Contributions aux communications inter-vues pour l'apprentissage collaboratif

Résumé de thèse

Denis Maurel

1 Contexte

Cette thèse portant sur l'étude des communications au sein de modèles d'apprentisage automatique collaboratif a été dirigée par Raja Chicky (ISEP) et co-encadrée par Jérémie Sublime (ISEP, Université Paris 13) et Sylvain Lefebvre (ISEP).

L'objectif principal de cette thèse était d'étudier les communications inter-vues au sein de modèles d'apprentissage collaboratifs afin d'améliorer la transmission d'informations entre les vues. Cette idée a été déclinée suivant deux axes suivant le type d'application concerné :

— Le clustering collaboratif pour lequel chaque vue disposera initialement d'un clustering local qui sera ensuite modifié afin d'arriver à une série de concensus entre les vues. La modification de chaque clustering local se base sur l'échange d'informations entre les vues, afin que les resultats obtenus localement puisse être utilisés par les vues externes. Le clustering collaboratif s'attache à trouver un concensus aussi global que possible plutôt qu'à améliorer chaque clustering local. Le clustering collaboratif est à distinguer de l'ensemble learning qui lui chercher à trouver un concensus unique entre toutes les vues à l'aide d'une fusion au sein d'un modèle global de l'ensemble des clusterings locaux. Au sein de cette thèse, nous parleros majoritairement de la version horizontale du clustering collaboratif, pour laquelle chaque vue dispose du même ensemble d'individus décrits au sein de chaque vue par un ensemble différentes de caractéristiques. Deux sous-axes ont été explorés concernant les communications inter-vues :

  1. L'optimisation des coefficients définissant l'importance que chaque vue accorde à l'information reçue de ses paires. Pour cela nous proposons une nouvelle méthode d'apprentissage permettant d'adpater dynamiquement ces poids à l'aide d'un apprentissage.
  2. La proposition d'une méthode d'apprentissage au cours du temps (que nous qualifierons d'en ligne) permettant à des vues de communiquer au fil du temps afin de faire évoluer les résultats obtenus localement à d'éventuels changements de distribution.

— La reconstruction collaborative dont le but est de reconstruire localement des données manquantes à l'aide d'informations présentes dans les vues externes. Cette application est définie est développée dans cette thèse avec la proposition d'un système permettant d'inférer l'approximation d'un individu à l'aide entre autre de réseaux de neurones. Ces réseaux seront utilisés soit pour coder l'information à transférer afin d'assurer une sécurité minimum, soit pour inférer les valeurs locales d'un individu en se basant sur l'information reçue de la vue externe.

2 Clustering collaboratif

Le clustering collaboratif est défini par un ensemble de base de données (appelées vues) ayant chacune opéré un clustering sur leurs données locales. Le but du clustering collaboratif va être de faire s'échanger de l'information entre les vues afin de modifier chaque clustering local pour finalement se rapprocher d'un concensus entre les vues.

Se pose alors le problème du recoupement d'information lorsqu'une vue locale reçoit des informations provenant de plus d'une source externe. Les méthodes de l'état de l'art se basent sur une pondération de ces informations à l'aide de coefficients scalaires [1, 10, 7, 4, 13, 11]. Cependant, la méthode de définition de ces coefficients est à chaque fois empirique, et c'est sur ce constat que se basent les travaux présentés dans la première partie de cette thèse. Un second constat après étude de l'état de l'art à été qu'il n'existait actuellement pas de méthode permettant d'effectuer un apprentissage en ligne (au fil du temps) de modèles collaboratifs. Un second sous axe d'exploration a ainsi consisté en la modification d'une méthode existante de clustering collaboratif [4] basée sur des cartes auto adaptatrices [6] afin de l'adapter à l'apprentissage en ligne. Ces travaux étaient de plus motivés par la volonté de rendre les modèles collaboratifs réactifs aux éventuels changement au cours du temps dans la distribution des données.

L'ensemble des travaux présentés dans cette section se base sur la fonction de coût définissant le score du modèle à chaque instant :

$$Q^i = Q^i_{local}(V_i) + Q^i_{collab}(V_i, V_{j\neq i}) \tag{1}$$

$$= Q^i_{local}(V_i) + \sum_{j\neq i} C^i_j(V_i, V_j) \tag{2}$$

Ces formules contiennent l'ensemble des éléments nécessaires pour définir un problème de clustering collaboratif. Q représente à chaque fois un critère d'évaluation, Q^i représente la valeur de ce critère pour la i-ème vue V_i, avec une distinction entre Q^i_{local} et Q^i_{collab} qui définissent les critères sur respectivement les résultats locaux du clustering ainsi que sur l'état du concensus entre les vues. C^i_j définit la dissimilarité entre les vues V_i et V_j. La pondération entre le critère local et les différentes mesures de similiraités est assurée par l'ensemble de coefficients α_i et β^i_j.

À noter que les définitions des critères Q_{local} et Q_{collab} sont propres soit à l'algorithme de clustering local utilisé dans chaque vue [4], soit à la définition

même du problème de clustering [8]. Dans les deux cas, le critère doit être redéfinit pour chaque problème.

2.1 Optimisation des poids pour du clustering collaboratif basé sur des cartes auto adaptatrices

Nos contributions presentées dans cette section sont les suivantes :

— Nous proposons une méthode d'optimisation automatisée et non-supervisée afin d'ajuster la valeur des coefficients définissant l'importance que les vues doivent mutuellement s'accorder lors de leur apprentissage collaboratif.

— Nous démontrons expérimentalement que notre méthode d'optimisation est capable de détecter les vues bruitées qui auraient pu détériorer les apprentissages finaux.

— Nous fournissons les propriétés théoriques de notre méthode. En particulier, nous montrons que notre méthode d'optimisation définit un méta-clustering sur les vues, en les regroupant suivant leurs similarités.

La définition de notre méthode d'optimisation s'est faite grâce à l'ajout de deux contraintes au problème initial. La première a été que, quelque soit la vue V_i, la valeur associée à Q^i_{local} devrait être égale à $\alpha_i = 1$. Cette contrainte traduit le fait qu'en divisant l'ensemble des β^i_j par α_i dans l'équation 1, on obtient un ensemble de coefficients se trouvant uniquement sur les collaborations. Le but des α et des β était d'établir une pondération *relative* des uns par rapport aux autres. L'aspect relatif de cette pondération nous permet de fixer artificiellement la valeur de l'un des coefficient (α_i) à 1.

La second contrainte a été posée sur l'ensemble des β :

$$\forall j \quad \prod_{j \neq i}^{N} \beta^i_j = 1, \quad \forall (i,j) \quad \beta^i_j > 0 \tag{3}$$

Ce type de contrainte a déjà pu être rencontrée dans des travaux relatifs on clustering multi-vues [2]. De plus, il a été montré dans [14] que la contrainte de prime abord plus intuitive $\sum_{j \neq i}^{N} \beta^i_j = 1$ mène à des résultat non satisfaisants et qu'un paramètre supplémentaire p devait ê défini est appris afin de parvenir à un résultat exploitable.

Le problème d'optimisation obtenu étant maintenant sous contrainte, nous avons utilisé la méthode de Karush-Kuhn-Tucker afin de déterminer les valeurs optimales des coefficients β. Pour tout $j \neq i$, nous obtenons :

$$\beta^i_j = \frac{(\prod_{k \neq j} C^i_k)^{\frac{1}{N-1}}}{C^i_j} \tag{4}$$

Si l'on essaye d'interpréter ce résultat, on constate que pour une vue donnée, notre méthode octroie plus d'importance aux vues qui ont des coefficients de dissimilarités C^i_j faible, avec des valeurs de $\beta > 0$ si la dissimilarité est inférieure à la moyenne géométrique des similarités avec les autre vues, et des valeurs de $\beta < 0$ dans le cas contraire. Notre méthode définit donc l'importance d'une collaboration suivant la similarité des résultats obtenus pour chaque vue. Ce point peut se comprendre intuitivement : si l'on cherche à obtenir le meilleur

score de concensus possible, il faut privilégier les collaborations de vues similaires et limiter les collaborations de vue en désaccord.

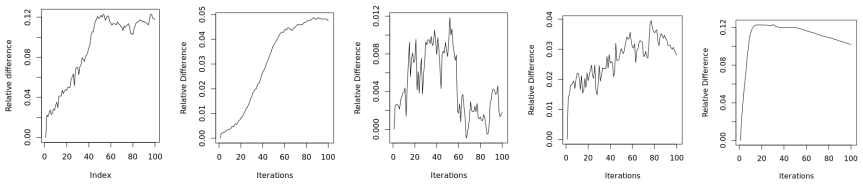
Notre méthode d'optimisation s'incrit dans le cadre d'un apprentissage collaboratif standard. L'algorithme détaillé peut être trouvé dans l'Algorithme 1.

Algorithm 1: Algorithme topologique de collaboration horizontale

Initialisation : Initialiser toutes les cartes de prototypes W aléatoirement.
Étape locale : Initialisation des cartes
forall *Vue* i **do**
  Minimize the objective function of the classical SOM Minimiser la fonction objectif des cartes auto-adaptatrices standards.
end
Étape collaborative :
forall *Vue* i **do**
  For w fixé, calculer : β en à l'aide de l'Équation 4 Mettre à jour les prototypes de toutes les cartes : $w^* = \text{argmin}_w \mathcal{C}(w, \alpha, \beta)$
end

La méthode précédente a été testée sur plusieurs jeux de données de tailles et de complexités variées : WDBC, Waveform, Spambase, Isolet et VHR Strasbourg. La comparaison avec une méthode sans adaptation de poids a été effectuée afin d'attester de l'efficacité de notre méthode. Cette comparaison s'est faite en étudiant la différence relative entre les critères 1 respectifs des deux méthodes.

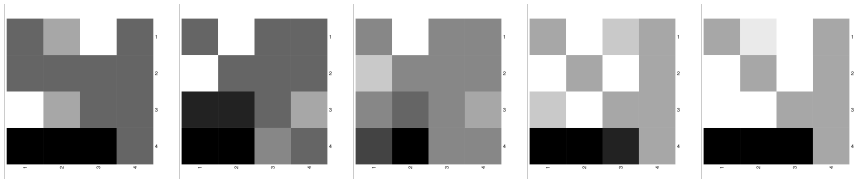
Les résultats (Figure 1) mettent en avant que la différences est toujours positive, démontrant que le critère pour la méthode avec optimisation des β améliore le score du modèle (pour rappel, plus un score est faible, plus les dissimilarités sont faibles, et plus on est proche du concensus). Les valeurs des β sont présentées graphiquement sur la figure 2.



(a) WDBC (b) Waveform (c) Spambase (d) Isolet (e) VHR Strasbourg

FIGURE 1 – Différences relatives des critères pondérés avec et sans optimisation des β tout au long du processus d'apprentissage

Ces cartes font clairement apparaître l'identification des vues bruitées par notre méthode. Tandis que toutes les vues arrivent à identifier les vues bruitées afin de ne pas prendre en compte leurs résultats, les vues bruitées considèrent les vues non bruitées indépendamment de leurs résultats. On peut aussi noter que les vues non bruitées ne coopèrent pas toujours entre elles. Ainsi pour WDBC,



(a) WDBC (b) Waveform (c) Spambase (d) Isolet (e) VHR Strasbourg

FIGURE 2 – Heatmap of the β matrices for each dataset. Colors go from white (strong collaboration) to black (weak collaboration). The gray color on the diagonal stands for $\beta = 1$.

les vues 1 et 3 collaborent exclusivement entre elles, tandis que la vue 2 tire son information des deux vues précédentes.

De plus, comme on peut le voir pour Waveform, les vues similaires ont tendance à se regrouper entre elles. Les deux vues bruitées collaborent exclusivement entre elles, de même que les deux vues non bruitées.

En conclusion, notre méthode permet d'adapter dynamiquement les communications inter-vues pour du clustering collaboratif à l'aide de coefficients scalaires représentant l'importance qu'une vue accorde à l'information d'une de ses paires. L'efficacité de la méthode ainsi que sa capacité à regrouper les vues similaires sont démontrées par les expériences. Dans la section suivante, nous présentons les résultats obtenus sur l'adaptation du clustering collaboratif afin de permettre son apprentissage en ligne. Cet axe a été étudié afin d'explorer l'impact qu'aurait un tel contexte sur les communications inter-vues.

2.2 Cartes auto adaptatrices incrémentales appliquées au clustering collaboratif

Dans cette section, nous présentons les contributions suivantes :

— La définition d'une méthode permettant d'apprendre des Cartes Auto Adaptatrices (CAA) en ligne (au cours du temps).

— L'adaptation d'une méthode de clustering collaboratif permettant de tenir compte des modifications apportées aux Cartes Auto Adaptatrices.

— Le développement et la présentations de résultats empiriques montrant l'efficacité de notre méthode.

L'adaptation du clustering collaboratif à l'apprentissage en ligne a nécessité l'adaptation du modèle utilisé localement pour obtenir une première version des clusterings locaux. Nous avons choisi les Cartes Auto Adaptatrices car elles constituent un modèle couramment rencontré dans la littérature sur le clustering collaboratif [5, 4, 11].

Bien que plusieurs versions en ligne des Cartes Auto Adaptatrices ont été proposées dans la littérature [3, 9], toutes se basent sur des modifications topologiques des cartes originales afin de les adapter à l'évolution des données. Ce type de changement n'est pas permi initialement par le clustering collaboratif, du fait des comparaisons qui sont susceptibles d'être faites neurones à neurones entre les cartes. Plutôt que d'adapter les règles du clustering collaboratif afin de permettre ce genre de modifications, nous avons choisi de définir une nouvelle

version en ligne de ces cartes pour ensuite l'adapter au clustering collaboratif.

La modification de ces cartes se base sur la modification de la fonction de température permettant de définir le voisinage influencé par la modification de chaque neurone. Cette fonction est normalement dépendante du temps, comme présenté dans la formule suivante :

$$\lambda(t) = \lambda_{\max}\Big(\frac{\lambda_{\min}}{\lambda_{\max}}\Big)^{\frac{1}{t}} \tag{5}$$

avec $\lambda_{\max}$ et $\lambda_{\min}$ deux constantes définissant respectivement les températures initiale et finale du modèle. Lorsque la carte est dite "chaude", la modification d'un neurone va impacter un large voisinage, c'est l'étape initiale durant laquelle la carte s'adapte grossièrement aux données. Plus l'apprentissage va avancé, plus la carte va se "refroidir", pour arriver à de petites valeurs de λ. Durant cette phase, la carte adaptera plus localement l'emplacement de ces neurones. L'avantage de cette méthode par rapport à une méthode telle que K-means est que l'on conserve une dimension topologique entre les clusters, alors que les centroids de K-means sont indépendants les uns par rapport aux autres.

Afin de s'affranchir de la dépendance temporelle de la fonction de température et afin de la rendre réactive aux éventuels changement dans la distribution des données, nous avons défini la fonction de température $\widetilde{\lambda}$ suivante :

$$\widetilde{\lambda}(B, W) = \frac{1}{N_{batch}} \sum_{i=1}^{N_{batch}} \|x_i - \omega_{\chi(x_i)}\|_2 \tag{6}$$

avec B le batch des N_{batch} dernières données arrivées, W l'ensemble des neurones de la carte et χ la fonction qui a un point associe l'indice du neurone ω le plus proche de la carte. Cette fonction présente le double avantage de ne pas être dépendante du temps tout en s'adaptant à l'état actuel des données : si en moyenne les données sont loins de la carte, la température sera élevée car l'ensemble de la carte aura besoin d'être adaptée. À l'inverse, si les données sont proches de leurs neurones respectifs, la température sera faible car seules des modifications locales des neurones seront nécessaires.

Cette modification a été inclue dans les équations régissant le comportement du clustering collaboratif afin de le rendre utilisable en ligne. Dans un soucis de concision, le détail des formules n'est pas précisé ici.

Afin d'attester de l'efficacité de notre méthode, nous avons effectuer des apprentissages sur plusieurs jeux de données : Spam base, Waveform, WDBC et Isolet. Pour chaque jeu de donnée, nous avons regardé quelle était l'erreur de quantification moyenne par vue avec et sans utilisation du clustering collaboratif online. L'erreur de quantification est définie par l'erreur quadratique moyenne entre les individus du batch et leurs neurones les plus proches.

Les résultats obtenus sont présentés dans le tableau 1.

Ces résultats indiquent que pour toutes les bases de données sauf Isolet, la carte auto adaptatrice en ligne que nous avons proposée obtient des scores avoisinnant ceux de la version avec clustering collaboratif. C'est un point utile car l'utilisation du clustering collaboratif peut éventuellement réduirel a qualité des résultats obtenus localement du fait de la recherche d'un concensus global. Pour la cas particulier d'Isolet, les meilleurs résultats pour la méthode collaborative peuvent être expliqués par la limitiation de l'impact des données bruitées (96% des données) grâce au clustering collaboratif.

TABLE 1 – Erreur de quantification moyenne pour chaque base de donnée. Les nombres en gras sont les plus petits pour chaque ligne

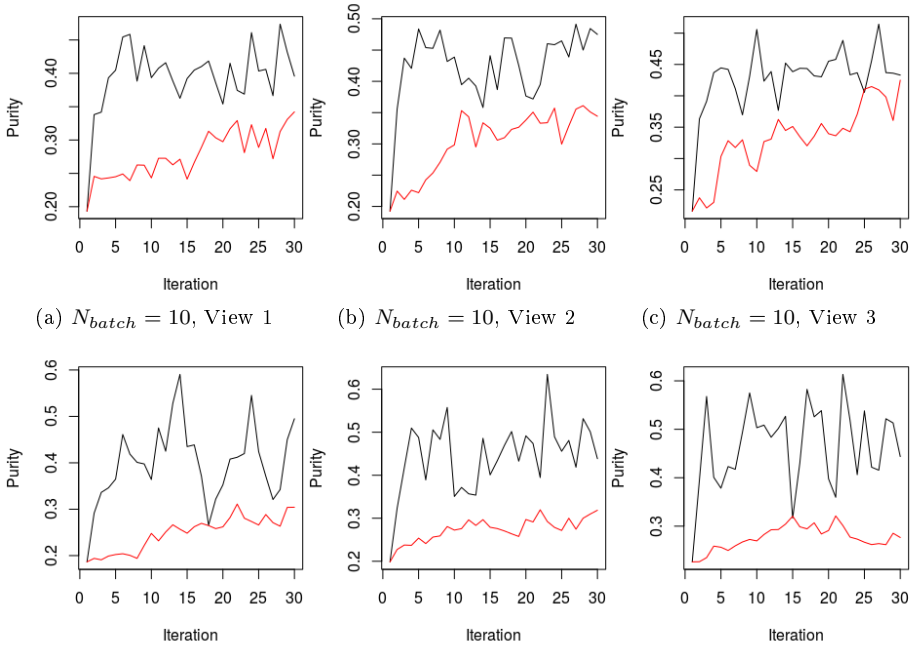
	Vue	CAA Incrémentales	Clustering Collaboratif Incrementale
Spam Base	1	0.31	**0.26**
	2	**0.18**	0.19
	3	0.18	**0.16**
Waveform	1	**0.18**	0.23
	2	**0.17**	0.19
	3	**0.24**	0.30
WDBC	1	**0.19**	0.19
	2	**0.16**	0.19
	3	0.20	**0.16**
Isolet	1	2.15	**1.27**
	2	2.84	**1.38**
	3	2.85	**1.37**

Nous avons de même étudié l'impact de notre méthode sur l'apprentissage au cours du temps d'un modèle collaboratif. Pour se faire, nous avons comparé les valeurs des puretés obtenus d'une part par notre méthode de clustering collaboratif online, et d'autre part par une méthode de clustering classique pour laquelle nous prenions chaque itération comme une unité de temps. Les résultats obtenus sont présentés sur la Figure 3. La pureté d'un neurone est égale à la fraction d'individus qui lui sont rattachés et qui appartiennent à la classe la plus représentée sur ce noeud. Par extension, la pureté d'une carte est égale à la pureté moyenne de ses noeuds.

Ces figures font apparaître une meilleure pureté pour notre méthode par rapport à la méthode classique dans la première phase de l'apprentissage. À l'adaptation en temps réel qui est faite sur la fonction de température, permettant d'obtenir de meilleurs résultats plus rapidement qu'avec une méthode classique. On peut de plus remarquer l'influence du paramère N_{batch} sur l'apprentissage : une valeur plus faible implique une variance plus importante de la pureté au cours du temps. Ce point se comprend intuitivement par le fait que lorsque N_{batch} est faible, le système dispose de peu d'informations pour adapter ses neurones, ce qui implique nécessairement une grande variabilité suivant l'échantillon de données en cours de traitement.

En conclusion, nous avons présenté dans cette section une méthode permettant d'effectuer un apprentissage en ligne des cartes auto adaptatrices sans utiliser de modification topologique. Cette méthode a ensuite été adaptée au clustering collaboratif, puis son efficacité a été présenté sur différents jeux de données. L'influence du nombre de données par échantillon a été étudiée et reliée à la variance des scores obtenus lors de l'apprentissage.

La section suivante présente un use case différent de celui du clustering traité jusqu'à présent. L'objectif principal de cette thèse étant d'explorer les possibi-



(a) $N_{batch} = 10$, View 1 (b) $N_{batch} = 10$, View 2 (c) $N_{batch} = 10$, View 3

(d) $N_{batch} = 3$, View 1 (e) $N_{batch} = 3$, View 2 (f) $N_{batch} = 3$, View 3

FIGURE 3 – Évolution des puretés for le jeu de données Isolet. Les lignes rouges représentes les CAA incrémentales tandis que les lignes noires représentent les CAA collaboratives. Chaque itération correspond à l'arrivée d'une nouvelle donnée

lités offertes par les communications inter-vues dans un cotnexte collaboratif, nous nous sommes intéressé au problème des données manquantes et aux manières d'y pallier.

3 Système de reconstruction collaborative

Les contributions présentées dans cette section sont les suivantes :

— Définition d'un nouveau cas d'utilisation dans un contexte d'apprentissage collaboratif : la reconstruction collaborative.
— Définition d'un modèle pouvant répondre au problème posé.
— Définition d'une nouvelle méthode de pondération permettant de combiner des vecteurs point à point.
— Attestation de l'efficacité du modèle au travers d'expériences menées sur divers jeux de données avec identification des limites du modèle et de pistes d'améliorations.

Le clustering collaboratif se base sur l'hypothèse que les vues communicantes disposent de suffisamment d'individus en commun pour échanger leurs résultats et les comparer. Cependant en pratique, la récupération de plusieurs bases de données sur un même ensemble d'individus est une chose difficile à mettre en place, et la récupération des données peut entraîner l'appartion de données manquantes. L'idée initiale développée dans cette section a été qu'il était possible

d'utiliser l'ensemble des informations disponibles sur le sous ensemble d'individus en commun entre les vues pour inférer les valeurs des individus manquants.

Une représentation de l'architecture de notre méthode est présentée sur la Figure 4. Alors que le clustering collaboratif se base sur une correspondance cluster à cluster, la reconstruction collaborative s'appuie elle sur l'inférence de données cibles à partir de données initiales. La difficulté de la mise en pratique de cette idée a motivé l'utilisation de réseaux de neurones, et plus particulièrement de Perceptrons Multi-Couches (aussi appelés Liens, ou Link en anglais), comme liens entre les vues afin de donner une première approximation de l'individu manquant. L'apprentissage de tels réseaux est possible du fait de la présence d'un ensemble d'individus en commun à une paire de vue. En utilisant les descriptions de la vue externe comme entrée et celles de la vue locale comme sortie, il était possible d'effectuer un apprentissage supervisé du réseau.

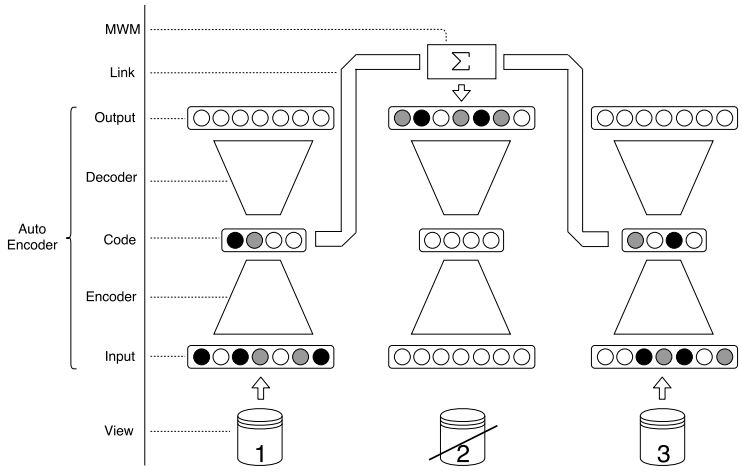


FIGURE 4 – Système de recontruction coopérative. Dans cet example, les vues 1 et 3 envoient leur versions codées de l'individu à la vue 2.

Un autre aspect propre au clustering collaboratif constitue la sécurité qu'il met en place concernant les données en transit : les données originales ne sont jamais transférées d'une vue à une autre. À la place sont transférées soit les identifiants des clusters auxquels appartiennent les individus [13, 12], soit des informations utilisées au cours de l'apprentissage comme c'est le cas pour l'apprentissage à l'aide de cartes SOM [4, 7]. Dans le cas de la reconstruction collaborative, l'utilisation d'un réseau de neurone comme système d'inférence implique nécessairement un codage des données initiales sous la forme d'un vecteur scalaire qui sera utilisé à la place des données orinales comme données d'entrée du réseau en charge de l'inférence. Ce codage sous forme de vecteur est assuré dans notre modèle par un auto-encodeur, une catégorie de réseaux de neurones ayant la particulier de reconstruire en sortie les données fournies en entrée [15].

Enfin, le dernier composant de notre méthode consiste en une méthode de pondération, que nous avons appelé méthode de pondération par masques, qui permet de combiner l'ensemble des approximations créées à partir des données reçues des vues externes ($N-1$ dans un système à N vues où chaque vue externe aurait une information sur l'individu manquant). L'idée fondamentale

de cette méthode consiste en l'utilisation de vecteurs permettant de pondérer des individus point à point (descripteur par descripteur) plutôt que d'utiliser un unique coefficient pondérant l'ensemble de l'individu. En effet, il est facile d'imaginer que pour un individu manquant donné, chaque vue externe permette d'en reconstruire seulement une partie. L'utilisation d'un coeffcient de pondération unique ne permet pas de prendre en compte cette différenciation. Un schéma décrivant le processus de pondération par un ensemble de vecteurs, que nous appelerons désormais masques, est présenté sur la Figure 5. Les valeurs des masques sont entraînées au préalable afin de mieux correspondre à chaque vue, et chaque vue possède $N-1$ masques, un par vue externe.

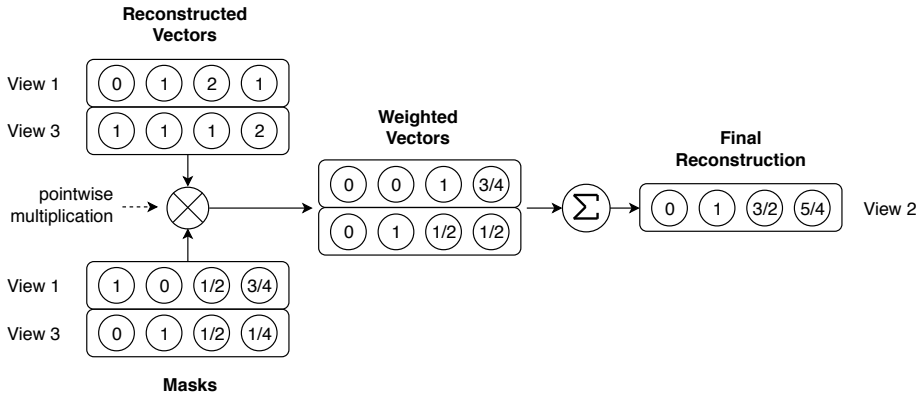


FIGURE 5 – La méthode de pondération par masques. La vue 2 possède les individus inférés à l'aide des informations des vues 1 et 3, et elle utilise ses masques entraînés au préalable afin d'obtenir le résultat pondéré final.

L'idée générale de la méthode d'apprentissage consiste simplement en une descente de gradient sur un critère définit comme la distance entre l'individu cible et sa version reconstruite à l'aide des réseaux de neurones et de notre méthode de pondération. Une seconde méthode itérative a été proposée et démontrée analytiquement en annulant le gradient précedemment obtenu et en mettant à jour les poids des masques de manières itérative. Dans un soucis de concision, le détail des calculs ne sera pas donné ici.

Concernant l'apprentissage du système dans sa globalité, il peut être effectué séquentiellement :

1. Chaque vue entraîne de manière indépendante un auto-encodeur en charge de chiffrer les données originales.
2. Chaque vue encode l'ensemble de sa base de donnée d'apprentissage et envoie le résultat à ses paires
3. Chaque vue entraîne $N-1$ perceptrons multi-couches, 1 par vue externe, en mettant en correspondance les individus chiffrés et les individus cibles.
4. Les vus utilisent leurs perceptrons pour inférer les valeurs des individus présents dans la base d'apprentissage, chaque vue possède alors $N-1$ base de données d'individus inférés.

5. Les poids des masques sont entraînés à l'aide des $N-1$ bases d'individus inférés et des données originales.

Une fois l'ensemble de ces étapes effecutées pour toutes les vues du problème, la reconstruction d'un individu manquant devient possible.

Plusieurs tests ont été effectué à l'aide de notre système de reconstruction collaborative :

— Afin de tester l'efficacité du système global, des reconstruction ont été faites et comparées aux versions originales en utilisant l'erreur quadratique moyenne.
— Le test suivant a consisté en une classification des individus reconstruits à l'aide de l'algorithme de Random Forest pour tester si la classe prédite correspondait à la classe réelle.
— Enfin, la qualité des reconstructions a été testée en remplaçant la méthode de pondération par masque par une simple moyenne. Le but de ce test a été de vérifier l'efficacité de notre nouvelle méthode de pondération.

Ces tests ont été effectués sur 4 jeux de données différents : WDBC, Multi-Features Digital Dataset (MFDD), Madelon et Cube. Ce dernier est un jeu de donnée artificiel crée spécialement pour tester l'efficacité de notre méthode de pondération. Il est constitué d'un ensemble de 4 groupes de points répartis dans un espace en 3 dimensions. Trois clusters se trouvent à chacune des extrémités des vecteurs de base, le quatrième se trouvant à l'origine de l'espace. Trois vues sont ensuite créées en projetant l'ensemble du jeu de données suivant chacun des trois vecteurs de base. L'intérêt d'une vue ainsi créée est que l'information permettant de reconstruire ses individus est par définition répartie dans les deux vues restantes. Si notre méthode de pondération fonctionne, on peut s'attendre à ce que les masques privilégient une caractéristique spécifique tout en rejetant totalement l'autre.

Un schéma décrivant la pondération par masque et comment elle permet de s'affranchir de caractéristiques mal reconstruites et/ou bruitées peut être trouvé sur la Figure 6.

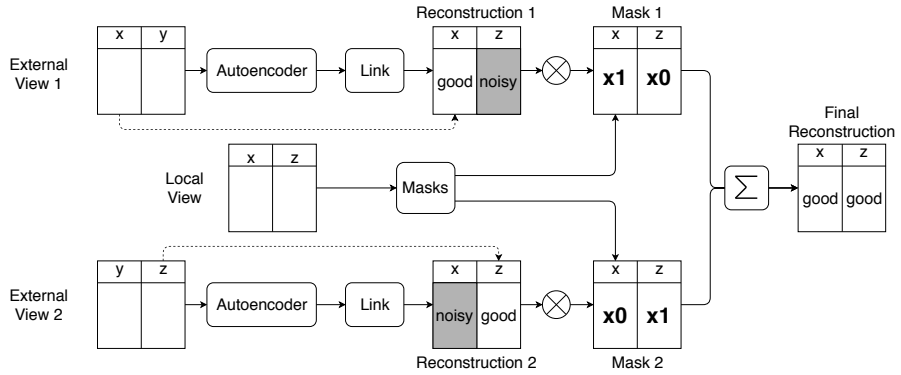
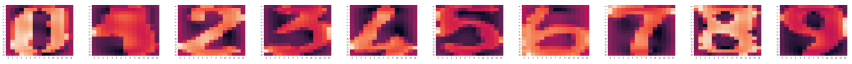


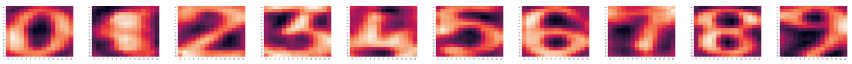
FIGURE 6 – Combinaison de deux reconstructions partiellement bonnes. Dans cet exemple, chaque vue dispose d'assez d'information pour reconstruire seulement une caractéristique sur les deux dans la vue locale (pointillés). La méthode de pondération par masques favorise les parties les mieux reconstruites de chaque résultat, d'où le $\times 0$ et $\times 1$ dans les masques.

L'utilisation du jeu de données MFDD permet d'obtenir des reconstructions visuelles, permettant d'appréhender plus facilement leurs qualités. La figure 7 montre un échantillon de 10 images avant apprentissage. La figure 8 présente 10 reconstructions considérées comme de bonnes qualités. Bien qu'il ne s'agisse que d'un résultat purement visuel, la majorité des individus présente une reconstruction de qualité avoisinante à celle des images présentées ci-dessous. Cependant dans certains cas, le système n'a pas été capable d'inférer efficacement les chiffres à reconstruire, ce qui a mené à des reconstructions comme celles présentées en figure 9.



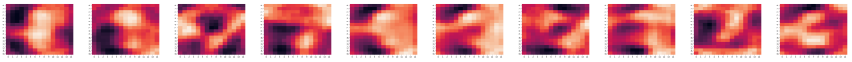
(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

FIGURE 7 – Échantillon des images disponibles dans le jeu de données MFDD.



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

FIGURE 8 – Échantillon d'images bien reconstruites pour le jeu de données MFDD.

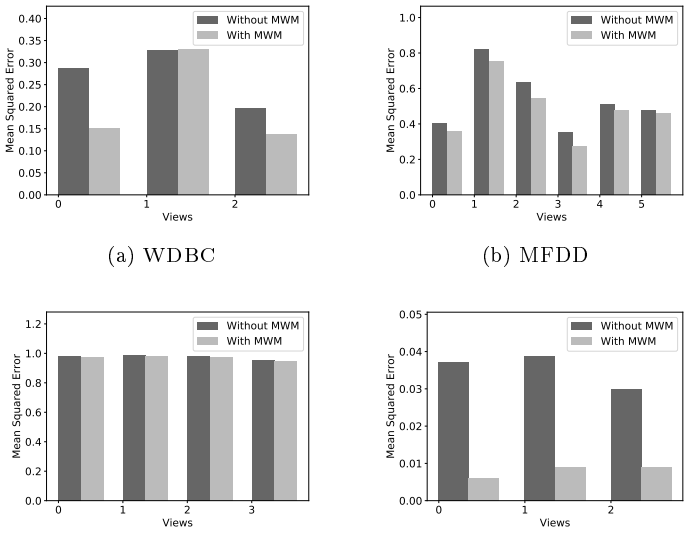


(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

FIGURE 9 – Échantillon d'images mal reconstruites pour le jeu de données MFDD.

Expérimentalement, le système obtient des résultats en terme d'erreur quadratique (relativement à chaque caractéristique) plutôt moyens (Figure 3). Cependant on constate sur un échantillon graphique que la reconstruction globale des individus présente une qualité qui permet de reconnaître l'individu initial. C'est aussi ce que montre les résultats en terme de classification présentés sur la Figure 3. Bien que les reconstructions ne soient pas exactement fidèles aux originales, elles sont suffisamment précises pour obtenir des scores en classification proches de ceux obtenus sur les données originales. En résumé, le système arrive à récupérer certaines informations caractéristiques des individus et à les retranscrire, bien que la qualité de la reconstruction en elle même soit améliorable.

Comme présenté sur la Figure 3 et 3, le test indiquant l'efficacité de la méthode de pondération par masque par rapport à une moyenne indique que notre méthode améliore sensiblement la qualité de reconstruction des individus. De plus, les tests conduits sur Cube montrent clairement que notre méthode arrive à détecter automatiquement quels caractéristiques sont à privilégier suivant la



(a) WDBC (b) MFDD

(c) Madelon (d) Cube

FIGURE 10 – Erreur quadratique moyenne pour tous les jeux de données. Une valeur plus faible correspond à un meilleur résultat.

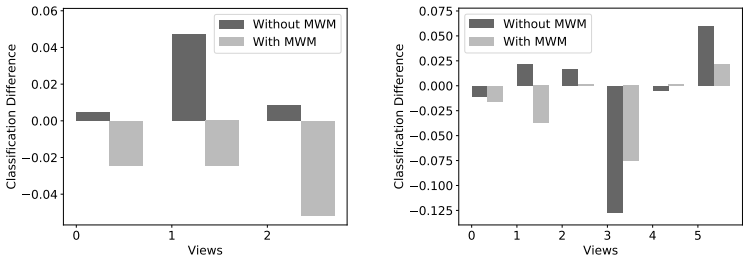
vue externe considérée. En effet, les valeurs des coefficient portant sur la caractéristique que les vues ont en commun avoisine toujours 0.92, tandis que les autres sont proches de 0.14, indiquant un favoritisme fort pour les caractéristiques en commun. Les résultats concernant la classification semblent cependant indiquer que notre méthode n'apporte pas d'amélioration significative de ce point de vue là.

Pour conclure cette section, nous avons défini un nouveau contexte d'apprentissage collaboratif permettant de reconstruire des individus manquant à partir d'informations présentes dans d'autres vues. Une nouvelle méthode de pondération collaborative a été définie et testée. Le système, bien que présentant des résultats améliorables en terme de reconstruction, permet de capturer des informations intrinsèques aux individus telles que leur classe, etl a méthode de pondération arrive à identifier clairement quels caractéristiques sont à privilégier lorsque plusieurs sources sont disponibles pour la reconstruction.

4 Résumé des contributions scientifiques et perspectives

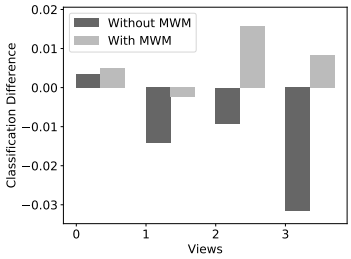
4.1 Contributions au clustering collaboratif

Durant cette thèse, nous avons pu définir une nouvelle méthode permettant d'apprendre automatiquement les coefficient définissant l'importance qu'une vue doit accordée à l'information fournie par une de ses paires. Le fondement théorique de cette méthode se base sur la définition d'un problème d'optimisation sous contrainte que nous avons résolu à l'aide de la méthode de Karush-Kuhn-

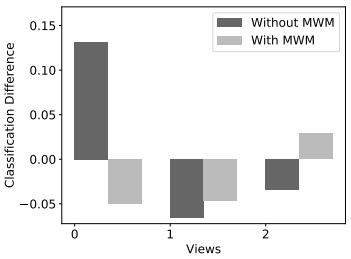


(a) WDBC

(b) MFDD



(c) Madelon



(d) Cube

FIGURE 11 – Scores et Différences de Classification pour WDBC, MFDD et Madelon. Une valeur au dessus de 0 indique que les classifications effectués à l'aide des données reconstruites sont meilleures que celles basées sur les données originales.

Tucker. Notre seconde contribution a consisté en la définition d'une nouvelle méthode permettant d'apprendre des cartes auto adaptatrices au cours du temps. Cette méthode présente l'avantage de ne pas dépendre de modifications topologiques, ce qui a permis son adaptation au clustering collaboratif. Cette méthode se base essentiellement sur la redéfinition de la fonction de châleur à la base des cartes auto adaptatrices pour faire en sorte qu'elle dépende de l'arrivée des nouvelles données plutôt que du temps.

4.2 Contribution à la reconstruction collaborative

L'étude des communications au sein des modèles collaboratifs nous a permis de définir un nouveau contexte d'apprentissage permettant de reconstruire des données manquantes localement à l'aide des informations contenues dans les vues externes. Afin de proposer une première approche, nous avons défini un système basé sur les réseaux de neurones et sur une nouvelle méthode de pondération. Nous avons pu tester son efficacité ainsi que l'efficacité de notre méthode de pondération sur des cas variés montrant que le système était capable de capturer et de reconstruire suffisamment d'informations pour permettre l'indentification graphique et le classement des individus reconstruits.

4.3 Perspectives

Plusieurs axes de recherches pourraient être développés à court et long terme en se basant sur les travaux effectués lors de cette thèse.

4.3.1 Perspectives à court termes

Concernant l'optimisation des pondérations dans un contexte multi-vues, des extensions possibles pourraient s'appliquer au clustering collaboratif vertical pour lequel les cartes auto adaptatrices doivent gérer des informations décrites par un même ensemble de caractéristiques appliquées à un ensemble d'individus différents. De plus, l'application des méthodes décrites dans cette thèse à des algorithmes de clustering tels que les Generative Topographic Maps pourrait être envisagée dans un premier temps, puis à des méthodes de clustering non topologiques dans un second temps.

Comme présenté dans la section ci-dessus, les résultats obtenus à l'aide du système de reconstruction collaborative sont encore perfectibles du point de vue du détail de reconstruction. Bien que les résultats préliminaires soient encourageant, des recherches approfondies pourraient être menées à court termes en autre sur le lien entre la taille du code utilisé pour le transfert de données et les difficultées rencontrés par les perceptrons multi-couches lors de leur entraînement. De même, l'application de la méthode de pondération par masques pourraient éventuellement être étudiés dans des contextes collaboratifs autres que celui de la reconstruction.

4.3.2 Perspectives à long termes

À plus long terme, l'étude de l'information transférée d'une vue à une autre et l'utilisation qu'il est possible d'en faire pourrait permettre de faire avancer le domaine de l'apprentissage collaboratif. De nos jours, la sécurité des données sont des sujets vivement étudiés qui doivent prendre en considération les différentes utilisations qui sont faites des données, et le contexte collaboratif se prête particulièrement bien à ce genre de réflexions et de recherches. Nos travaux dans le domaine de la reconstruction se basent sur des Auto-encodeurs pour fournir un minimum de sécurité, mais cette solution n'est pas satisfaisante sur le long terme.

Enfin, l'étude théorique d'un framework collaboratif permettant son application à des cas d'utilisations autre que la reconstruction et le clustering pourrait s'avérer utile. L'échange d'informations entre deux vues liées à la fois par des intérêts (obtenir de nouvelles informations pour améliorer les résultats locaux) et par des contraintes (partage de données sensibles) est un concept qui pourrait être utilisé comme base pour de futures recherches.

Références

[1] Antoine Cornuejols, Cédric Wemmert, Pierre Gançarski, and Younès Bennani. Collaborative clustering : Why, when, what and how. *Information Fusion*, 39 :81–95, 2018.

[2] Francisco de Carvalho, Filipe M. de Melo, and Yves Lechevallier. A multi-view relational fuzzy c-medoid vectors clustering algorithm. *Neurocomputing*, 163 :115–123, 2015.

[3] Da Deng and Nikola Kasabov. Esom : An algorithm to evolve self-organizing maps from online data streams. In *Neural Networks*, volume 6, pages 3–8. IEEE, 2000.

[4] Mohamad Ghassany, Nistor Grozavu, and Younès Bennani. Collaborative generative topographic mapping. In *International Conference on Neural Information Processing*, pages 591–598. Springer, 2012.

[5] Nistor Grozavu and Younes Bennani. Topological collaborative clustering. *Australian Journal of Intelligent Information Processing Systems*, 12(2), 2010.

[6] T. Kohonen. Self-organized formation of topologically correct feature maps. *Biol. Cyb.*, 43 :59–69, 1982.

[7] Denis Maurel, Jérémie Sublime, and Sylvain Lefebvre. Incremental self-organizing maps for collaborative clustering. In *International Conference on Neural Information Processing*, pages 497–504. Springer, 2017.

[8] Pierre-Alexandre Murena, Jeremie Sublime, Basarab Matei, and Antoine Cornuéjols. An information theory based approach to multisource clustering. 07 2018.

[9] Andrew P Papliński. Incremental self-organizing map (isom) in categorization of visual objects. In *ICONIP*, pages 125–132. Springer, 2012.

[10] Witold Pedrycz. Collaborative fuzzy clustering. *Pattern Recognition Letters*, 23(14) :1675–1686, 2002.

[11] Parisa Rastin, Guénaël Cabanes, Nistor Grozavu, and Younes Bennani. Collaborative clustering : How to select the optimal collaborators ? In *Computational Intelligence, 2015 IEEE Symposium Series on*, pages 787–794. IEEE, 2015.

[12] Jérémie Sublime, Nistor Grozavu, Younès Bennani, and Antoine Cornuéjols. Collaborative clustering with heterogeneous algorithms. In *2015 International Joint Conference on Neural Networks, IJCNN 2015, Killarney, Ireland, July 12-18, 2015*, 2015.

[13] Jérémie Sublime, Nistor Grozavu, Guénaêl Cabanes, Younès Bennani, and Antoine Cornuéjols. Collaborative learning using topographic maps. In *AAFD and SFC'16 Conférence Internationale Francophone" Science des données. Défis Mathématiques et algorithmiques"*, page np, 2016.

[14] Jérémie Sublime, Basarab Matei, and Pierre-Alexandre Murena. Analysis of the influence of diversity in collaborative and multi-view clustering. In *2017 International Joint Conference on Neural Networks, IJCNN 2017, Anchorage, USA, May 14-19, 2017*, 2017.

[15] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103. ACM, 2008.