

# SELF-SUPERVISED PRIME-DUAL NETWORKS FOR FEW-SHOT IMAGE CLASSIFICATION

Reproduction par Nawel Astouati, Maya Touzari, Dao Thauvin

## DESCRIPTION DE LA TÂCHE

### Contexte

Le few-shot learning (FSL) est une tâche de machine learning où l'objectif est d'apprendre un modèle à partir d'un dataset contenant peu de données. L'article traite du FSL dans un contexte de classification d'images dans un cadre à  $K$ -way  $N$ -shot, où  $N$  représente le nombre d'images étiquetées de chacune des  $K$  classes. Pendant l'entraînement, le réseau prend en entrée un ensemble de support  $S = \{S_{kn} \mid 1 \leq k \leq K, 1 \leq n \leq N\}$  et un ensemble de queries  $Q = \{Q_{km} \mid 1 \leq k \leq K, 1 \leq m \leq M\}$ , le but est de réussir à prédire les labels du query set à partir du support set. La relation apprise doit ensuite pouvoir être généralisée à des classes non vues pendant l'entraînement.

### Introduction

La solution proposée dans l'article utilise 2 réseaux : un **primal**  $\Phi_{S \rightarrow Q} : S \times Q \times L(S) \rightarrow \hat{L}(Q)$  prédisant les labels  $\hat{L}(Q)$  des queries et un **dual**  $\Gamma_{Q \rightarrow S} : S \times Q \times L(Q) \rightarrow \hat{L}(S)$  prédisant les labels  $\hat{L}(S)$  de l'ensemble support ; l'ensemble query devient l'ensemble dont on connaît les classes, les labels prédits par le primal sont utilisés, et l'ensemble support devient l'ensemble dont on doit prédire les labels. En cascade les deux réseaux ensemble, on a  $\Gamma_{Q \rightarrow S}(\Phi_{S \rightarrow Q}(L(S))) = \Gamma_{Q \rightarrow S}(\hat{L}_\Phi(Q)) = \hat{L}_{\Phi, \Gamma}(S)$ . L'article propose de privilégier les labels prédits  $\hat{L}_{\Phi, \Gamma}(S)$  reconstruisant le mieux les labels des supports.

### Dataset

Dans l'article, le modèle a été testé sur cinq datasets. Nous en avons choisi un parmi les 5 pour entraîner notre reproduction du modèle : Mini-imageNet. Mini-ImageNet a été proposé pour l'évaluation du few-shot learning. Il contient 100 classes sélectionnées aléatoirement du dataset ImageNet. Chaque classe comporte 600 exemples de taille  $84 \times 84$ . L'ensemble d'entraînement contient 64 classes, l'ensemble de validation contient 16 classes et l'ensemble de test comporte 20 classes. Le modèle appris sera aussi testé sur le dataset CUB contenant 200 classes d'oiseaux pour tester le modèle sur du cross-domain FSL.

## DESCRIPTION DE LA PIPELINE

### SELF-SUPERVISED LEARNING

Dans un contexte de FSL, les deux ensembles support  $S$  et query  $Q$  sont symétriques. Plus précisément, si un modèle apprend à classifier les images du query set  $Q$  à partir de l'ensemble support  $S$ , l'ordre peut être inversé en prédisant les classes de l'ensemble support  $S$  à partir de  $Q$  en utilisant la même architecture de réseau. Cette caractéristique permet de concevoir un réseau primal-dual commutatif comme illustré par la figure suivante :

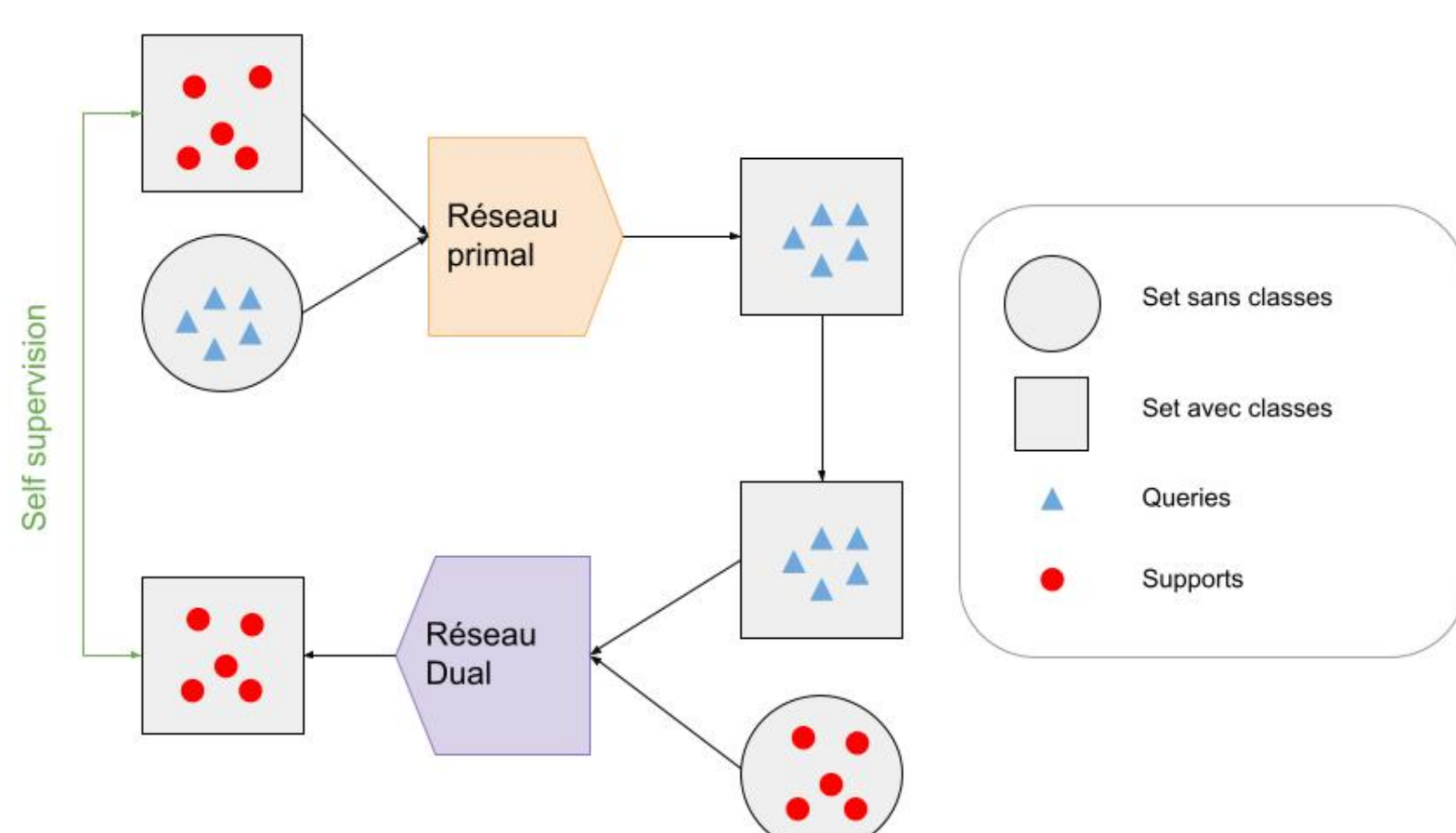


Figure 1: Architecture générale du réseau primal-dual.

Plus précisément, à partir de l'ensemble de support, le réseau primal apprend à prédire les labels de l'ensemble de requêtes. Comme dans les réseaux FSL existants, le coût correspond à :

$$\mathcal{L}_Q^P = \|\hat{L}_\Phi(Q) - L(Q)\|_2.$$

L'ensemble des requêtes et les labels prédits sont donnés comme entrée du réseau dual. Le réseau peut alors prédire les labels de l'ensemble des supports et le coût (self-supervision loss) est donné par l'équation :

$$\mathcal{L}_S^P = \|\hat{L}_{\Phi, \Gamma}(S) - L(S)\|_2.$$

Ces deux coûts sont combinés pour former la fonction de coût du primal:

$$L^P = \|\hat{L}_\Phi(Q) - L(Q)\|_2 + \alpha \cdot \|\hat{L}_{\Phi, \Gamma}(S) - L(S)\|_2 \quad (1)$$

Avec  $\alpha = 0.5$ .

De même, pour la formation du réseau dual, la fonction de coût est donnée par:

$$L^D = \|\hat{L}_\Gamma(S) - L(S)\|_2 + \alpha \cdot \|\hat{L}_{\Phi, \Gamma}(Q) - L(Q)\|_2 \quad (2)$$

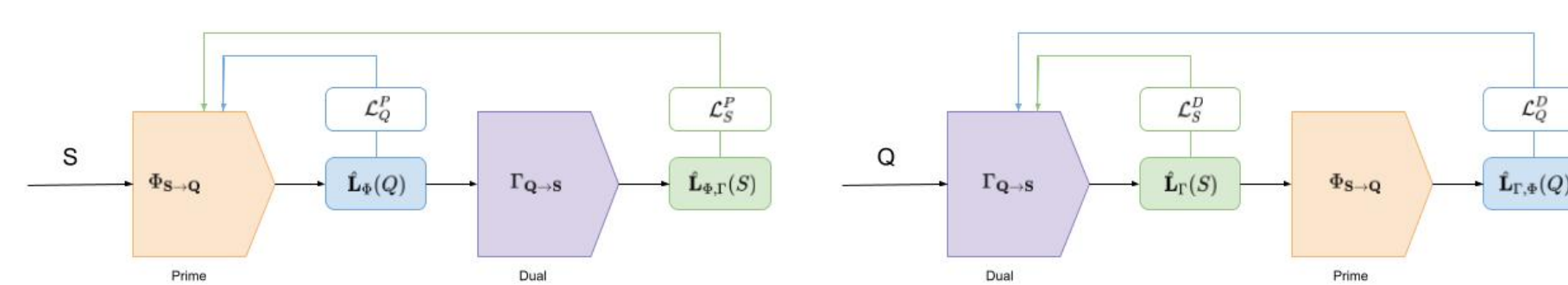


Figure 2: Optim. du primal.

Figure 3: Optim. du dual.

### Graph Neural Network (GNN)

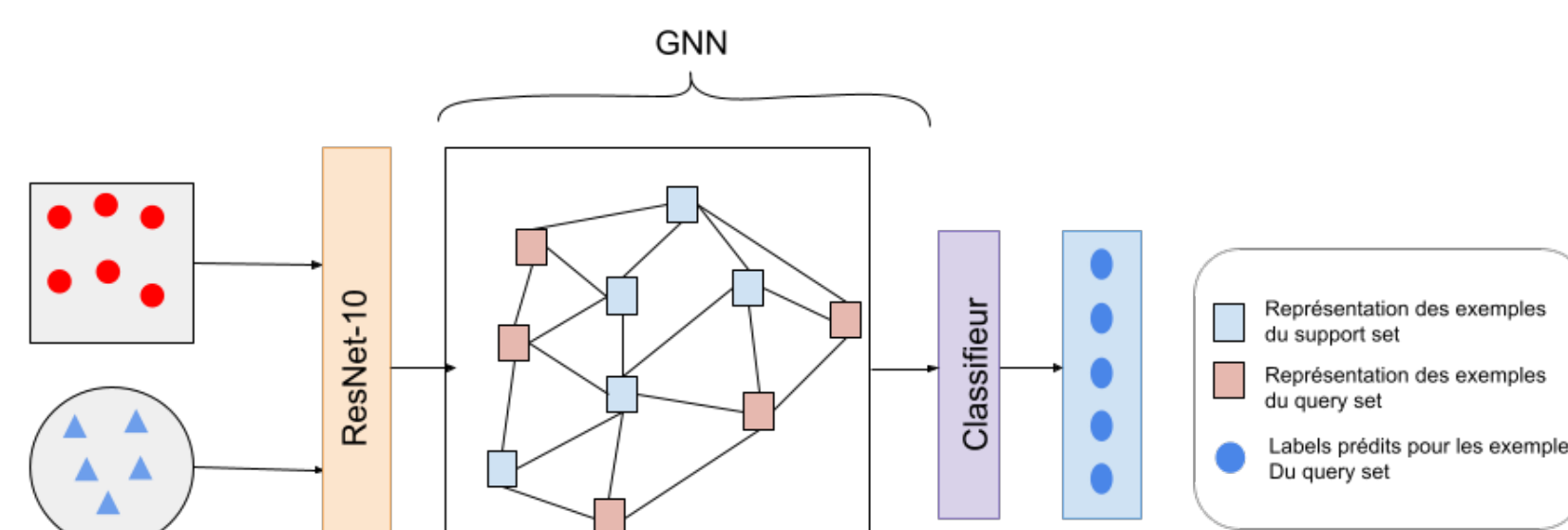


Figure 4: Graph neural network.

- **Noeuds:**  $\{x_j^t \mid 1 \leq j \leq J\}$ ,  $J = K \times (N + 1)$ , Ici, les noeuds à  $t=0$  représentent la représentation initiale venant d'un ResNet des supports et des queries.
- **Arêtes:**  $\psi(x_i^t, x_j^t)$
- **Initialisation de la corrélation pour les supports :**

$$\psi(x_i^t, x_j^t) = \begin{cases} 1 & \text{if } L(x_i^t) = L(x_j^t) \\ 0 & \text{if } L(x_i^t) \neq L(x_j^t) \end{cases}$$

- **Initialisation de la corrélation pour les queries:**

$$\psi(x_i^t, x_j^t) = \text{Random}(0, 1)$$

- **Apprentissage de la corrélation :**

$$\psi(x_i^t, x_j^t) = \mathcal{G}_e[x_i^t, x_j^t],$$

- **Apprentissage des représentations des noeuds:**

$$x_j^{t+1} = \mathcal{G}_o \left[ \sum_{i=1}^J x_i^t \cdot \psi(x_i^t, x_j^t) \right]$$

### SELF-SUPERVISED OPTIMIZATION

L'intérêt de la loss  $L_S^P$  est qu'elle est aussi applicable sans connaître les classes des queries et donc durant l'inférence. Ainsi, le SELF-SUPERVISED OPTIMIZATION ou SSO va affiner les prédictions avec cette loss lors de l'inférence.

Le processus est le suivant:

- Prédire  $\hat{L}(Q)$  avec le primal. Chaque query a une probabilité pour chaque classe.
- Récupérer les  $K$  vecteurs de classes sans incertitude, les plus proches (au sens de la distance euclidienne) du vecteur de classes prédit:  $\hat{L}^i(Q)$   $i \in 1, \dots, K$ .
- Pour chaque vecteur, prédire  $\hat{L}_{\Phi, \Gamma}(S)$  avec le dual.
- Rechercher  $\hat{L}^i(Q)$  avec la loss  $L_S^P$  la plus faible, c'est-à-dire, celui ayant le mieux reconstruit les classes des supports. Ce vecteur correspond alors aux classes prédites.

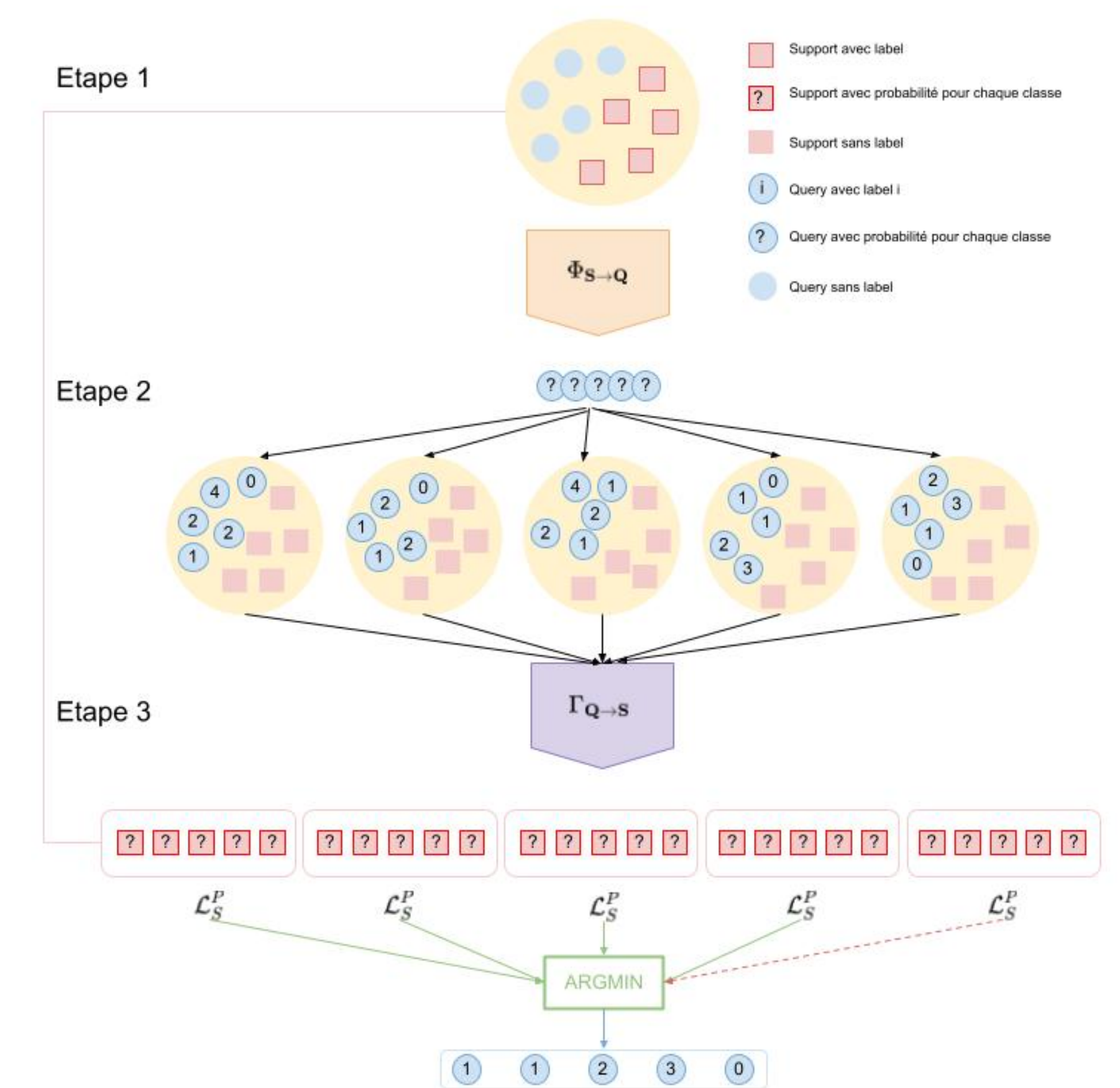


Figure 5: Exemple de SSO en 5-way 1-shot.

### Remarque:

Dans le cas où il y a plus de supports que de queries, on enlève la classe d'une partie seulement des supports.

## RÉSULTATS

### REPRODUCTIVITÉ

- De nombreux paramètres manquent dans l'article : l'implémentation de  $\mathcal{G}_e$  et  $\mathcal{G}_o$ , le nombre d'itération  $t$  avec le GNN, etc.
- Un code est fourni mais, celui-ci est incohérent avec l'article: nombre de données par batch, initialisation des arêtes, utilisation des couches de MAML [2] (ce qui n'est pas indiqué dans l'article), etc.
- Le code contient des erreurs : certaines des possibilités de classes des queries ne sont pas testées avec le SSO, une couche linéaire n'est jamais entraînée.

## CONCLUSION

- L'article est difficilement reproductible.
- A la différence de l'article, on observe une baisse de performance avec SSO.

## RÉFÉRENCES

- [1] Cao, Wenming, et al. Self-Supervised Prime-Dual Networks for Few-Shot Image Classification. septembre 2021. openreview.net, <https://openreview.net/forum?id=SHnXjl3vTJ>.
- [2] Finn, Chelsea, et al. « Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks ». arXiv:1703.03400 [cs], juillet 2017. arXiv.org, <http://arxiv.org/abs/1703.03400>.