Apprentissage incrémental liant réseaux de neurones convolutifs pré-entrainés et mémoires associatives binaires

Ghouthi BOUKLI HACENE, Vincent GRIPON, Nicolas FARRUGIA, Matthieu ARZEL, Michel JEZEQUEL

IMT Atlantique
655 Avenue du Technopôle, 29280 Plouzané, France
prenom.nom@imt-atlantique.fr

Résumé – Grâce à leurs capacités à absorber de grandes quantités de données, les réseaux de neurones convolutifs obtiennent les meilleures performances dans de nombreux domaines de vision par ordinateur, se comparant parfois même à la vision biologique. Ils s'appuient généralement sur des méthodes d'optimisation nécessitant une grande puissance de calcul, ce qui lève des problèmes pour leur implémentation sur des systèmes embarqués. Nous nous intéressons au problème de l'apprentissage incrémental, où le système s'adapte en fonction des nouveaux exemples et des nouvelles catégories arrivant au fil de l'eau. Pour y répondre, nous combinons des réseaux de neurones convolutifs pré-entrainés sur de grandes bases de données génériques avec des mémoires associatives binaires, ces dernières permettant l'apprentissage en un seul coup. Pour relier les deux systèmes, nous utilisons un produit de quantification échantillonnant aléatoirement les données manipulées. L'architecture obtenue nécessite une puissance de calcul et une occupation mémoire considérablement réduites en comparaison avec d'autres méthodes. Nous montrons par ailleurs que l'architecture proposée peut apprendre en n'utilisant qu'une fraction des données d'entraînement tout en conservant une précision comparable à l'état de l'art.

Abstract – Thanks to their ability to absorb large amounts of data, Convolutional Neural Networks (CNNs) have become the state-of-the-art in various vision challenges, sometimes even on par with biological vision. CNNs rely on optimisation routines that typically require intensive computational power, thus the question of implementing CNNs on embedded architectures is a very active field of research. Of particular interest is the problem of incremental learning, where the device adapts to new observations or classes. To tackle this challenging problem, we propose to combine pre-trained CNNs with Binary Associative Memories, using product random sampling as an intermediate between the two methods. The obtained architecture requires significantly less computational power and memory usage than existing counterparts. Moreover, using various challenging vision datasets we show that the proposed architecture is able to perform one-shot learning **–** even using only part of the dataset **–**, while keeping very good accuracy.

1 Introduction

En quelques années, les réseaux de neurones profonds ont atteint les meilleures performances [1, 2, 3] dans plusieurs domaines de la vision par ordinateur [4]. Ils se composent d'un grand nombre de paramètres (parfois des milliards) entraînés grâce à de grandes base de données. Ils requièrent donc un calcul intensif et une grande occupation mémoire durant la phase d'apprentissage. Cette limitation devient critique dans le contexte des systèmes embarqués, comme les téléphones portables ou les circuits intégrés reconfigurables.

Dans le contexte des systèmes embarqués, la notion d'apprentissage incrémental prend tout son sens. Il s'agit d'une méthode permettant à un modèle d'apprendre les données de façon séquentielle, utilisant à chaque étape des sous-ensembles de la base de données. Plus précisément, une approche incrémentale peut se définir par [5, 6] : a) la capacité d'apprendre des informations supplémentaires à partir de nouvelles données (incrément par les exemples), b) l'absence du besoin de stocker ou de réutiliser les données originales qui ont servi à entraîner les classifieurs (afin de limiter l'occupation mémoire), c) la préservation des connaissances préalablement acquises (éviter l'oubli catastrophique) et d) la capacité de gérer de nouvelles catégories qui peuvent être introduites avec de nouvelles données (incrément par les catégories).

Certaines méthodes d'apprentissage incrémental ont été proposées. Les auteurs de [6, 7] proposent d'ajouter de nouveaux classifieurs pour traiter les nouvelles données, au risque de se retrouver avec un très grand nombre d'entre eux. Dans [8, 9], les auteurs s'appuient sur des machines à vecteurs de support qu'il est nécessaire de réentraîner lors de l'acquisition de nouvelles données, générant de l'oubli catastrophique [10, 11]. Afin de répondre à ces deux problèmes, une combinaison de machines à vecteurs de support avec l'algorithme *learn++* a été proposée [12, 13]. Cette combinaison offre des performances prometteuses [13]. Cependant, elle requiert l'entraînement systématique d'un classifieur s'appuyant sur les nouvelles et anciennes données, et certaines informations sont oubliées alors que de nouvelles sont apprises.

Nous proposons dans ce papier une méthode d'apprentissage incrémentale présentant les caractéristiques suivantes : a) il est possible d'adapter le modèle à de nouvelles données sans le

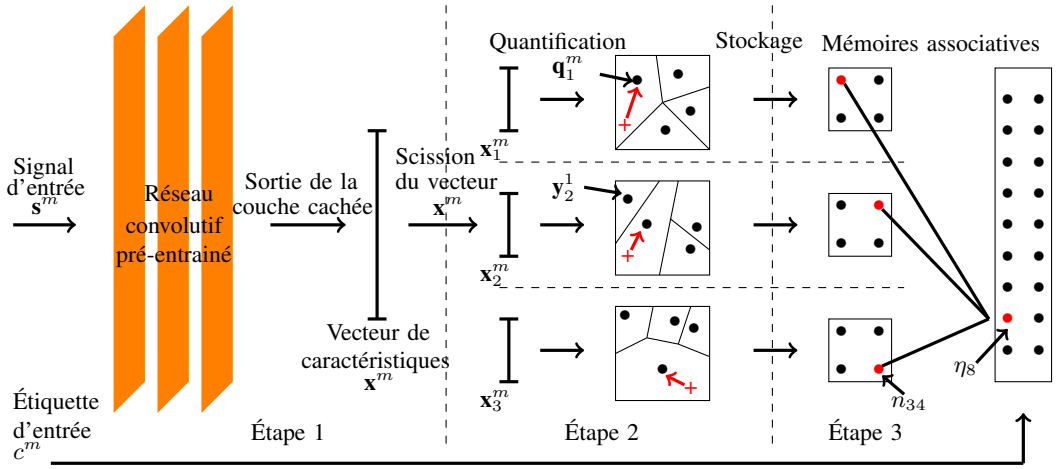

FIGURE 1 – Aperçu de la méthode d'apprentissage incrémental proposée. Elle se compose de trois étapes principales : Étape 1) étant donné un ensemble d'échantillons, un réseau convolutif pré-entrainé est utilisé pour en extraire des caractéristiques. Étape 2) une technique de produit de quantification permet de transformer les vecteurs de caractéristiques en des mots de taille fixe sur un alphabet fini. Étape 3) des mémoires associatives binaires stockent puis classifient les données quantifiées.

réentraîner, b) la méthode proposée requiert une puissance de calcul considérablement inférieure à celle des méthodes existantes, c) la méthode proposée offre une précision comparable à celle de l'état de l'art sur des ensembles de données de vision par ordinateur (CIFAR10, ImageNet), d) la méthode proposée réduit considérablement l'occupation mémoire (de plusieurs ordres de grandeur) par rapport à la recherche exhaustive du plus proche voisin, et finalement e) la méthode ne nécessite qu'une fraction des exemples pour l'apprentissage. Une méthode de plus en plus populaire pour exploiter les performances des réseaux de neurones profonds sans avoir à les entraîner consiste à exploiter un tel réseau spécialisé sur une base de données générique de grande taille, en le considérant comme un extracteur de caractéristiques. Cette méthode est qualifiée dans la littérature de *transfer learning* [14, 2].

Nous proposons de combiner *transfer learning* avec des mémoires associatives binaires. Ces dernières sont des dispositifs capable d'apprendre les exemples un par un, résultant en une occupation mémoire et des ressources de calcul minimes. Un aperçu de la méthode proposée est représenté dans la figure 1. Dans les sections suivantes, nous évaluons la méthode proposée sur des bases de données de vision par ordinateur (CIFAR10 et ImageNet), et comparons la précision obtenue et les ressources utilisées avec des méthodes standards non incrémentales.

2 Méthode

La méthode proposée repose sur trois idées principales : 1) l'utilisation d'un réseau convolutif pré-entrainé agissant comme un extracteur de caractéristiques, 2) l'utilisation de méthodes de produit de quantification afin de transporter les données sur un alphabet fini, et 3) l'utilisation de mémoires associatives binaires afin de stocker et de classifier les données à la manière de la recherche du plus proche voisin. Ces trois étapes sont détaillées dans les prochains paragraphes.

La première étape consiste à utiliser les couches internes d'un réseau convolutif pré-entrainé [3], agissant comme un extracteur de caractéristiques, pour associer un signal d'entrée $\mathbf{s}^m$ à un vecteur de caractéristiques $\mathbf{x}^m$ (*cf.* figure 1 Étape 1).

Une fois obtenu l'ensemble de vecteurs de caractéristiques $X = \{\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^M\}$, la deuxième étape consiste à encoder chaque $\mathbf{x}^m$ à l'aide d'un alphabet fini. Cette étape est cruciale afin de faire correspondre les sorties de l'Étape 1 aux entrées de l'Étape 3. De nombreuses méthodes existent dans la littérature pour répondre à ce problème. UN produit de quantification [15] est utilisé dans ce cas, consistant à scinder les vecteurs en P sous-vecteurs $\left(\mathbf{x}_p^m\right)_{1\leq p\leq P}$ de taille égale, puis à approcher chacun par une "ancre" la plus proche – une ancre étant un vecteur fixé dans le sous-espace correspondant. Plus précisément, et par souci de complexité de calcul, nous avons opté pour un échantillonnage aléatoire parmi les vecteurs $\left(\mathbf{x}_p^m\right)_{1\leq p\leq P}$ des ancres de chaque sous-espace. Dans la suite de ce document, nous notons K le nombre de vecteurs ancres dans chaque sous-espace et $Y_p = \mathbf{y}_{p1}, ..., \mathbf{y}_{pK}$ les vecteurs ancres à proprement parler. Nous avons donc pour chaque $\mathbf{y}_p^k$ l'existence d'un $\mathbf{x}^m \in X$ tel que $\mathbf{x}_p^m = \mathbf{y}_p^k$.

Une fois l'Étape 2) achevée, chaque vecteur de caractéristiques $\mathbf{x}^m$ est transformé en un mot de taille fixe $\left(\mathbf{q}_p^m\right)_{1\leq p\leq P}$ sur un alphabet fini (l'alphabet constitué par les K ancres choisies aléatoirement) (*cf.* figure 1 Étape 2). L'étape 3) consiste à associer à ces ancres la sortie indicatrice de la catégorie correspondante c^m en utilisant une mémoire associative binaire [16]. L'idée est de représenter chaque ancre par un neurone n_{pk} et chaque catégorie par un neurone η_c, puis de relier ceux correspondant à $\mathbf{q}_p^m$ avec celui de la catégorie c^m (*cf.* figure 1 Étape 3). Le même processus est appliqué pour chaque nouvelle don-

née, ce qui permet un apprentissage incrémental. Nous insistons sur le fait que ce réseau de neurones est entièrement binaire : une connexion existe ou n'existe pas, et son utilisation à plusieurs reprises lors de l'apprentissage ne renforce donc pas sa valeur.

La méthode proposée est une combinaison entre un réseau convolutif pré-entrainé ne changeant pas durant la phase d'apprentissage, et des mémoires associatives binaires modifiées pour chaque nouvel exemple ou catégorie introduits. Cette combinaison permet de gérer les deux approches incrémentales susmentionnées : incrément par exemple et incrément par catégorie [17]. Ces propriétés ne requièrent pas d'information a priori sur la base de donnée utilisée, et ne nécessitent qu'une fraction des exemples sans besoin de réentraîner le modèle ou d'endommager les informations déjà apprises [18].

3 Résultats

Nous avons évalué les performances de la méthode proposée sur trois bases de données issues de CIFAR10 et ImageNet. Plus précisément, pour la base de données ImageNet, nous avons extrait deux ensembles de 10 catégories parmi celles n'ayant pas servi à entraîner le réseau convolutif, appelées ImageNet1 et ImageNet2. Les trois bases obtenues contiennent ainsi 10 catégories et de l'ordre du millier d'exemples pour l'apprentissage. Les figures 2 et 3 présentent l'évolution de la précision en fonction du nombre de catégories présentées de façon incrémentale, et en fonction de la proportion des exemples d'apprentissage utilisés, respectivement. A chaque ajout d'une catégorie, la précision diminue pour enfin converger, alors que l'ajout de nouveux exemples améliore a chaque fois la précision. Dans les deux cas, la précision converge vers la même valeur et qui est la valeur de la précision quand on toute la base de données d'apprentissage est disponible en entrée. Une comparaison en termes de précision, de coût de calcul (complexité) et d'occupation mémoire avec une recherche exhaustive du plus proche voisin et une recherche approximative accélérée sont présentées dans le tableau 1. Enfin, le tableau 3 compare les performances obtenues en fonction des paramètres de l'Étape 2, à savoir le nombre d'échantillons K pour chaque sous-espace et le nombre P de sous-espaces. Un réseau de neurones entrainé sur toute la base de données en utilisant X comme entrée donne une précision de 89% pour Cifar10 et $96,0\%$ pour ImageNet2.

4 Conclusion

Nous avons introduit une nouvelle méthode d'apprentissage incrémental s'appuyant sur un réseau convolutif pré-entrainé et des mémoires associatives binaires pour la classification d'images. Alors que le réseau convolutif utilise les poids de connexions pour reconnaître les images, la mémoire associative encode l'information dans l'existence de ses connexions. Cette combinaison permet d'apprendre et de reconnaître les données en n'utilisant que peu d'exemples, et en offrant une occupation

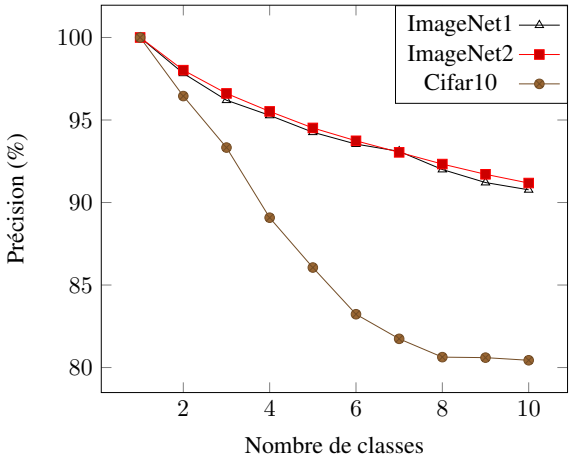

FIGURE 2 – Evolution de la précision de la méthode proposée en fonction du nombre de catégories pour $P = 16$, $K = 20$ (ImageNet1, ImageNet2 et Cifar10).

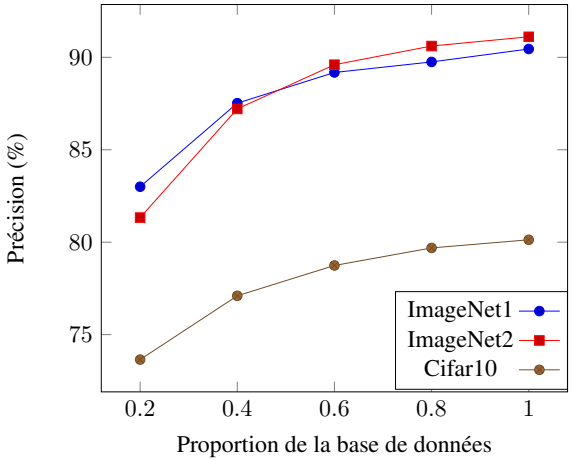

FIGURE 3 – Evolution de la précision en fonction du nombre d'exemples d'apprentissage ($P = 16$ et $K = 200$) (ImageNet1, ImageNet2 et Cifar10).

mémoire ainsi qu'une puissance de calcul réduites par rapport à une recherche du plus proche voisin. La précision fournie par la méthode proposée se compare à celle obtenue par les méthodes de l'état de l'art s'appuyant sur le *transfer learning*. Nous concluons que cette méthode est prometteuse pour les systèmes embarqués et envisageons à l'avenir d'en proposer des implémentations matérielles efficaces.

Références

[1] S. Hong, T. You, S. Kwak, and B. Han, "Online tracking by learning discriminative saliency map with convolutional neural network," *CoRR*, vol. abs/1502.06796, 2015. [Online]. Available : http ://arxiv.org/abs/1502.06796

[2] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*,

TABLE 1 – Précision, complexité et occupation mémoire de la méthode proposée ($P = 64, K = 200$) comparée à une recherche des λ Plus Proches Voisins (λ-PPV) ou des λ Plus Proches Voisins Approximatifs (λ-PPVA) utilisant un produit de quantification avec les mêmes paramètres pour Cifar10. Pour le produit de quantification, un algorithme K-MEANS a été utilisé pour trouver les ancres, les nombres entre parenthèses correspondent à une sélection aléatoire des ancres comme pour notre méthode.

	Méthode proposée	Autres techniques			
		1-PPV	5-PPV	1-PPVA	5-PPVA
Précision(%)	82	85	**87**	82.6(82)	86.07(83)
complexité lors de l'apprentissage	**négligeable**	**négligeable**	**négligeable**	$\geq 2 \cdot 10^{10}$	$\geq 2 \cdot 10^{10}$
complexité lors de la reconnaissance	$\mathbf{4.1 \cdot 10^5}$	10^8	10^8	$3.2 \cdot 10^6$	$3.2 \cdot 10^6$
Occupation mémoire lors de l'apprentissage	$\mathbf{1.3 \cdot 10^7}$	$3.3 \cdot 10^9$	$3.3 \cdot 10^9$	$3.7 \cdot 10^7$	$3.7 \cdot 10^7$
Occupation mémoire lors de la reconnaissance	$\mathbf{1.3 \cdot 10^7}$	$3.3 \cdot 10^9$	$3.3 \cdot 10^9$	$3.7 \cdot 10^7$	$3.7 \cdot 10^7$

Comparaison de la précision obtenue par la méthode proposée sur la base de données CIFAR10 en fonction des paramètres P et K.

	Précision(%)				
	$P = 1$	$P = 8$	$P = 16$	$P = 32$	$P = 64$
$K = 200$	72	78.12	80.62	81	**82**
$K = 150$	70.52	77.35	79.24	80.03	**81**
$K = 100$	67.05	75.63	77.5	78.7	**79.52**
$K = 50$	60	71.15	74.46	76.36	**77.3**

vol. 22, no. 10, pp. 1345–1359, 2010.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[4] C. F. Cadieu, H. Hong, D. L. Yamins, N. Pinto, D. Ardila, E. A. Solomon, N. J. Majaj, and J. J. DiCarlo, "Deep neural networks rival the representation of primate it cortex for core visual object recognition," *PLoS Comput Biol*, vol. 10, no. 12, p. e1003963, 2014.

[5] R. Polikar, L. Udpa, S. S. Udpa, and V. Honavar, "Learn++ : an incremental learning algorithm for multilayer perceptron networks," in *Acoustics, Speech, and Signal Processing. ICASSP'00. Proceedings.IEEE International Conference on*, vol. 6. IEEE, 2000, pp. 3414–3417.

[6] R. Polikar, L. Upda, S. S. Upda, and V. Honavar, "Learn++ : An incremental learning algorithm for supervised neural networks," *IEEE transactions on systems, man, and cybernetics, part C (applications and reviews)*, vol. 31, no. 4, pp. 497–508, 2001.

[7] Y. Sun, K. Tang, L. L. Minku, S. Wang, and X. Yao, "Online ensemble learning of data streams with gradually evolved classes," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 6, pp. 1532–1545, 2016.

[8] N. A. Syed, S. Huan, L. Kah, and K. Sung, "Incremental learning with support vector machines," 1999.

[9] T. Poggio and G. Cauwenberghs, "Incremental and decremental support vector machine learning," *Advances in neural information processing systems*, vol. 13, p. 409, 2001.

[10] N. Kasabov, *Evolving connectionist systems : Methods and applications in bioinformatics, brain study and intelligent machines*. Springer Science & Business Media, 2013.

[11] R. M. French, "Catastrophic forgetting in connectionist networks," *Trends in cognitive sciences*, vol. 3, no. 4, pp. 128–135, 1999.

[12] Z. Erdem, R. Polikar, F. Gurgen, and N. Yumusak, "Ensemble of svms for incremental learning," in *International Workshop on Multiple Classifier Systems*. Springer, 2005, pp. 246–256.

[13] J. F. G. Molina, L. Zheng, M. Sertdemir, D. J. Dinter, S. Schönberg, and M. Rädle, "Incremental learning with svm for multimodal classification of prostatic adenocarcinoma," *PloS one*, vol. 9, no. 4, p. e93600, 2014.

[14] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[15] H. Jegou, M. Douze, and C. Schmid, "Product quantization for nearest neighbor search," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 1, pp. 117–128, 2011.

[16] V. Gripon and C. Berrou, "Sparse neural networks with large learning diversity," *IEEE transactions on neural networks*, vol. 22, no. 7, pp. 1087–1096, 2011.

[17] Z.-H. Zhou and Z.-Q. Chen, "Hybrid decision tree," *Knowledge-based systems*, vol. 15, no. 8, pp. 515–528, 2002.

[18] I. J. Goodfellow, M. Mirza, D. Xiao, A. Courville, and Y. Bengio, "An empirical investigation of catastrophic forgetting in gradient-based neural networks," *arXiv preprint arXiv :1312.6211*, 2013.