Classes adversaires dans l'apprentissage avec peu d'exemples

Raphael LAFARGUE[1,2],Bastien PASDELOUP[1], Jean-Philippe DIGUET[2], Vincent GRIPON[1],

[1] IMT Atlantique, Technopole Brest Iroise 29238 BREST CEDEX, France

[2] IRL Crossing, Adelaide SA 5000, Australie

prénom.nom@imt-atlantique.fr, jean-philippe.diguet@cnrs.fr

Résumé – L'apprentissage avec peu d'exemples est un problème de l'apprentissage automatique où très peu d'exemples étiquetés sont disponibles. Souvent, les connaissances d'un modèle pré-entraîné avec beaucoup de données sur des classes disjointes sont transférées pour répondre au problème. Il est communément admis que, plus ces données de pré-entraînement sont nombreuses et variées, meilleure sera la performance. Au contraire, nous montrons que moins peut faire mieux. Au travers d'expériences et d'une modélisation basée sur le rapport signal à bruit, nous montrons qu'un choix plus judicieux de ces données peut être fait.

Abstract – Few-Shot Learning is a machine learning framework where few labeled data are available. Knowledge learned on abundant data is often transferred to address this challenge. It is commonly accepted that this pretraining data should be large and diverse. We challenge this claim with experiments and a model based on signal-to-noise ratio, and show that a wiser choice of pretraining data can be made.

1 Introduction

L'apprentissage profond est aujourd'hui la méthode incontournable en apprentissage automatique, notamment lorsqu'on traite des signaux naturels complexes, comme des images [1]. Ces modèles reposent sur des architectures comprenant parfois des millions de paramètres ajustables, lesquels sont typiquement optimisés sur de grands jeux de données d'entraînement.

Cette méthode présente un inconvénient évident : il est a priori difficile d'apprendre avec peu d'exemples. En effet, entraîner un modèle contenant un si grand nombre de paramètres avec seulement quelques données aurait irrémédiablement pour effet un sur-apprentissage menant à des performances catastrophiques [2]. Des contextes où peu de données sont disponibles sont toutefois fréquents, notamment dans des domaines où l'acquisition est coûteuse, où leur étiquetage requiert des experts, ou lorsque les événements d'intérêt sont rares. On parle alors d'Apprentissage avec Peu d'Exemples (APE).

Pour pallier ce problème, une méthode couramment employée consiste à pré-entraîner un modèle sur un grand jeu de données *générique*, puis à transférer cette connaissance pour traiter le nouveau problème d'intérêt via un extracteur de caractéristiques (EC) [6]. Même si les jeux de données utilisés sont différents, le fait qu'ils manipulent des signaux similaires permet, au moins en partie, d'outrepasser la limite liée au nombre de signaux étiquetés disponibles : si un EC est entraîné à classifier des races de chiens, il sera sûrement bien adapté pour classifier de nouvelles races de chiens jusqu'ici jamais vues.

On comprend facilement qu'utiliser un jeu de données *générique* qui ne ressemble en rien à celui du problème considéré apporterait peu ou pas d'information utile pour gagner en performance. En revanche, il est communément admis qu'utiliser un jeu de données *générique* gros et diversifié est bénéfique, tant que les signaux ressemblent à ceux que l'on veut traiter [3].

Dans cet article, nous souhaitons montrer que cette conjecture n'est pas nécessairement vérifiée en pratique. Nous proposons un modèle mathématique pour expliquer pourquoi il n'est pas toujours souhaitable d'avoir un EC universel, et exhibons certaines situations suggérant qu'il serait préférable d'utiliser des extracteurs particuliers. Des expériences sur le jeu de données standard Mini-ImageNet appuient nos affirmations. [1]

2 Définitions

Dans le contexte de la recherche en APE, il est fréquent d'utiliser des jeux de données constitués ainsi pour comparer les méthodes :

1. On dispose d'un jeu de données *générique* $\mathcal{G}$, contenant G classes $\{\mathcal{G}^1, \dots, \mathcal{G}^G\}$ avec beaucoup d'exemples. On l'utilise, conjointement à un jeu de données de *validation* contenant d'autres classes, pour entraîner l'EC ;
2. On dispose également d'un jeu de données *nouveau* $\mathcal{N}$ contenant N classes $\{\mathcal{N}^1, \dots, \mathcal{N}^N\}$ n'apparaissant pas dans les jeux de données précédents. Il est utilisé pour construire artificiellement des problèmes d'APE, en sélectionnant au hasard 5 classes nouvelles, et 1 ou 5 exemples par classe. L'ensemble de ces exemples étiquetés est appelé ensemble *support*. Les échantillons à classifier (typiquement 15 par classe) forment l'ensemble *requête*.

Pour entraîner un modèle, nous utilisons une routine classique consistant à minimiser une fonction d'erreur de classi-

1. https ://github.com/RafLaf/Adversarial_classes

fication de type entropie-croisée. Cette optimisation s'effectue sur le jeu de données *générique*. Une fois le modèle entraîné, la dernière couche linéaire de classification est retirée. On obtient alors un EC dont on fixe les poids. Lorsqu'on est amené à traiter le nouveau jeu de données, les signaux sont transformés en vecteurs caractéristiques grâce à cet EC. Cette méthodologie est souvent nommée "transfert d'apprentissage" [6].

3 Méthode

Les comparaisons de méthodes d'APE partent pour la plupart du postulat que le jeu de données *générique* est imposé. En pratique, la question de la pertinence de cette hypothèse se pose, et nous montrons qu'elle peut avoir un impact important sur les performances. Ainsi, pour mener à bien notre étude, nous nous autorisons à retirer une partie du jeu de données *générique* avant d'entraîner notre modèle.

On note f_θ l'EC entraîné sur le jeu de données *générique*. Ainsi, face à un problème d'APE avec des signaux et leurs labels correspondants $\{(\mathbf{x}, y)\}$, la première étape consiste à les transformer en caractéristiques $\{\mathbf{z} = f_\theta(\mathbf{x}), y\}$.

La plupart des méthodes utilisées pour l'APE [4, 5] reposent sur l'utilisation d'un classifieur *plus proche centroïde* (PPC), qui calcule la moyenne par classe des vecteurs dans l'ensemble support, et attribue aux vecteurs dans l'ensemble requête la classe du centroïde le plus proche. Cette méthode suppose que les vecteurs caractéristiques d'une même classe se répartissent autour d'un centroïde dans l'espace des caractéristiques.

De fait, une métrique adaptée pour mesurer la qualité de l'espace des caractéristiques pour répondre à un problème d'APE entre les classes (i,j) est le rapport signal sur bruit (RSB), défini par :

$$RSB(i,j) = \frac{marge}{bruit} = 2\frac{\|\mathbb{E}(\mathcal{N}^i) - \mathbb{E}(\mathcal{N}^j)\|_2}{\sigma(\mathcal{N}^i) + \sigma(\mathcal{N}^j)} \qquad (1)$$

où les espérances et écarts-types sont mesurés dans l'espace des caractéristiques. On nomme $\boldsymbol{\delta} = \mathbb{E}(\mathcal{N}^i) - \mathbb{E}(\mathcal{N}^j)$ la *marge* et $\frac{\sigma(\mathcal{N}^i)+\sigma(\mathcal{N}^j)}{2}$ le *bruit*. Contrairement au cas présent où des problèmes d'APE sont générés artificiellement à partir d'un abondant nouveau jeu de données, dans un cadre applicatif réel d'APE, le RSB n'est pas mesurable car peu de données étiquetées sont disponibles. Il ne peut donc être utilisé pour connaître les classes à retirer. Le RSB a donc simplement un intérêt pour la compréhension des effets du retrait d'une classe *générique*.

L'EC, par construction, est adapté pour reconnaître les caractéristiques associées aux classes du jeu de données *générique*. De manière approximative, on peut donc décomposer tout vecteur $\mathbf{z}$ de l'espace des caractéristiques de la façon suivante :

$$\mathbf{z} \approx \sum_{k=1}^{G} \alpha_k \mathbb{E}(\mathcal{G}^k) \qquad (2)$$

avec α_k la contribution de la classe *générique* $\mathcal{G}^k$ au vecteur, cette contribution étant alignée avec $\mathbb{E}(\mathcal{G}^k)$. Empiriquement, cette décomposition explique 67.8% de la variance de notre jeu de données *nouveau*.

La similarité cosinus paire à paire entre les $\mathbb{E}(\mathcal{G}^k)$ est de -0.02 ± 0.11 avec tout de même un maximum de 0.77. On peut donc, en première approximation, considérer les $\mathbb{E}(\mathcal{G}^k)$ $(\forall k)$ comme orthogonaux ($\perp$). Ainsi, toute contribution d'une classe *générique* peut-être vue comme indépendante des autres. Cette affirmation est basée sur le fait que l'entropie croisée a pour effet d'orthogonaliser les classes *génériques* dans l'espace caractéristique. On peut donc imaginer trois effets que peut avoir le retrait d'une classe *générique* :

1. Si $\mathbb{E}(\mathcal{G}^k) \perp \boldsymbol{\delta}$, le retrait de la classe est *neutre* pour la marge. La marge reste inchangée mais pas le bruit. Cela peut donc diminuer le RSB. Cet effet peut être induit par la présence d'objets parasites non liés à la tâche. Par exemple, la présence d'humains est fréquente pour les photographies de chiens, mais n'est a priori pas liée à la reconnaissance de leur race.
2. Si $\mathbb{E}(\mathcal{G}^k) \not\perp \boldsymbol{\delta}$, alors la classe $\mathcal{G}^k$ aura un effet sur le numérateur du RSB :
   (a) Si les deux classes à discriminer partagent des caractéristiques communes liées à une classe *générique*, alors cette classe serait *adversaire* car son retrait causerait une augmentation de la marge. C'est par exemple le cas si une classe du jeu de données *générique* présente un environnement particulier comme un milieu sous-marin. Dans ce cas, la discrimination dans le jeu de données nouveau de deux classes liées à des environnements sous-marins sera plus difficile ;
   (b) À l'inverse, si cette classe *générique* aide à la discrimination entre $\mathcal{N}^i$ et $\mathcal{N}^j$ en augmentant la marge, alors elle est *bénéfique*. C'est le cas le plus favorable qui est susceptible de se produire lorsque une des classes nouvelles est très similaire à une classe *générique*. Retirer une telle classe diminue la marge.

Pour simplifier, nous présupposons que retirer une classe *générique* d'un jeu de données d'entraînement conserve la géométrie du reste de l'espace caractéristique. L'effet réel est un peu plus complexe, car l'absence d'une classe pendant l'entraînement modifie la géométrie de l'espace latent, alors que retirer une composante a pour effet d'annuler les caractéristiques reliées.

4 Expériences

Pour évaluer ces postulats, nous proposons une étude systématique de l'effet du retrait d'une classe *générique* sur le RSB et la performance d'APE. Le retrait de plus d'une classe est possible mais une exploration exhaustive serait trop longue. Quelques essais montrent que retirer deux classes adversaires donne des effets cumulatifs. Nous utilisons le jeu de données Mini-ImageNet, constitué de $G = 64$ classes *génériques* et $N = 20$ classes nouvelles, chacune contenant 600 exemples.

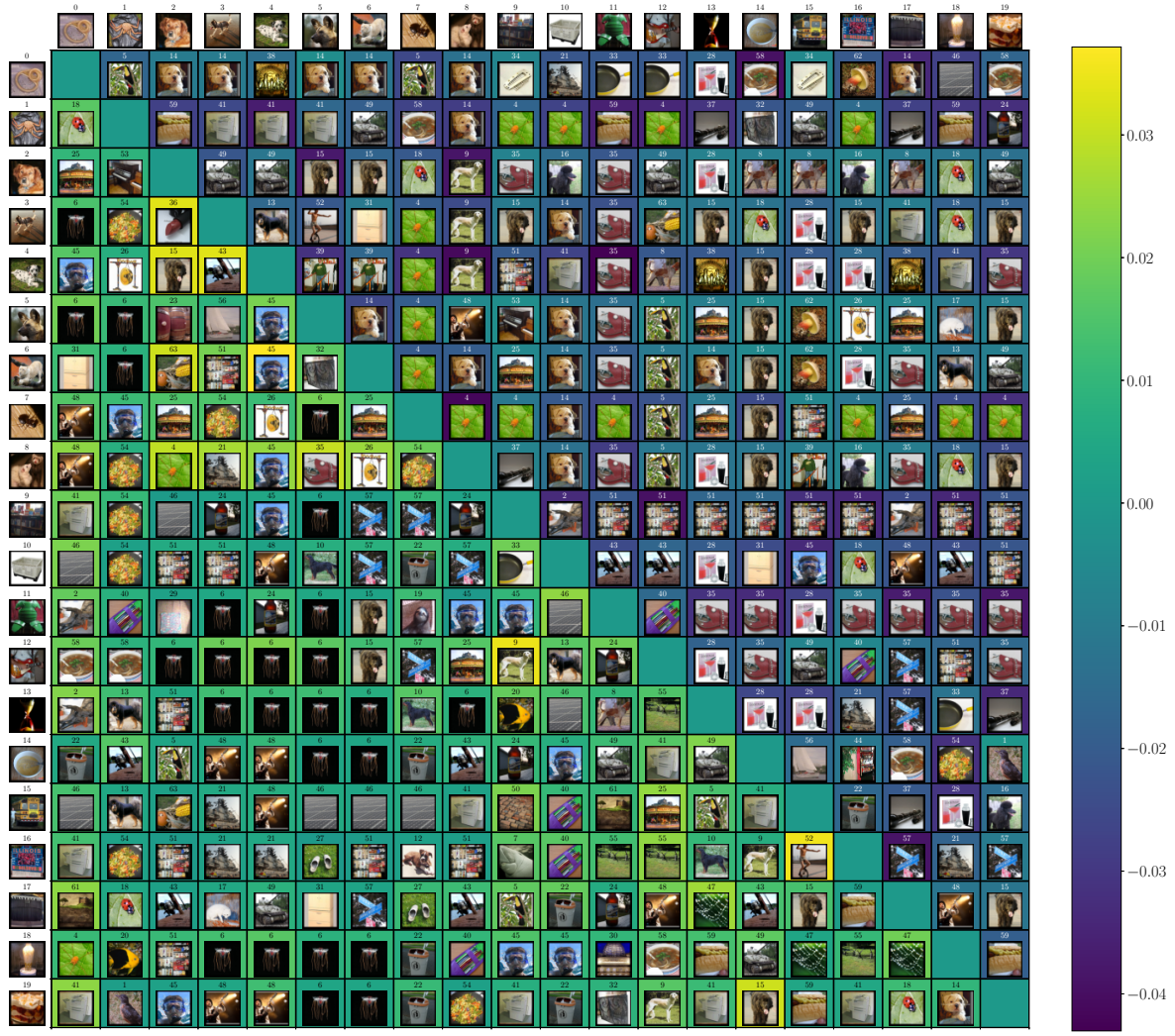

FIGURE 1 – Meilleurs (triangulaire supérieure) et pires (triangulaire inférieure) gains de performance en APE sur deux nouvelles classes par rapport au niveau de l'EC *standard*. Les ECs correspondant à ces gains sont identifiés grâce au numéro de la classe *générique* retirée et d'une image issue de cette classe. Sur les axes sont représentées les 20 nouvelles classes possibles.

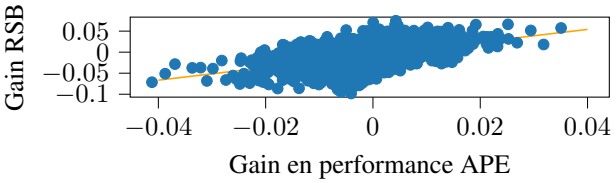

FIGURE 2 – Gain en RSB en fonction du gain en APE pour tout EC pour tout couple de classes *génériques*.

La performance d'APE (ratio d'images bien classifiées) est une mesure directe de l'effet du retrait de données sur les performances de classification. Correctement classifier un échantillon impacte largement la performance sur un problème d'APE. En effet, le nombre de requêtes étant faible, la performance doit être moyennée sur de nombreux problèmes d'APE. *A contrario*, le RSB mesure des effets de manière continue et sur un plus grand nombre d'échantillons.

Pour chaque couple de classes nouvelles, on mesure la performance APE et le RSB. La performance APE est mesurée avec un échantillon étiqueté par classe, et 15 requêtes par classe. La performance retenue est la moyenne des performances sur 20 000 problèmes d'APE créés artificiellement. Le RSB est, lui, obtenu en appliquant la formule (1) sur les 2×600 images que compte chaque couple de classes.

Un EC, dit *standard*, est entraîné sur les 64 classes *génériques* de Mini-ImageNet. On entraîne également 64 ECs, chacun sur 63 classes *génériques* seulement, en retirant une classe *générique* par EC. Chacun de ces 65 ECs repose sur l'architecture ResNet-12 [9] et est entraîné avec les méthodes décrites en [5] sans augmentation par rognage ni méthodes ensemblistes. *In fine*, on obtient pour chaque triplet de classes $(\mathcal{G}^k, \mathcal{N}^i, \mathcal{N}^j)$ un gain en APE et un gain en RSB. Ces deux gains pouvant être positifs ou négatifs. $\mathcal{G}^k$ est la classe générique retirée lors de l'entraînement de l'EC et $\mathcal{N}^i, \mathcal{N}^j$ forment le couple de classes nouvelles du problème d'APE considéré.

5 Résultats

La Figure 2 montre que le RSB est un bon indicateur, significativement corrélé avec la performance. Un modèle de forêt d'arbres de décision aléatoires permet même de prédire le gain avec un facteur de détermination de $R^2 = 55\%$. Ce modèle utilise le RSB et les similarités cosinus entre les vecteurs caractéristiques des triplets $(\mathbb{E}(\mathcal{G}^k), \mathbb{E}(\mathcal{N}^i), \mathbb{E}(\mathcal{N}^j))$. Ces relations géométriques sur les triplets ne permettent pas, à elles seules, de prédire le gain de performance APE avec une grande précision. Les distributions de points associées semblent nécessaires. On note tout de même une faible corrélation (20%) entre les performances APE et la géométrie du triplet. On conclut que pour obtenir un gain positif il faut des classes nouvelles proches et une classe *générique* relativement éloignée du couple. On note que pour un couple de classes *nouvelles* donné, l'écart-type mesuré sur les 20 000 tentatives est de $8.3 \pm 2.7\%$.

Analysons quelques cas représentatifs de gain ou de perte de la Figure 1 au regard du modèle proposé :

1. On note un gain de plus de 3.5% de performance lorsque les classes nouvelles sont "Dalmatien" ($\mathcal{N}^4$) et "Lion" ($\mathcal{N}^6$), et que l'EC est celui dont la classe "Masque-tuba" ($\mathcal{G}^{45}$) est retirée. On aurait donc ici l'effet 2a) décrit en Section 2. L'uniformité de l'arrière-plan des images sous-marines peut expliquer le rapprochement (en présence de $\mathcal{G}^{45}$) de classes dont le fond est aussi uniforme comme un lion dans la savane ou un dalmatien sur une pelouse. Nous observons effectivement une augmentation de la marge de plus de 10% (après retrait) et une très légère diminution du bruit de 0.5%. $\mathcal{G}^{45}$ est adversaire ici ;
2. Un autre gain de performance significatif est obtenu lorsque la classe "Monocycle" ($\mathcal{G}^{52}$) est retirée et que le problème d'APE concerne les classes "Bus scolaire" ($\mathcal{N}^{15}$) et "Tableau d'affichage" ($\mathcal{N}^{16}$). On peut légitimement penser que des routes, voire même des bus, sont présents dans les photographies de monocycles. Des "Tableaux d'affichage" à proximité de route peuvent aussi créer la confusion. Il s'agit ici de l'effet 1) décrit en Section 2. Dans ce cas, le bruit diminue de plus de 1% et la marge augmente d'environ 7 %. L'effet 2a) est ici à l'oeuvre. $\mathcal{G}^{52}$ est adversaire ici ;
3. On note une perte d'environ 4% avec le triplet ("Bartabac" $\mathcal{G}^{51}$, "Librairie" $\mathcal{N}^9$, "Guitare électrique" $\mathcal{N}^{12}$) . On voit ici l'effet 2b) décrit en Section 2, où une classe *générique*, très proche d'une classe nouvelle par similarité cosinus entre vecteurs caractéristiques, participe à la discrimination. Les classes $\mathcal{G}^{51}$ et $\mathcal{N}^9$ sont les deux plus proches tous jeux de données confondus. La marge diminue de plus de 12% tandis que le bruit augmente légèrement, d'environ 0.5%. $\mathcal{G}^{51}$ est bénéfique ici ;
4. Étudions maintenant le RSB. La plus forte augmentation de la marge en valeur absolue concerne le triplet ("Miroir Solaire" $\mathcal{G}^{46}$, "Cuirasse" $\mathcal{N}^{11}$, "Caisse" $\mathcal{N}^{10}$). C'est aussi le meilleur retrait de classe pour maximiser les performances d'APE. Ici, une interprétation visuelle est possible. La classe "Miroir Solaire" présente des surfaces lisses et réfléchissantes, comme les "Cuirasse", mais aussi des formes rectilignes comme les "Caisse". Cette classe rapproche donc les points de $\mathcal{N}^{10}$ et $\mathcal{N}^{11}$ dans l'espace caractéristique grâce à l'effet 2a). $\mathcal{G}^{46}$ est adversaire ici.

Ces expériences montrent donc que certaines classes utilisées pour entraîner l'EC sont potentiellement adversaires pour les taches considérées ensuite, remettant en question l'idée communément admise qu'il existe des "descripteurs universels" atteignables à partir de très grandes banques de données. Au contraire, une connaissance du terrain conduisant à sélectionner quelles classes utiliser pendant l'entraînement peut significativement améliorer les performances.

6 Conclusion

Au travers de trois effets, nous avons montré que certaines classes *génériques* peuvent jouer un rôle "bénéfique" ou "adversaire". Cela indique que pour un problème d'APE donné, le jeu de données *générique* utilisé est d'une grande importance et doit être adapté au problème considéré. Ce travail ouvre de nouvelles perspectives quant au choix du jeu de données *générique* dans le contexte de l'APE voire celui du transfert d'apprentissage.

Références

[1] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature, 521(7553), 436-444.

[2] Ravi, S., & Larochelle, H. (2016). Optimization as a model for few-shot learning.

[3] Triantafillou, Eleni, et al. "Meta-dataset : A dataset of datasets for learning to learn from few examples." arXiv preprint arXiv :1903.03096 (2019).

[4] Wang, Y., Chao, W. L., Weinberger, K. Q., & van der Maaten, L. (2019). Simpleshot : Revisiting nearest-neighbor classification for few-shot learning. arXiv preprint arXiv :1911.04623.

[5] Bendou, Yassir, et al. "EASY : Ensemble Augmented-Shot Y-shaped Learning : State-Of-The-Art Few-Shot Classification with Simple Ingredients." arXiv preprint arXiv :2201.09699 (2022).

[6] Torrey, L., & Shavlik, J. (2010). Transfer learning. In Handbook of research on machine learning applications and trends : algorithms, methods, and techniques (pp. 242-264). IGI global.

[7] Yuan, T., Deng, W., Tang, J., Tang, Y., & Chen, B. (2019). Signal-to-noise ratio : A robust distance metric for deep metric learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 4815-4824).

[8] Dicker, L. H. (2014). Variance estimation in high-dimensional linear models. Biometrika, 101(2), 269-284.

[9] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).