Projet d'Intelligence Artificielle et Multimédia : Fix-Match

Héloïse Lafargue, Julie Remenant, Romain Peyremorte

ENSEEIHT Toulouse, Sciences du Numérique, Février 2024

1 Introduction

Dans le cadre de ce projet, nous avons exploré l'efficacité de la méthode de semi-supervision FixMatch dans l'apprentissage automatique, particulièrement dans les contextes où les données labellisées sont limitées. L'objectif est d'analyser l'impact de la quantité et du choix des données labellisées sur la performance des modèles.

2 Méthode Fix-Match

2.1 Principe, objectifs et fonctionnement de FixMatch

La méthode FixMatch combine deux approches de manière intelligente, à savoir la régularisation de cohérence et le pseudo-etiquetage. La régularisation de cohérence consiste à obtenir des prédictions similaires lorsque le modèle est alimenté avec des images déformées d'une même image. Le pseudo-etiquetage exploite l'idée d'utiliser le modèle lui même pour obtenir des étiquettes artficielles pour les données non étiquetées. L'objectif de cette méthode est d'utiliser un ensemble de données non étiqueté afin de rendre la classification sur l'ensemble de données étiqueté plus facile. La fonction de perte que l'on veut optimiser L se compose de deux parties:

$$L = L_s + \lambda L_u$$

avec $L_s = H(\hat{Y}, Y)$ la fonction perte de la partie supervisée c'est à dire l'entropie croisée entre les étiquettes prédites et les étiquettes réelles, L_u la fonction perte de la partie non supervisée et λ un hyperparamètre scalaire fixe dénotant le poids relatif de la perte non étiquetée.
Voici un shéma expliquant le fonctionnement de FixMatch présenté sur la figure 1:

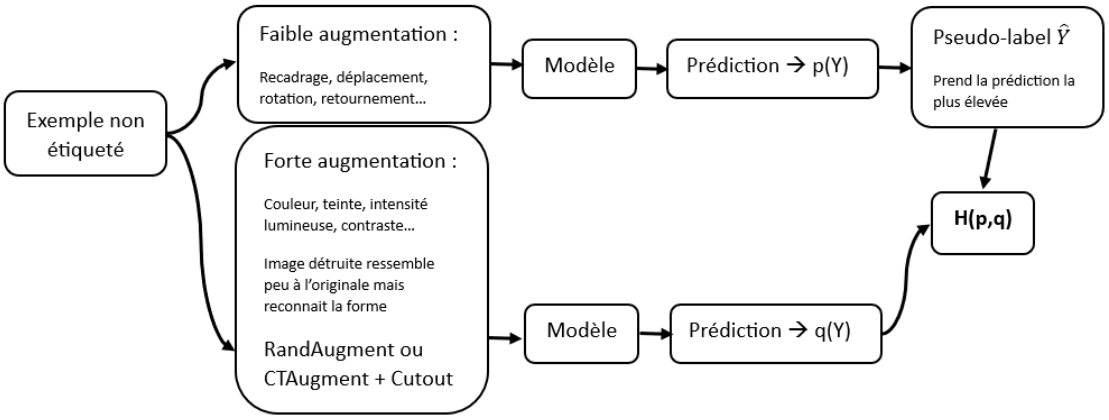

Figure 1: Diagramme fonctionnement FixMatch

Le modèle entrainé de la sorte apprend à ignorer les augmentations comme si aucune distorsion n'avait été appliquée. Pour toutes les images, à la fin, le modèle doit etre capable d'associer la même étiquette à une image fortement et faiblement augmentée issue de la même image de départ. Cette fonction perte L_s est conçue pour que le modèle ne fasse pas de distinction entre les différentes versions d'augmentations d'une même image.

2.2 Influence des différents paramètres

Pour que le modèle soit le plus performant possible, plusieurs paramètres rentrent en jeu:
- La stratégie d'augmentation: il faut à la fois Cutout et CTAugment ou RandAugment pour obtenir le taux d'erreur le plus faible. Dans notre application, nous avons utilisé l'outil *ImageDataGenerator* de la librairie *tensorflow.keras.preprocessing.image* pour réaliser les augmentations.
- L'optimiseur: descente du gradient stochastique avec momentum plus performant que Adam.
- Le ratio de données non étiquetées: plus il est élevé est plus le taux d'erreur diminue.
- La méthode de dégradation des pondérations (ou *weight decay*): le choix d'une valeur supérieure ou inférieure d'un ordre de grandeur à la valeur optimale peut coûter 10% pour les bases de données avec peu d'étiquettes.
- Le seuillage: la précision des pseudo-étiquettes pour les données non étiquetées augmente avec des valeurs de seuil plus élevées.

3 Augmentations

Après avoir réalisé une première implémentation, nous avons cherché les paramètres d'augmentations les plus adaptés. Les augmentations appliquées sont une rotation, un zoom et un décalage (shift) pour l'augmentation forte, et juste un décalage pour l'augmentation faible. Nous avons fait en sorte que le décalage appliqué sur l'augmentation faible soit le même sur l'augmentation forte.

Pour déterminer les paramètres les plus adaptés à notre cas, nous avons initialisé les paramètres à une rotation à 20°, un zoom à 0,3 et un shift à 3,5 pixels, d'après les augmentations réalisées avec Pi-model et les indications du papier (1) (13,5% de 28 pixels donne 3,5). Nous avons ensuite entrainé le modèle FixMatch avec 100 données labellisées sur 2000 epochs en modifiant qu'un seul paramètre et regardé les valeurs de précision du modèle (accuracy).

Modification	Rotation				Zoom					Shift					
Valeur	10	20	30	40	0.1	0.2	0.3	0.4	0.5	0.5	1.5	2	3.5	5	6
Train accuracy	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.783	1.0	1.0	1.0	1.0	0.9771
Test accuracy	0.94	0.95	0.94	0.93	0.93	0.949	0.942	0.948	0.947	0.93	0.966	0.964	0.95	0.94	0.87

Table 1: Résultats tests sur paramètres d'augmentation

Les paramètres que nous jugeons donc adaptés sont : rotation de 20°, zoom de 0,2 et shift de 1,5.

4 Résultats intermédiaires

A partir des augmentations trouvées précédement, nous avons réalisé des entrainements sur le jeu de données MNIST avec un nombre de données déterminé d'images labellisées (100, 50 et 10) couvrant toutes les classes possibles pour comparer le résultat des entrainements du modèle semi-supervisé FixMatch, d'un modèle supervisé où les données d'entraînement n'étaient pas augmentées, d'une modèle supervisé où elles étaient faiblement augmentées et d'un modèle où elles étaient fortement augmentées. Les entraînements des quatre modèles ont été réalisés en parallèle sur les mêmes données, choisies aléatoirement par epoch. Chaque modèle est entrainé sur 10 000 epochs.

4.1 100 données labellisées

Nous avons donc commencé avec 100 données labellisées.

Comme on peut le voir sur la figure 2, les mesures sont plutôt bruitées à cause de l'aléatoire des augmentations et ainsi peu lisibles. Donc par la suite nous afficherons des versions moyennées des résultats sur des plages de 250 epochs pour mieux voir les tendances (comme la figure 3).

Le modèle FixMatch arrive à atteindre un test accuracy maximal de 97,49% durant les 10 000 epochs. On peut observer qu'environ 90% des données non-labellisées sont considérées comme des pseudo-labels avec un faible taux de fausses estimations. On remarque que la loss du modèle diminue encore après 10000 epochs donc on peut supposer que le modèle peut encore atteindre de meilleurs résultats mais on peut voir que le test accuracy n'augmente presque plus.

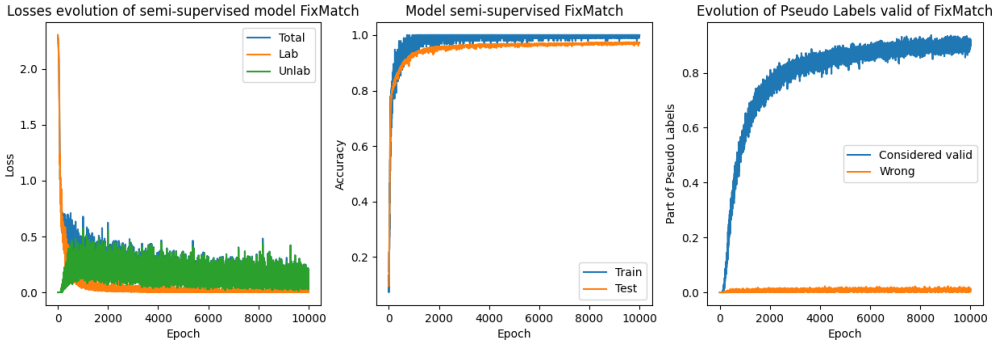

Figure 2: Résultats entrainements avec 100 données labellisées (bruités)

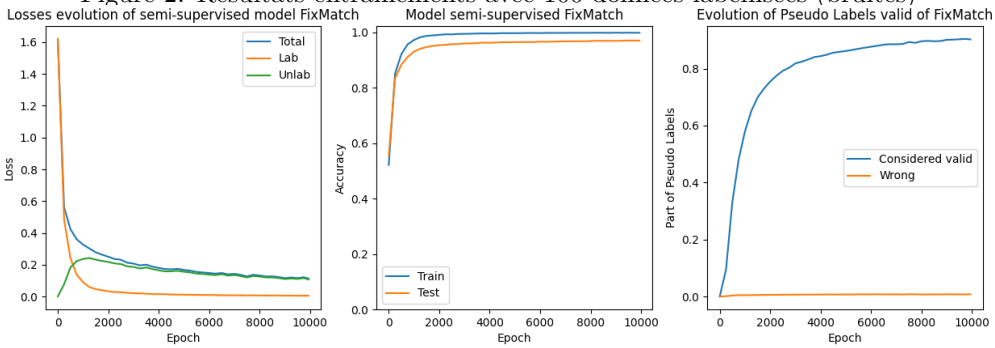

Figure 3: Résultats entrainements avec 100 données labellisées (moyennés)

En comparant sur la figure 4 les résultats des différents modèles semi-supervisés et complétement supervisés, on remarque que le modèle FixMatch atteint un meilleur score maximal de 97,49% par rapport aux autres modèles qui n'atteignent que 93,32% pour le meilleur supervisé qui applique une forte augmentation. Sans augmentation, le modèle supervisé n'atteint que 83,3%, avec même une baisse de performance à 8000 epochs. On remarquera que le FixMatch met plus de temps à atteindre ses meilleurs performances que les modèles supervisés.

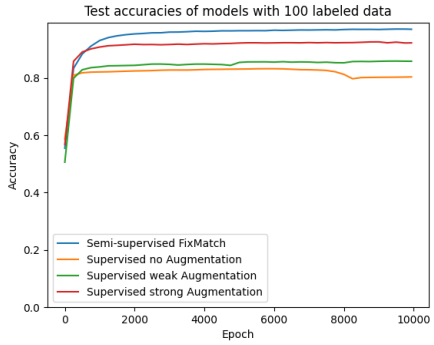

Figure 4: Test accuracy des modèles avec 100 données labellisées

4.2 50 données labellisées

En deuxième test, nous utilisons que 50 données labellisées sur l'ensemble du dataset MNIST.

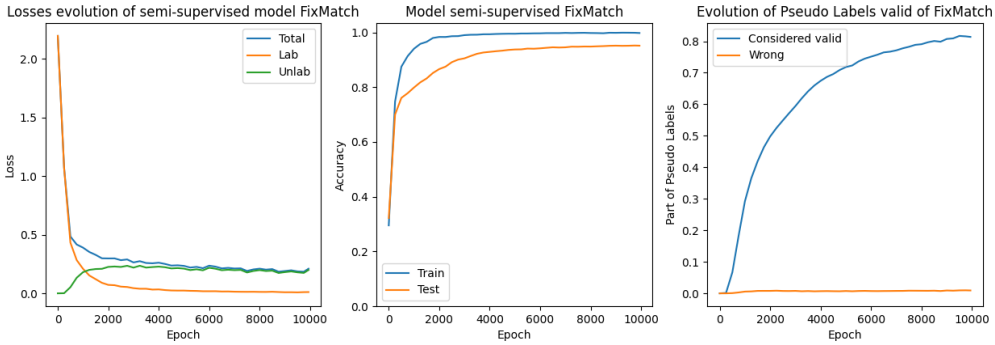

Figure 5: Résultats entrainements avec 50 données labellisées

Avec 50 données labellisées, le modèle FixMatch atteint une accuracy de test de 95,75%, ce qui est

seulement une baisse de 2 points par rapport à 100. Cependant, cette fois-ci le taux de pseudo-labels par rapport aux données non-labellisées n'est que 80% avec la loss qui évolue moins vite en étant plus stable que précédemment.

Les modèles supervisés atteignent des scores bien moindres : comme cela est visible sur la figure 6, le modèle entrainé sans augmentation atteint à peine la barre des 69,93% d'accuracy et celui entrainé avec une faible augmentation la barre des 74,69%. L'augmentation des données d'entrainements montre sont influences avec l'écart entres les modèles supervisées car le modèle avec forte augmentation atteint une accuracy de 87,41%.

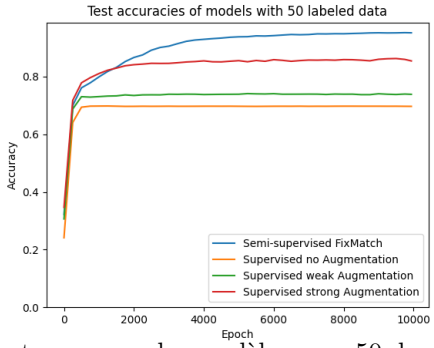

Figure 6: Test accuracy des modèles avec 50 données labellisées

4.3 10 données labellisées

Pour poursuivre la diminution du nombre de données labellisées, nous sommes descendus à 10, c'est-à-dire une de chaque classe.

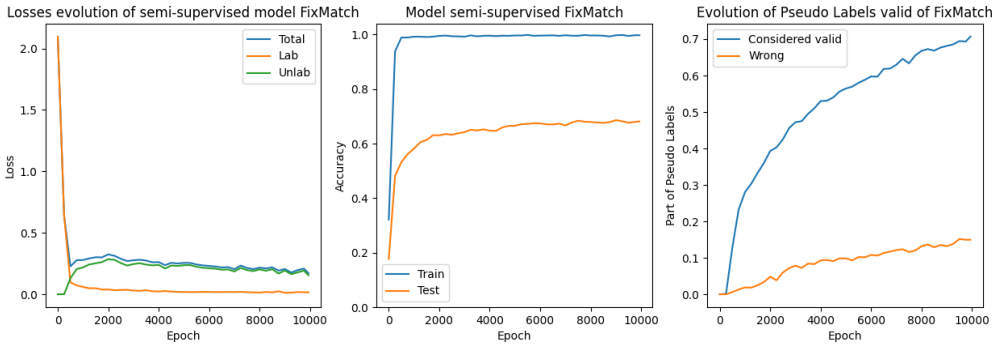

Figure 7: Résultats entrainements avec 10 données labellisées

Avec 10 données labellisées, le modèle FixMatch atteint au maximum une accuracy de 71,15%. Le modèle semble avoir tiré toutes les informations possibles des données labellisées du fait de l'accuracy d'entrainement à 100% et donc le modèle ne peut apprendre que des pseudo-labels. Ces derniers augmentent mais la part d'entre eux qui sont faux augmente aussi (fig.7) : continuer l'entrainement risque de ne pas améliorer le modèle car il apprendrait sur des faux pseudo-labels.

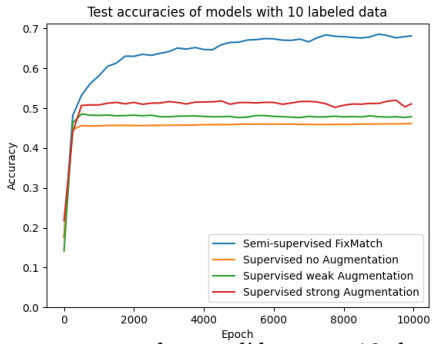

Figure 8: Test accuracy des modèles avec 10 données labellisées

Malgré une forte chute de l'accuracy de FixMatch, ce dernier reste supérieur aux modèles supervisés qui ont des résultats seulement entre 45 et 50%. Le modèle entrainé par forte augmentation reste toujours celui ayant la meilleur accuracy des modèles supervisés.

5 Identification du meilleur ensemble de données labellisées

Dans cette partie, nous cherchons à identifier le meilleur ensemble de données labellisées possible, c'est-à-dire celui qui conduit à la meilleure performance sur l'ensemble de test. Nous avons choisi d'utiliser la mesure de confiance, en vous appuyant sur l'article (2), pour déterminer le meilleur ensemble de données labellisées :

$$\frac{1}{N}\sum_{i=1}^{N}\max f_{\theta_i}(x)$$

Pour chaque modèle i, on prend le score de confiance maximal parmi toutes les classes prédites pour l'exemple x. Cela indique la probabilité avec laquelle le modèle croit en sa prédiction la plus probable. Ensuite, nous effectuons la moyenne de ces scores maximaux sur tous les modèles, ce qui donne un indicateur global de la confiance de l'ensemble dans sa prédiction pour chaque exemple.
Lors de l'entraînement du modèle de base, nous avons limité l'entraînement initial à un petit nombre d'époques (5) avant de procéder à la sélection des meilleures données pour un entraînement plus approfondi. Le but est de minimiser le risque d'obtenir des scores de confiance trop uniformément élevés parmi les prédictions du modèle. Si presque toutes les prédictions sont très confiantes ($score \approx 1$), il devient difficile de distinguer les données qui pourraient réellement bénéficier au modèle en termes d'apprentissage, ce qui fausserait le processus de sélection des meilleures données labellisées. En limitant l'entraînement initial à un petit nombre d'époques, on encourage le modèle à maintenir un certain degré d'incertitude dans ses prédictions. Cela permet de mieux identifier les exemples pour lesquels le modèle est relativement sûr, et donc potentiellement plus représentatifs.

Les résultats obtenus sont présentés dans le tableau 2 et la figure 9 (avec n le nombre de données pour chaque classe). On note des précisions supérieures à 50% même avec seulement 10 données labellisées pour l'entraînement et des pertes assez faibles. Il semble que les augmentations faibles fonctionnent particulièrement bien pour trouver le meilleur jeu de données.

Augmentation	n	Perte (Orig.)	Précision (Orig.)	Perte (Réentr.)	Précision (Réentr.)
Sans Augmentation	1	0.0364	98.94%	1.4639	54.60%
	5	0.0352	98.91%	1.0124	71.93%
	10	0.0352	98.91%	1.1620	74.86%
Weak Augmentation	1	0.0374	98.79%	1.3599	56.15%
	5	0.0309	99.05%	1.0364	74.03%
	10	0.0466	98.73%	0.6838	82.21%
Strong Augmentation	1	0.0447	98.54%	1.6055	50.02%
	5	0.0379	98.75%	0.9449	74.30%
	10	0.0384	98.80%	0.9599	76.05%

Table 2: Résultats de l'évaluation des modèles avec différentes stratégies d'augmentation

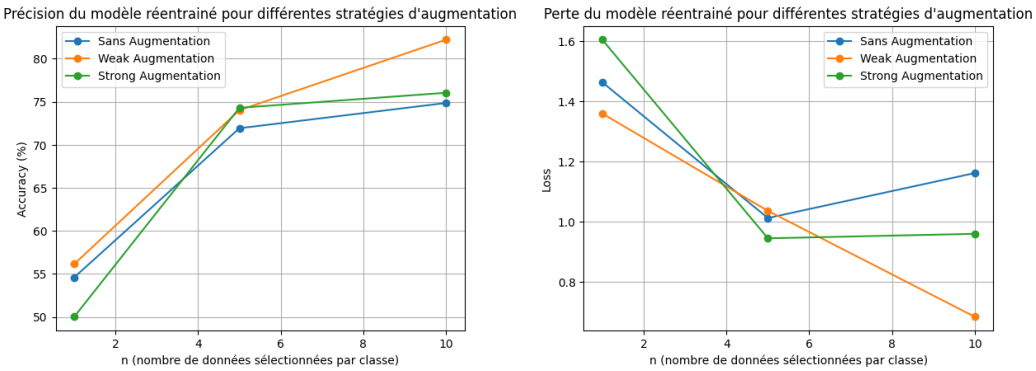

Figure 9: Précision et Perte du modèle réentrainé pour différentes stratégies d'augmentation

6 Résultats finaux

Dans cette partie finale, nous reprenons les entrainements réalisés dans la partie 4 mais appliquant les données labélisés sélectionnées dans la partie 5.

6.1 100 données labellisées

Pour 100 données labellisées, nous avons utilisé les 100 meilleurs labels obtenus par forte augmentation.

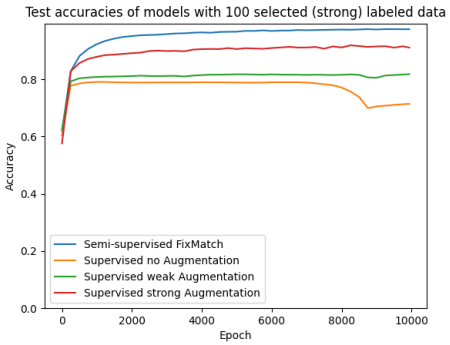

Figure 10: Test accuracy des modèles avec 100 données labellisées choisies

Comme on peut le voir sur la figure 10, les résultats obtenus sur les modèles sont similaires à ceux obtenus lors de la partie 4.1, avec un FixMatch qui atteint un test accuracy de 97,82%, soit une augmentation de seulement 0,33%. Cette faible différence peut soit venir du fait qu'il n'y a aucun gain avec les 100 données sélectionnées, ou soit que les 100 données sélectionnées aléatoirement sont déjà des images apportant des informations intéressantes à l'apprentissage (les données aléatoires par hasard seraient aussi intéressantes que celles sélectionnées).

6.2 50 données labellisées

Nous avons ensuite testé les 50 données sélectionnées par forte augmentation et par faible augmentation.

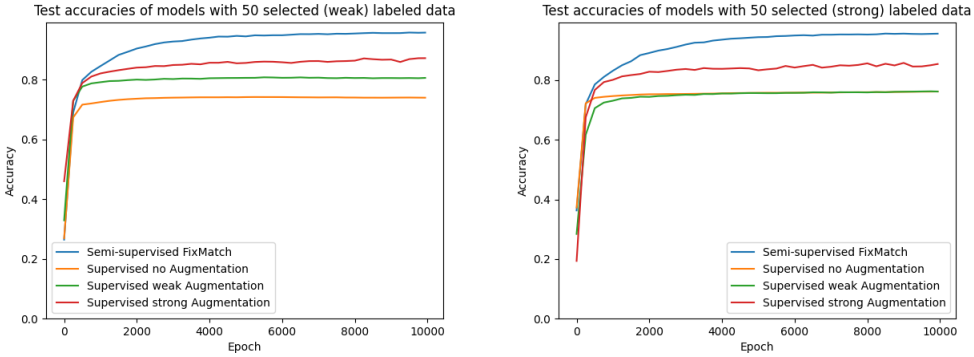

Figure 11: Test accuracy des modèles avec 50 données labellisées par faible et forte augmentation

Dans les deux cas, le FixMatch reste au même niveau, à 96,14% (faible) et 96,15% (forte) d'accuracy sur les tests. Comme pour la partie 6.1, les même suppositions s'appliquent.

6.3 10 données labellisées

Enfin, nous allons observer si la sélection influence dans le cadre de 10 données labellisées.

Avec les 10 données sélectionnées par faible augmentation, on remarque que le modèle FixMatch ne progresse pas, avec 72,93% d'accuracy sur les tests. On peut observer sur la figure 12 que les modèles supervisés profitent eux de cette sélection, avec le modèle entrainté par forte augmentation qui atteint même 69,08% au maximum, ce qui est très proche du FixMatch.

Avec 10 données sélectionnées par forte augmentation, les résulats sont équivalents à ceux avec les données sélectionnées par faible augmentation. Cependant, cette fois-ci, le modèle semi-supervisé atteint jusqu'à 86,19% d'accuracy, ce qui est supérieur aux résultats obtenus lorsque les modèles supervisés sans augmentation réalisée ou faible augmentation appliquée avaient 100 données labélisées.

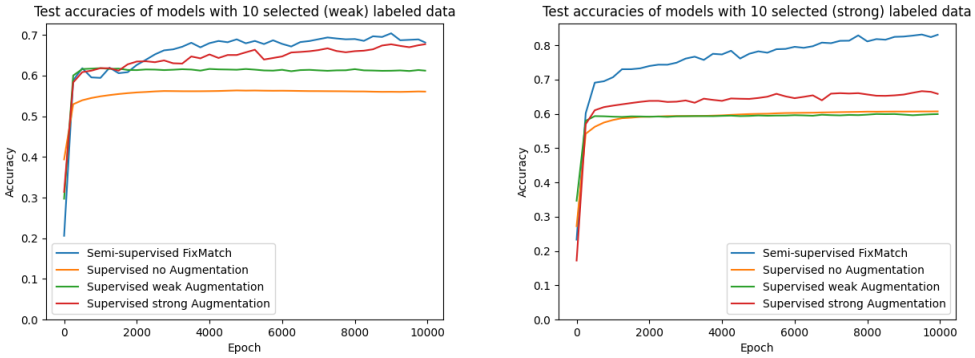

Figure 12: Test accuracy des modèles avec 10 données labellisées par faible et forte augmentation

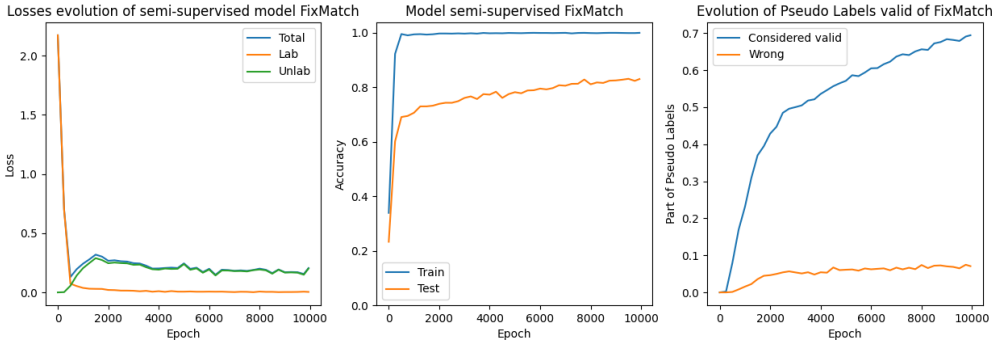

Figure 13: Résultats entrainements avec 10 données labellisées choisies par forte augmentation

Comme on peut le voir sur la figure 13, les losses comme le nombre de pseudo-labels sont similaires aux résultats de la figure 8. La différence est dans le nombre de pseudo-labels faux (mal estimés) qui augmente moins. Cela indique que les 10 données sélectionnées par forte augmentation donnent une base suffisante à FixMatch pour pouvoir estimer les pseudo-labels plus correctement.

7 Conclusion

La semi-supervision réalisée par FixMatch a montré durant ce projet des résultats prometteurs face aux modèles complétement supervisés dans le cas où la quantité d'informations labellisées est faible. On a aussi observé que le choix des données labellisées peut améliorer les résultats comme le montre bien la figure 14 où le choix à 10 données labellisées montre de bons gains.

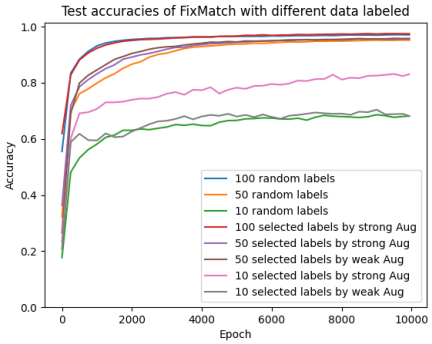

Figure 14: Test accuracy de FixMatch avec différents données labellisées

References

[1] Sohn K, Berthelot D, Li CL, Zhang Z, Carlini N, Cubuk ED, et al.. FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence. arXiv;. Available from: http://arxiv.org/abs/2001.07685.

[2] Carlini N, Erlingsson Papernot N. Distribution Density, Tails, and Outliers in Machine Learning: Metrics and Applications. arXiv;. Available from: http://arxiv.org/abs/1910.13427.