et de données, possiblement en proposant des systèmes d'AVI agnostiques du modèle et des données.

18.3.2. Plus de diversité dans les techniques d'explications

Nous observons que, jusqu'en 2021, (i) de nombreux travaux destinés à la communauté d'AVI incluent rarement des modèles et des méthodes d'explication récents, et (ii) les travaux destinés aux communautés AP et Apprentissage Profond Explicable (APX) reposent souvent sur des techniques de visualisation simple. Par exemple, les méthodes standard d'explicabilité type attribution de caractéristiques de l'état de l'art (*c.-à-d.* SHAP [35], Grad-CAM [52] et LIME [48]) ne sont incluses que dans cinq systèmes analysés dans cette enquête ; des méthodes plus anciennes sont utilisées à leur place (*p. ex.* déconvolution [66], cartes de saillance standard [54]). Les approches hybrides [13] sont tout simplement absentes des outils d'AVI. La même observation vaut pour les autres catégories de méthodes d'explication (voir la section 4.1) et les modèles utilisés. Si ces choix n'ont aucun impact sur la qualité des systèmes AVI, ils peuvent limiter leur utilité et leur diffusion, notamment lorsqu'ils s'adressent à des experts AP.

Spécificités à l'attribution de caractéristiques

En ce qui concerne les méthodes de type attribution de caractéristiques, la façon standard de représenter les cartes de saillance consiste à générer une carte de chaleur, souvent avec la palette `jet` et la superposer sur l'image. Cette représentation majoritaire est sous-optimale et devrait être remplacée par d'autres. Une amélioration mineure consiste à utiliser des palettes plus pertinentes, tandis qu'une amélioration majeure consiste à utiliser d'autres types de représentations [67] telles que l'utilisation de masque binaire (il faut donc binariser la

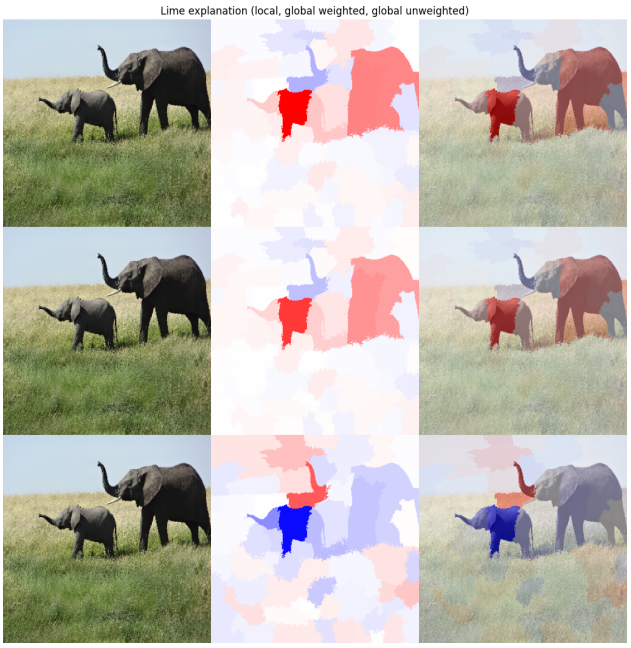


Figure 18.5. – Différentes stratégies de fusion de cartes d'explication générées par LIME. La première ligne représente l'explication de la classe majoritaire. La seconde ligne représente une somme pondérée de toutes les cartes par la probabilité de la classe associée. La troisième ligne représente la moyenne de toutes les cartes.

carte de saillance au préalable), de masque flou, ou d'isolignes (voir la figure 18.3 page 109 pour avoir une idée des représentations possibles).

La majorité des méthodes génèrent une carte de saillance par classe. Il est cependant intéressant d'avoir une vision globale pour l'ensemble des classes. L'utilisation d'une représentation de type « multiples petits » n'est pas satisfaisante (voir figure 18.4 page ci-contre), tout comme l'agrégation des différentes explications (voir figure 18.5). En considérant les multiples cartes de saillance comme étant des cartes de densité multiclasse, il est possible d'utiliser d'autres techniques de représentation [28]. Certaines peuvent être efficaces pour ce type d'application et méritent d'être étudiées, notamment en affichant des informations agrégées dans des diagrammes de Voronoi. Il est intéressant d'explorer et évaluer de telles approches.

18.3.3. Plus de diversité dans les médiums de visualisation

L'*écran d'ordinateur* reste le principal support utilisé pour afficher des explications. Cependant, d'autres types de médiums sont intéressants : *la représentation sur papier*, *la réalité virtuelle*, et *la réalité augmentée* qui ont leurs propres contraintes (encore plus restrictives).

La représentation sur papier manque d'interactivité par définition, mais reste un compagnon intéressant d'un système AVI. Concernant la représentation de réseaux, un exemple intéressant est présenté dans *Net2Vis* [4], mais l'interactivité peut toujours être apportée en utilisant des interfaces tangibles [21] associées à de la visualisation physique [19]. À noter que certains systèmes pour écran fournissent des représentations suffisamment simples pour être utilisés dans une version imprimée [11]. Concernant les explications elles-mêmes, il y a un grand vide dans l'espace des possibilités déjà explorées. Les articles publiés foisonnent de cartes de chaleur, fragments d'images ou autres résumés d'images. Il y a surement des avancées à apporter pour les données non-images et/ou à l'utilisation a des fins journalistiques plutôt que scientifiques.

La réalité virtuelle (RV) permet d'échapper virtuellement à la limitation des pixels d'un écran fixe en ajoutant la composante de profondeur. Il ne semble pas y avoir de travaux publiés sur la RV pour l'IA Explicable (IAX), mais les recherches futures [33] sur le sujet pourraient ouvrir de nouvelles voies pour expliquer les modèles d'AP. Également, dans un cadre pédagogique, le système « Ride Your Data » [10] pourrait être trivialement utilisé pour visualiser l'évolution de la fonction de coût d'un réseau au cours de l'apprentissage à l'aide de sa simulation de montagnes russes.

La réalité augmentée est également une alternative intéressante, mais elle a été très peu étudiée. Elle a montré son intérêt dans une étude liée aux systèmes de recommandation [39] ou sur le diagnostic des comportements erronés d'un robot [34]. Nous pouvons supposer sans risque qu'il serait utile et intéressant dans le contexte plus large du APX et qu'il devrait être étudié également.

18.4. Combler le fossé entre les communauté apprentissage, visualisation et explication

Combler le fossé entre les communautés apprentissage, visualisation et explication est une nécessité pour améliorer la qualité, l'acceptabilité et l'utilisabilité des systèmes (voir la figure 18.6 page suivante).

18.4.1. Sensibiliser mutuellement les différentes communautés

Ces trois communautés évoluent avec peu d'interactions entre elles. Il est nécessaire d'améliorer ces interactions afin qu'elles

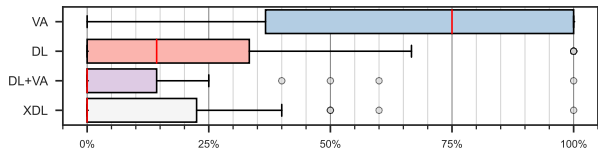


Figure 18.7. – Distribution des aires d'expertise parmi les auteurs d'un ensemble de papiers liant explicabilité pour les réseaux de neurones profonds et l'AVI.

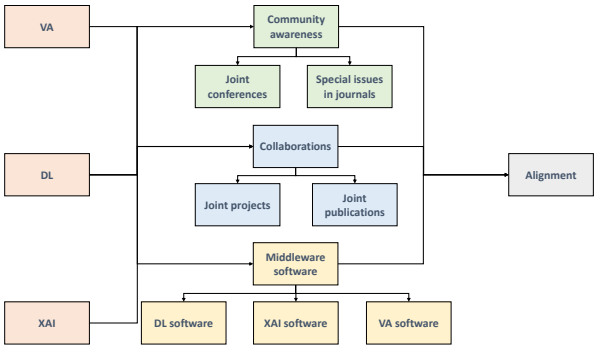


Figure 18.6. – Les outils d'AVI pour l'explicabilité devraient aider la société à mieux appréhender les systèmes reposant sur l'IA.

aient mutuellement connaissance des avancées et limites des autres afin de construire conjointement des systèmes efficaces. Pour le moment, plusieurs workshops dans des conférences IA et VI ont eu lieu indépendamment (j'ai d'ailleurs participé à l'organisation de trois d'entre eux, et suis en train d'en organiser d'autres). Il serait préférable d'organiser des évènements communs à toutes les communautés. La revue en ligne *distill.pub* était une autre initiative dans cette direction. Elle promouvait des articles interactifs, relus par des pairs, où les utilisateurs peuvent interagir avec les modèles et données du papier. Plusieurs articles traitent d'explicabilité et d'AP, ce qui prouve que l'interaction est un élément clé pour mieux les comprendre.

18.4.2. Collaboration entre les communautés visualisation, apprentissage et explication

Un plus grand nombre d'initiatives, et une prise de conscience plus profonde conduiraient à une augmentation des collaborations entre les communautés. En effet, nous observons que deux tiers des auteurs [32] (170 sur 234) des articles analysés proviennent de la communauté AVI, et un tiers de la communauté AP. Parmi eux, seule une petite fraction (23 auteurs) possède une expertise sur IAX (c'est-à-dire qu'elle a publié au moins deux articles sur le sujet). C'est également confirmé par la répartition des auteurs des articles (voir la figure 18.7), où l'AVI est le domaine dominant, alors que moins de la moitié des articles sont consacrés à l'XAI. L'objectif de cette phase sera de rendre la répartition plus équilibrée et de promouvoir l'inclusion d'experts de tous les domaines dès les premières phases de la conception de ces solutions.

18.4.3. Cadres logiciels

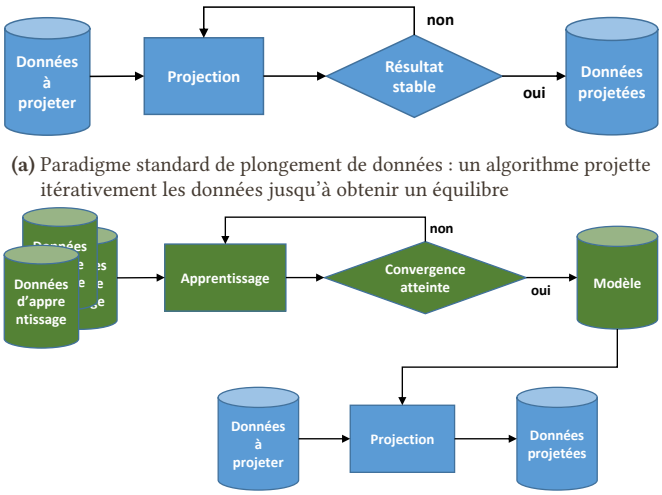
Une autre étape importante est la construction d'interfaces communes entre l'AVI et l'AP en termes de bibliothèques et d'outils. L'un des ingrédients clés du succès de l'AP est la disponibilité de bibliothèques et d'outils unifiés pour soutenir le développement rapide du domaine. Actuellement, pour télécharger une, ou un ensemble de données personnalisé dans un système d'analyse visuelle, l'utilisateur doit suivre des instructions, souvent adaptées au système AVI spécifique. Les mêmes difficultés apparaissent lorsque les chercheurs de l'AVI doivent adapter leurs systèmes à différents modèles et flux de travail. Par conséquent, il est nécessaire de disposer d'un ensemble d'outils et d'API qui, à partir des cadres déjà disponibles (*p. ex.* PyTorch [45], TensorFlow [1], OpenAI Gym [6]) peuvent être utilisés comme interface entre les bibliothèques AP et les bibliothèques AVI (*p. ex.* D3.js [5]). Idéalement, ils devraient abstraire l'accès aux cadres AP, facilitant la prise en charge d'un large éventail de modifications et accélérant la diffusion de ces systèmes. Bien que certaines tentatives initiales aient été faites, proposant des plugins pour les frameworks AP, il leur manque toujours la possibilité de modifier les composants visuels. La disponibilité de ces outils pourrait également augmenter le nombre de travaux qui publient leur code associé à leur papier.

18.4.4. Alignement des domaines de recherche

Les actions décrites dans les paragraphes précédents pourraient aider les communautés à combler l'écart temporel (les méthodes et modèles récents ne sont pas pris en compte rapidement par les outils d'analyse) entre la solution adoptée dans AP, IAX, et AVI. En effet, une sensibilisation et une collaboration accrues entre les communautés, ainsi que la disponibilité de nouveaux outils, faciliteraient et accéléreraient la mise en œuvre de nouvelles architectures et de méthodes IAX sur les systèmes AVI. Il serait possible de réduire l'écart temporel observé entre la disponibilité de nouvelles solutions d'IA pour le public et leur prise en charge par les systèmes AVI. En outre, les systèmes d'AVI qui prennent en charge les modèles et les méthodes d'explication de pointe pourraient accélérer davantage les innovations dans les domaines de l'AP et l'IAX, les principaux domaines et utilisateurs cibles de ces systèmes, rendant ces efforts profitables pour les trois domaines.

Chapitre 19.

Apprentissage pour la visualisation



(a) Paradigme standard de plongement de données : un algorithme projette itérativement les données jusqu'à obtenir un équilibre

(b) Paradigme futur de plongement de données : un algorithme projette directement les données. Cependant une phase d'apprentissage itérative a été nécessaire

Figure 19.1. – Changement de paradigme pour le plongement de données.

L'apprentissage pour la visualisation est un sujet déjà adopté par les communautés Intelligence Artificielle (IA) et Visualisation d'Information (VI) [62, 64, 68]. Il reste cependant toujours des fossés à combler dans les thématiques que nous avons commencé à explorer. La section 19.1 discute des méthodes de plongement de données par Apprentissage Profond (AP), la section 19.2 aborde le sujet de la définition de métriques d'évaluation à l'aide de réseaux, que la section 19.3 généralise à la procédure d'évaluation complète et la section 19.4 est plus générale sur l'analyse de documents.

19.1. Améliorer les techniques de plongement de données avec de l'apprentissage profond

Les outils de VI reposent souvent sur des techniques de plongement de données dans un espace 2D pour les représenter à l'écran. On peut identifier des variations de techniques en fonction de la nature des données. Nous nous intéresserons aux graphes et aux données multidimensionnelles. Ces méthodes sont généralement itératives et longues à s'exécuter. Utiliser un Réseau de Neurones Profond (RNP) peut aider à éviter cette étape itérative (qui, en fait, est déplacée au moment de l'apprentissage du réseau) afin d'exécuter une simple fonction.

Nous avons montré qu'une telle approche est faisable pour le dessin de graphe (voir la section 14.2 page 87) et nous avons montré que des approches basées optimisation par descente de gradient fonctionnent pour supprimer les chevauchements dans le plongement de données (voir la section 14.3 page 90). De tels résultats suggèrent qu'il est possible de trouver des architectures et des fonctions de coût capables de résoudre le problème de plongement de données à l'aide d'une fonction qui ne se base pas sur des itérations (voir la figure 19.1), et qui est donc plus rapide que les méthodes standard de l'état de l'art (*p. ex.* TSNE [60] ou UMAP [37]). Ces approches doivent nécessairement être apprises de façon non supervisée afin de ne pas se limiter à la reproduction de résultats obtenables avec des méthodes classiques.

19.1.1. Application au dessin de graphes

Nos travaux sur le dessin de graphes par AP (voir la section 14.1.1) ont montré qu'il est possible de dessiner un graphe de taille raisonnable à l'aide d'un Réseau de Neurones (RN) sur graphe. L'avantage de la technique est d'obtenir une projection des nœuds assez rapidement étant donné qu'il n'y a plus d'itérations comme avec les méthodes basées sur les forces. Cependant, elle fonctionne actuellement uniquement avec des graphes de taille raisonnable, et peut souffrir de problèmes de repliement lorsque la topologie de différents sous-graphes du graphe à dessiner est semblable.

Il est important de penser à d'autres formes d'architectures, notamment en se basant sur l'approche transformeur pour contrebalancer ce problème. En effet, le modèle d'attention

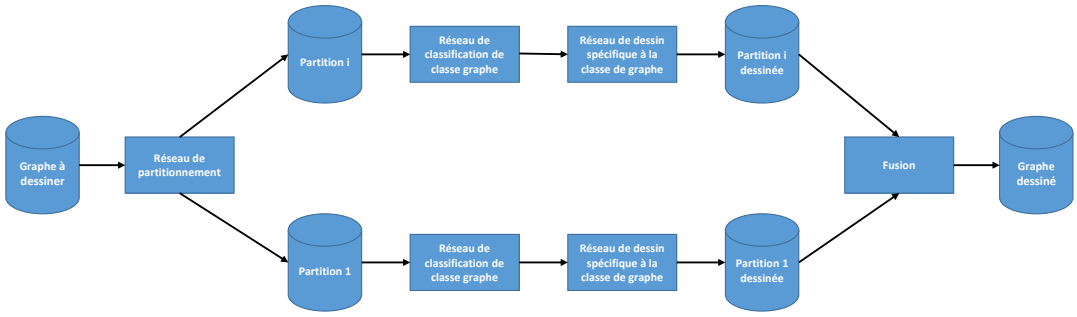


Figure 19.2. – Amélioration de la méthode de dessin de graphes en détectant automatiquement des partitions, utilisant des réseaux adaptés à chaque partition. L'approche peut naturellement être hiérarchique en appliquant la procédure sur chaque partition à dessiner jusqu'à un certain critère d'arrêt.

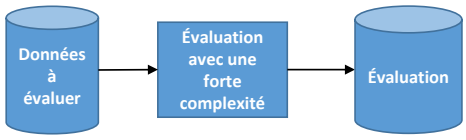
des transformeurs permettrait à chaque nœud d'avoir connaissance d'informations liées à la projection des autres afin de la prendre en compte pour éviter les recouvrements. De nombreuses méthodes de dessin de graphe se basent sur une décomposition (éventuellement récursive) du graphe en sous-graphes qui sont individuellement dessinés avec une technique appropriée à leur famille (*p. ex.* les quasi-arbres sont dessinés avec des méthodes de dessin d'arbre). On peut imaginer apporter ce genre de technique à notre approche de dessin de graphe : un réseau partitionne le graphe, un autre détermine quel réseau parmi une liste est le plus approprié pour afficher telle partition (voir la 19.2).

19.1.2. Application aux données multidimensionnelles

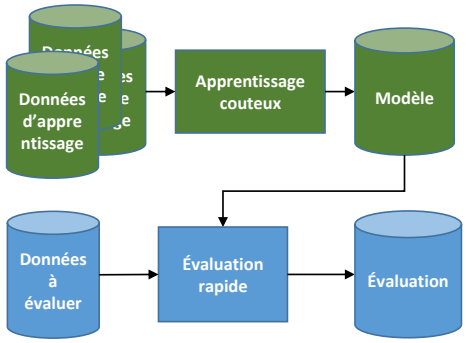
Nous avons vu que de nombreuses techniques permettent de projeter des données multidimensionnelles dans un espace $\mathbb{R}^2$ ou $\mathbb{N}^2$. Cependant, elles souffrent de différents défauts non mutuellement exclusifs (certains sont d'ailleurs également présents avec les techniques de dessin de graphe) : (i) non prise en compte de l'aire de la représentation visuelle de chaque symbole, (ii) non-unicité des coordonnées de destination, (iii) algorithme non stable, (iv) algorithme difficile à paramétrer pour obtenir une projection « fidèle ».

Des techniques à base d'auto-encodeurs ou de réseaux profonds spécifiques [20, 31] peuvent être utilisées pour faire des projections de données. Je pense qu'il sera intéressant de travailler sur des techniques reposant sur plusieurs briques technologiques distinctes afin de créer des méthodes de projection qui ne souffrent pas de tels défauts :

- l'utilisation de techniques d'AP pour générer des fonctions stables n'effectuant qu'une seule passe sur les données (traitement du point iii) ;
- l'utilisation de réseau sur graphe et/ou d'attention à la transformeur pour gérer les aspects recouvrement de données (traitement des points i et ii) ;
- l'autoapprentissage pour générer plusieurs familles de fonctions sélectionnables automatiquement en fonction du jeu de données à projeter (iv).



(a) Paradigme standard de calcul de métrique : un algorithme, potentiellement de complexité quadratique, calcule la métrique.

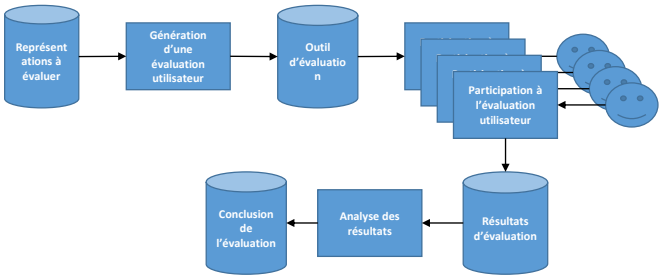


(b) Paradigme futur de calcul de métrique : un algorithme estime la métrique directement. Cependant une phase d'apprentissage a été nécessaire.

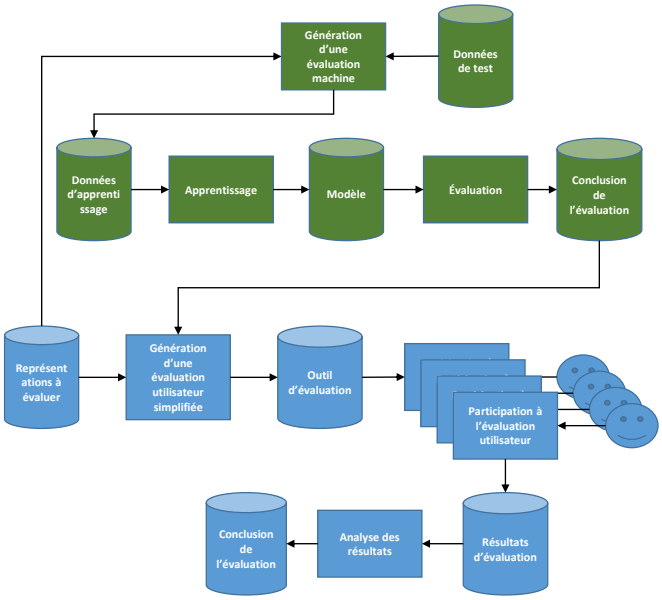
Figure 19.3. – Changement de paradigme pour le calcul de métrique d'évaluation.

19.2. Remplacer des métriques d'évaluations coûteuses par des réseaux

Plusieurs techniques de visualisation reposent sur l'utilisation d'une fonction de coût à optimiser (elle est donc appelée plusieurs fois lors du calcul). De telles fonctions peuvent être



(a) Paradigme standard d'évaluation de technique de visualisation : une expérience utilisateur, potentiellement mal configurée, est réalisée.



(b) Paradigme futur d'évaluation de technique de visualisation : une pré-étude est effectuée automatiquement afin de mieux configurer l'expérience utilisateur.

Figure 19.4. – Changement de paradigme d'évaluation de visualisation.

coûteuses à calculer, ce qui les rend inutilisables sur des jeux de données de taille conséquente.

Un exemple trivial est la fonction de stress couramment utilisée pour évaluer la qualité d'une projection et qui compare toutes les paires de points. Elle est donc en $O\left(n^2\right)$ avec n le nombre d'exemples dans le jeu de données, si on ne prend même pas en compte la dimension d (qui peut également être relativement importante) des données à projeter. On peut imaginer utiliser un RNP qui effectue une estimation de la fonction de stress avec une complexité inférieure (voir la figure 19.3). Pour diminuer la complexité, il est naturellement nécessaire de ne plus faire cette comparaison 2 à 2, mais d'utiliser des informations différentes (*p. ex.* des informations liées au rendu dans l'espace image et d'un résumé spécifique des données de l'espace multidimensionnel). Le problème n'est pas trivial à résoudre et mérite d'être investigué longuement, d'autant que la métrique sera toujours estimée. Bien entendu, on peut le généraliser à d'autres types de fonctions de coût et d'autres problèmes.

De tels travaux commencent à émerger dans la littérature. Par exemple, Tiezzi et al. proposent le concept de « neural esthete »[59] qui correspond à des réseaux estimant des propriétés d'évaluation de la qualité de dessin de graphe. On peut considérer que ce type d'approche se rapproche des « jumeaux numériques » [58] utilisés dans l'industrie pour simuler des systèmes complexes de façon algorithmique.

19.3. Systématiser l'évaluation de techniques de visualisation avec de l'apprentissage profond

Nous avons montré qu'il est possible d'utiliser des méthodes d'AP pour évaluer des techniques de représentation (voir la section 14.1) ou simuler la perception humaine [25, 26]. Cependant, nous ne savons pas si une telle approche est généralisable à d'autres techniques de représentation. Il est nécessaire d'élargir le panel d'évaluation, reproduire des évaluations existantes pour lesquelles les humains ont été remplacés par des machines, puis comparer ces résultats avec les originaux (voir la figure 19.4).

De même, les approches que nous avons testées fonctionnent uniquement sur des représentations statiques alors que les humains utilisent des interactions pour les guider. Il est donc nécessaire d'utiliser des approches automatiques qui reposent également sur des représentations interactives. L'utilisation d'apprentissage par renforcement [57] semble appropriée : l'agent déplace la souris pour acquérir de la connaissance grâce à la modification de la représentation en fonction de ses mouvements. Identifier les architectures qui ne se comportent pas comme des humains est également un aspect important, afin de favoriser les autres et améliorer leur compréhension.

Cai et al. [8] ont montré qu'on peut utiliser des réseaux pour reproduire la préférence humaine de dessin de réseau. Une telle approche peut être utilisée dans l'évaluation automatique de dessins.

19.4. Faire le lien avec la communauté analyse de documents

La communauté analyse de documents est spécialisée dans la compréhension de documents quelconques, souvent à partir de leur représentation image (*c.-à-d.* un scan du document original). Elle a réalisé des travaux sur différents aspects tels

que la segmentation de documents [41], l'extraction de données structurées comme des tables [55], ou des graphiques [17, 18].

La communauté visualisation a développé un ensemble de techniques pour générer des images à partir de données brutes, et de langage de description de représentation [51]. Elle est également capable de choisir automatiquement le bon type de représentation en fonction des données [40].

Andry et al. ont montré l'utilité d'utiliser des figures embellies pour le confort de l'utilisateur avec un impact minimal sur la lisibilité [3]. Il est donc intéressant d'étudier la possibilité d'utiliser des réseaux générateurs modernes pour créer automatiquement des versions embellies de graphiques simples.

La fusion des compétences de ces deux communautés ouvre la porte à d'autres perspectives, telles que la conversion automatique de documents depuis un document d'une certaine qualité de compréhension, vers un autre avec le même contenu, mais une meilleure qualité de compréhension. On peut imaginer la réorganisation du « layout » de façon globale ou locale, le changement de thème, la reconstruction de figures, le remplacement de figures par des tables et vice-versa.

Chapitre 20.

Bibliographie des perspectives de recherche

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu et X. Zheng. « TensorFlow: A System for Large-Scale Machine Learning ». Dans : *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*. USENIX Association, 2016, p. 265-283. isbn : 978-1-931971-33-1 (cf. p. 112).

[2] J. Adebayo, J. Gilmer, M. Muelly, I. J. Goodfellow, M. Hardt et B. Kim. « Sanity Checks for Saliency Maps ». Dans : (2018), p. 9525-9536 (cf. p. 107).

[3] T. Andry, C. Hurter, F. Lambotte, P. Fastrez et A. Telea. « Interpreting the effect of embellishment on chart visualizations ». Dans : *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 2021, p. 1-15 (cf. p. 116).

[4] A. Bäuerle, C. Van Onzenoodt et T. Ropinski. « Net2Vis–A Visual Grammar for Automatically Generating Publication-Tailored CNN Architecture Visualizations ». Dans : *IEEE transactions on visualization and computer graphics* 27.6 (2021), p. 2980-2991 (cf. p. 111).

[5] M. Bostock, V. Ogievetsky et J. Heer. « D^3 Data-Driven Documents ». Dans : *IEEE Transactions on Visualization and Computer Graphics* 17.12 (2011), p. 2301-2309 (cf. p. 112).

[6] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang et W. Zaremba. « OpenAI Gym ». Dans : *arXiv preprint arXiv:1606.01540* (2016). arXiv : 1606.01540 [cs.LG] (cf. p. 112).

[7] Á. A. Cabrera, W. Epperson, F. Hohman, M. Kahng, J. Morgenstern et D. H. Chau. « FairVis: Visual analytics for discovering intersectional bias in machine learning ». Dans : *2019 IEEE Conference on Visual Analytics Science and Technology (VAST)*. IEEE. 2019, p. 46-56 (cf. p. 107).

[8] S. Cai, S.-H. Hong, J. Shen et T. Liu. « A Machine Learning Approach for Predicting Human Preference for Graph Layouts ». Dans : *2021 IEEE 14th Pacific Visualization Symposium (PacificVis)*. 2021, p. 6-10 (cf. p. 115).

[9] P. D. S. Carvalho, P. Hitzelberger, F. Bouali et G. Venturini. « A Visual Technique to Assess the Quality of Datasets-Understanding the Structure and Detecting Errors and Missing Values in Open Data CSV Files ». Dans : *International Conference on Data Management Technologies and Applications*. T. 2. SCITEPRESS. 2015, p. 134-141 (cf. p. 107).

[10] V. Casamayou, Y. Jansen, P. Dragicevic et A. Prouzeau. « Ride Your Data: Raise your Arms, Scream, and Experience your Data from a Roller Coaster Cart ». Dans : *alt. VIS 2022*. 2022 (cf. p. 111).

[11] D. Cashman, A. Perer, R. Chang et H. Strobelt. « Ablate, Variate, and Contemplate: Visual Analytics for Discovering Neural Architectures ». Dans : *IEEE Transactions on Visualization and Computer Graphics* 26.1 (2020), p. 863-873 (cf. p. 111).

[12] D. Ceneda, T. Gschwandtner, T. May, S. Miksch, H.-J. Schulz, M. Streit et C. Tominski. « Characterizing Guidance in Visual Analytics ». Dans : *IEEE Transactions on Visualization and Computer Graphics* 23.1 (2017), p. 111-120 (cf. p. 108).

[13] M. Charachon, P.-H. Cournède, C. Hudelot et R. Ardon. « Visual explanation by unifying adversarial generation and feature importance attributions ». Dans : *Interpretability of Machine Intelligence in Medical Image Computing, and Topological Data Analysis and Its Applications for Medical Data: 4th International Workshop, iMIMIC 2021, and 1st International Workshop, TDA4MedicalData 2021, Held in Conjunction with MICCAI 2021, Strasbourg, France, September 27, 2021, Proceedings 4*. Springer. 2021, p. 44-55 (cf. p. 110).

[14] A. Chatzimparmpas, R. M. Martins, I. Jusufi et A. Kerren. « A survey of surveys on the use of visualization for interpreting machine learning models ». Dans : *Information Visualization* 19.3 (2020), p. 207-233 (cf. p. 108).

[15] A. Chatzimparmpas, R. M. Martins, I. Jusufi, K. Kucher, F. Rossi et A. Kerren. « The State of the Art in Enhancing Trust in Machine Learning Models with the Use of Visualizations ». Dans : *Computer Graphics Forum* (2020). issn : 1467-8659 (cf. p. 108).

[16] R. Xu-Darme, J. Benois-Pineau, R. **Giot**, G. Quénot, Z. Chihani, M.-C. Rousset et A. Zhukov. « On the stability, correctness and plausibility of visual explanation methods based on feature importance ». Dans : *Proceedings of the 20th International Conference on Content-based Multimedia Indexing (CBMI), special session on Explainability in Multimedia Analysis (ExMA)*. 2023 (cf. p. 108).

[17] K. Davila, B. U. Kota, S. Setlur, V. Govindaraju, C. Tensmeyer, S. Shekhar et R. Chaudhry. « ICDAR 2019 competition on harvesting raw tables from infographics (chart-infographics) ». Dans : *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE. 2019, p. 1594-1599 (cf. p. 116).

[18] K. Davila, S. Setlur, D. Doermann, B. U. Kota et V. Govindaraju. « Chart mining: A survey of methods for automated chart analysis ». Dans : *IEEE transactions on pattern analysis and machine intelligence* 43.11 (2020), p. 3799-3819 (cf. p. 116).

[19] P. Dragicevic, Y. Jansen et A. Vande Moere. « Data physicalization ». Dans : *Handbook of Human Computer Interaction* (2020), p. 1-51 (cf. p. 111).

[20] M. Espadoto, N. S. T. Hirata et A. C. Telea. « Self-supervised dimensionality reduction with neural networks and pseudo-labeling ». Dans : *Proceedings*. 2021 (cf. p. 114).

[21] K. P. Fishkin. « A taxonomy for and analysis of tangible interfaces ». Dans : *Personal and Ubiquitous computing* 8.5 (2004), p. 347-358 (cf. p. 111).

[22] J. Françoise, B. Caramiaux et T. Sanchez. « Marcelle: composing interactive machine learning workflows and interfaces ». Dans : *The 34th Annual ACM Symposium on User Interface Software and Technology*. 2021, p. 39-53 (cf. p. 108).

[23] J. Gawlikowski, C. R. N. Tassi, M. Ali, J. Lee, M. Humt, J. Feng, A. Kruspe, R. Triebel, P. Jung, R. Roscher et al. « A survey of uncertainty in deep neural networks ». Dans : *Artificial Intelligence Review* (2023), p. 1-77 (cf. p. 108).

[24] L. H. Gilpin, C. Testart, N. Fruchte et J. Adebayo. « Explaining Explanations to Society ». Dans : *arXiv preprint arXiv:1901.06560* (2018) (cf. p. 109).

[25] L. Giovannangeli, R. Bourqui, R. **Giot** et D. Auber. « Color and Shape efficiency for outlier detection from automated to user evaluations ». Dans : *Visual Informatics (VI)* (2022). issn : 2468-502X (cf. p. 115).

[26] L. Giovannangeli, R. **Giot**, D. Auber, J. Benois-Pineau et R. Bourqui. « Analysis of Deep Neural Networks Correlations with Human Subjects on a Perception Task ». Dans : *Proceedings of the 25th International Conference Information Visualisation (IV 2021)*. 2021, p. 8 (cf. p. 115).

[27] T. Gomez, T. Fréour et H. Mouchère. « Metrics for saliency map evaluation of deep learning explanation methods ». Dans : *International Conference on Pattern Recognition and Artificial Intelligence*. Springer. 2022, p. 84-95 (cf. p. 107).

[28] J. Jo, F. Vernier, P. Dragicevic et J.-D. Fekete. « A declarative rendering model for multiclass density maps ». Dans : *IEEE transactions on visualization and computer graphics* 25.1 (2018), p. 470-480 (cf. p. 111).

[29] A. Kamal, P. Dhakal, A. Y. Javaid, V. K. Devabhaktuni, D. Kaur, J. Zaientz et R. Marinier. « Recent advances and challenges in uncertainty visualization: a survey ». Dans : *Journal of Visualization* 24.5 (2021), p. 861-890 (cf. p. 108).

[30] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas et al. « Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav) ». Dans : *International conference on machine learning*. PMLR. 2018, p. 2668-2677 (cf. p. 109).

[31] Y. Kim, M. Espadoto, S. Trager, J. B. Roerdink et A. Telea. « SDR-NNP: Sharpened dimensionality reduction with neural networks ». Dans : *Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*. SciTePress. 2022, p. 63-76 (cf. p. 114).

[32] B. La Rosa, G. Blasili, R. Bourqui, D. Auber, G. Santucci, R. Capobianco, E. Bertini, R. **Giot** et M. Angelini. « State of the Art of Visual Analytics for eXplainable Deep Learning ». Dans : *Computer Graphics forum (CGF)* 42.1 (2023), p. 319-355 (cf. p. 107, 112).

[33] J. G. Linders. « Visualizing Explainable Artificial Intelligence in Virtual Reality ». Mém. de mast. Eindhoven University of Technology, 2023 (cf. p. 111).

[34] H. Liu, Y. Zhu et S.-C. Zhu. « Patching interpretable And-Or-Graph knowledge representation using augmented reality ». Dans : *Applied AI Letters* 2.4 (2021), e43 (cf. p. 111).

[35] S. M. Lundberg et S.-I. Lee. « A Unified Approach to Interpreting Model Predictions ». Dans : *Advances in Neural Information Processing Systems*. Sous la dir. d'I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan et R. Garnett. T. 30. Curran Associates, Inc., 2017 (cf. p. 110).

[36] J. Mariette, O. Blanchard, O. Berné et T. B. Ari. « An open-source tool to assess the carbon footprint of research ». Dans : *arXiv preprint arXiv:2101.10124* (2021) (cf. p. 110).

[37] L. McInnes, J. Healy et J. Melville. « Umap: Uniform manifold approximation and projection for dimension reduction ». Dans : (2020). arXiv : 1802.03426v3 [stat] (cf. p. 113).

[38] C. Meske, E. Bunde, J. Schneider et M. Gersch. « Explainable artificial intelligence: objectives, stakeholders, and future research opportunities ». Dans : *Information Systems Management* (2021), p. 1-11 (cf. p. 108).

[39] D. Mora, R. Zimmermann, D. Cirqueira, M. Bezbradica, M. Helfert, A. Auinger et D. Werth. « Who wants to use an augmented reality shopping assistant application? » Dans : *Proceedings of the 4th International Conference on Computer-Human Interaction Research and Applications-WUDESHI-DR*. 2020 (cf. p. 111).

[40] D. Moritz, C. Wang, G. L. Nelson, H. Lin, A. M. Smith, B. Howe et J. Heer. « Formalizing visualization design knowledge as constraints: Actionable and extensible models in draco ». Dans : *IEEE transactions on visualization and computer graphics* 25.1 (2018), p. 438-448 (cf. p. 116).

[41] S. A. Oliveira, B. Seguin et F. Kaplan. « dhSegment: A generic deep-learning approach for document segmentation ». Dans : *2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)*. IEEE. 2018, p. 7-12 (cf. p. 116).

[42] OpenAi. *ChatGPT: Optimizing Language Models for Dialogue* (cf. p. 110).

[43] OpenAi. *DALL·E API Now Available in Public Beta* (cf. p. 110).

[44] C. Panigutti, A. Perotti et D. Pedreschi. « Doctor XAI: an ontology-based approach to black-box sequential data classification explanations ». Dans : *Proceedings of the 2020 conference on fairness, accountability, and transparency*. 2020, p. 629-639 (cf. p. 108).

[45] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai et S. Chintala. « PyTorch: An Imperative Style, High-Performance Deep Learning Library ». Dans : *Advances in Neural Information Processing Systems*. Sous la dir. de H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox et R. Garnett. T. 32. Curran Associates, Inc., 2019 (cf. p. 112).

[46] R. Pierrard, J.-P. Poli et C. Hudelot. « Spatial relation learning for explainable image classification and annotation in critical applications ». Dans : *Artificial Intelligence* 292 (2021), p. 103434 (cf. p. 109).

[47] J.-P. Poli, W. Ouerdane et R. Pierrard. « Generation of Textual Explanations in XAI: the Case of Semantic Annotation ». Dans : *2021 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. IEEE. 2021, p. 1-6 (cf. p. 108).

[48] M. T. Ribeiro, S. Singh et C. Guestrin. « "Why Should I Trust You?" » Dans : *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2016 (cf. p. 110).

[49] F. Rossi. « Building trust in artificial intelligence ». Dans : *Journal of international affairs* 72.1 (2018), p. 127-134 (cf. p. 107).

[50] T. Sanchez, B. Caramiaux, P. Thiel et W. E. Mackay. « Deep learning uncertainty in machine teaching ». Dans : *27th International Conference on Intelligent User Interfaces*. 2022, p. 173-190 (cf. p. 108).

[51] A. Satyanarayan, D. Moritz, K. Wongsuphasawat et J. Heer. « Vega-lite: A grammar of interactive graphics ». Dans : *IEEE transactions on visualization and computer graphics* 23.1 (2016), p. 341-350 (cf. p. 116).

[52] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh et D. Batra. « Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization ». Dans : *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2017 (cf. p. 110).

[53] R. Sevastjanova, F. Becker, B. Ell, C. Turkay, R. Henkin, M. Butt, D. Keim et E. Mennatallah. « Going beyond Visualization. Verbalization as Complementary Medium to Explain Machine Learning Models ». Dans : (2018) (cf. p. 109).

[54] K. Simonyan et A. Zisserman. « Very Deep Convolutional Networks for Large-Scale Image Recognition ». Dans : *arXiv preprint arXiv:1409.1556* (2014). arXiv : 1409.1556 [cs.CV] (cf. p. 110).

[55] B. Smock, R. Pesala et R. Abraham. « PubTables-1M: Towards comprehensive table extraction from unstructured documents ». Dans : *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, p. 4634-4642 (cf. p. 116).

[56] F. Sovrano, S. Sapienza, M. Palmirani et F. Vitali. « Metrics, explainability and the European AI act proposal ». Dans : *J* 5.1 (2022), p. 126-138 (cf. p. 108).

[57] R. S. Sutton, A. G. Barto et al. « Reinforcement learning ». Dans : *Journal of Cognitive Neuroscience* 11.1 (1999), p. 126-134 (cf. p. 115).

[58] F. Tao, B. Xiao, Q. Qi, J. Cheng et P. Ji. « Digital twin modeling ». Dans : *Journal of Manufacturing Systems* 64 (2022), p. 372-389 (cf. p. 115).

[59] M. Tiezzi, G. Ciravegna et M. Gori. « Graph neural networks for graph drawing ». Dans : *IEEE Transactions on Neural Networks and Learning Systems* (2022) (cf. p. 115).

[60] L. Van der Maaten et G. Hinton. « Visualizing data using t-SNE. » Dans : *Journal of machine learning research* 9.11 (2008) (cf. p. 113).

[61] G. Vilone et L. Longo. « Notions of explainability and evaluation approaches for explainable artificial intelligence ». Dans : *Information Fusion* 76 (2021), p. 89-106 (cf. p. 108).

[62] Q. Wang, Z. Chen, Y. Wang et H. Qu. « A survey on ML4VIS: Applying machine learning advances to data visualization ». Dans : *IEEE transactions on visualization and computer graphics* 28.12 (2021), p. 5134-5153 (cf. p. 113).

[63] J. Wexler, M. Pushkarna, T. Bolukbasi, M. Wattenberg, F. Viégas et J. Wilson. « The what-if tool: Interactive probing of machine learning models ». Dans : *IEEE transactions on visualization and computer graphics* 26.1 (2019), p. 56-65 (cf. p. 108).

[64] A. Wu, Y. Wang, X. Shu, D. Moritz, W. Cui, H. Zhang, D. Zhang et H. Qu. « Ai4vis: Survey on artificial intelligence approaches for data visualization ». Dans : *IEEE Transactions on Visualization and Computer Graphics* (2021) (cf. p. 113).

[65] M. E. Zarlenga, Z. Shams, M. E. Nelson, B. Kim et M. Jamnik. « TabCBM: Concept-based Interpretable Neural Networks for Tabular Data ». Dans : *Transactions on Machine Learning Research* (2023) (cf. p. 109).

[66] M. D. Zeiler et R. Fergus. « Visualizing and Understanding Convolutional Networks ». Dans : *Computer Vision – ECCV 2014*. Springer International Publishing, 2014, p. 818-833 (cf. p. 110).

[67] Y. Zhao, M. Li et M. Berger. « Graphical Perception of Saliency-based Model Explanations ». Dans : *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 2023, p. 1-15 (cf. p. 107, 110).

[68] S. Zhu, G. Sun, Q. Jiang, M. Zha et R. Liang. « A survey on automatic infographics and visualization recommendations ». Dans : *Visual Informatics* 4.3 (2020), p. 24-40 (cf. p. 113).

[69] A. Zhukov, J. Benois-Pineau et R. **Giot**. « Evaluation of Explanation Methods of AI - CNNs in Image Classification Tasks with Reference-based and No-reference Metrics ». Dans : *Advances in Artificial Intelligence and Machine Learning (OAJAIML)* 3 (2023), p. 620-646 (cf. p. 107).

Sixième partie

Conclusion

Chapitre 21.

Conclusion générale

Ce manuscrit a présenté une partie des travaux que j'ai effectués depuis mon arrivée à l'IUT de Bordeaux et au LaBRI en septembre 2013.

La partie II a concerné certains travaux liés à l'utilisation de techniques d'analyse de données et d'Apprentissage Automatique (AA) dans différents contextes tels que (a) l'authentification biométrique, et (b)la génération de carte de saillance pour expliquer les Réseau de Neurones (RN) profonds. Les autres travaux non présentés concernent la classification hiérarchique de documents, la détection de panne dans les systèmes de vélos en libre-service, la prédiction de fraude dans les systèmes de paiement mobile, et la segmentation d'image satellite. Ces travaux reposent principalement sur les connaissances acquises lors de mes travaux de thèse de doctorat. Des collaborations avec des collègues des universités de Caen Basse-Normandie/France, Campinas/Brésil, São Paulo/Brésil, Fraunhofer/Allemagne, et d'Orange Labs.France ont permis de réaliser certains de ces travaux.

La partie III a concerné certains travaux liés à la Visualisation d'Information (VI). Les méthodes présentées concernent (a) le dessin de grands graphes, afin de gérer convenablement le placement des nœuds et le routage des arêtes, ainsi que (b) la représentation de masses de données dans une grille. Les autres travaux non présentés concernent (a) la création d'un prototype de rendu de glyphes 2D reposant sur le « geometry shader » et l'utilisation de fonctions de distance pour représenter les glyphes afin de diminuer au maximum le nombre de triangles à manipuler et d'informations à transmettre à la carte graphique, ainsi que de (b) la VI pour visualisation pour la détection de fraude dans des systèmes de paiement mobile. Ces travaux reposent principalement sur les nouvelles connaissances acquises après ma prise de fonction au LaBRI et à l'IUT de Bordeaux.

La partie IV a concerné les travaux qui combinent les compétences des parties II et III : la symbiose entre la VI et l'AA. 'A mon avis, il s'agit des travaux les plus intéressants de ce manuscrit étant donné qu'ils concernent deux thématiques qui étaient indépendantes en 2013, bien que ça soit de moins en moins le cas aujourd'hui. Deux familles de travaux sont présentées : la VI pour l'AA et l'AA pour la VI. La VI pour l'AA a été illustrée sur l'application pour des bases de documents étiquetés, l'authentification biométrique, et l'explication des réseaux de neurones profonds. L'AA pour la VI a été illustré sur l'évaluation de techniques de représentation, le dessin de graphes de taille raisonnable, et la suppression des chevauchements de glyphes dans ces dessins de graphes. Les travaux liés à l'analyse des limites de perception humaines, ou la suppression de chevauchements dans les nuages de points dans l'espace-écran, plutôt que l'espace géométrique n'ont pas été présentés. Des collaborations avec des collègues des universités de Campinas/Brésil, Rome Sapienza/Italie, New York City/États-Unis d'Amérique, Rafik Hariri/Liban ont permis de réaliser certains de ces travaux.

La partie V présente les perspectives de recherche pour mes travaux à venir dans les prochaines années sur la symbiose entre la VI et l'AA. L'obtention de cette Habilitation à Diriger des Recherches me permettra de poursuivre dans cette thématique, avec de nouvelles directions de doctorants en totale autonomie. Les perspectives listées prouvent que de nombreuses améliorations de l'état-de-l'art sont encore possibles.

Un (petit) pas vers l'explicabilité de l'intelligence artificielle

Résumé : Ce manuscrit retrace une partie de mes contributions depuis mon arrivée au LaBRI en septembre 2013, ainsi que les perspectives de recherche pour les prochaines années. Les aspects principaux de ma recherche concernent :

- L'*analyse de données et l'apprentissage automatique*. Je présente des travaux liés à l'authentification biométrique et l'attribution de caractéristiques dans les réseaux de neurones profonds.
- La *visualisation d'informations*. Je décris des travaux sur le dessin de grands graphes et la représentation de données dans des grilles.
- La *symbiose entre la visualisation d'informations et l'apprentissage automatique*. Je discute des travaux sur la visualisation pour l'apprentissage (en authentification biométrique, analyse de base étiquetée et visualisation pour les réseaux de neurones profonds), et l'apprentissage pour la visualisation (pour la comparaison automatique de techniques de visualisation, le dessin de graphes et la suppression de chevauchements dans les nœuds en dessin de graphe). Les perspectives de recherche se focalisent sur cette thématique.

Mots-clés : Apprentissage automatique, Authentification biométrique, Visualisation de graphes, Visualisation de masses de données, Explicabilité des modèles d'apprentissage, Apprentissage pour la visualisation, Visualisation pour l'apprentissage.

A (small) step to the explainability of artificial intelligence

Abstract: This document outlines my contributions since joining the LaBRI in September 2013, and my research plans for the coming years. The primary areas of research I discuss are:

- *Data analysis and machine learning*. Here, I present work on biometric authentication and how features are attributed within deep neural networks.
- *Information visualization*. This involves my research on techniques for drawing large graphs and grid-based visualization methods.
- The *intersection of information visualization and machine learning*. I describe research on using visualization to aid machine learning, including biometric authentication, analysis and visualization of labeled datasets, and enhancing the understanding of deep neural networks. Additionally, I discuss using machine learning to improve visualization, such as automatically comparing visualization techniques, improving graph drawing, and removing overlaps in graph visualization. Research perspectives focus on this aspect.

Keywords: Machine learning, Biometric authentication, Graph visualization, Big data visualization, Explainability in machine learning, Machine learning for visualization, Visualization for machine learning.

LaBRI
UMR CNRS 58000, Université de Bordeaux, 33000 Bordeaux, France.