

# Rapport Technique

Étude Comparative : Fondements Mathématiques des CNN et  
RNN

**Laboratoire de Deep Learning From Scratch**

26 Janvier 2025

## Résumé

Ce rapport détaille les implémentations "from scratch" des réseaux de neurones convolutionnels (CNN) et récurrents (RNN). Nous explorons la divergence mathématique entre l'opération de convolution spatiale et la récursion temporelle, ainsi que leur impact sur la propagation du gradient.

# 1 Introduction

Le développement de modèles sans framework (TensorFlow/PyTorch) nécessite une compréhension précise de l'algèbre linéaire. Alors que les CNN exploitent l'invariance par translation, les RNN se concentrent sur la persistance de l'information à travers le temps.

## 2 Analyse Mathématique du CNN

### 2.1 L'Opération de Convolution

L'essence d'un CNN réside dans l'application d'un noyau (kernel)  $W$  sur une entrée  $X$ . Pour une image 2D, la valeur à la position  $(i, j)$  de la carte de caractéristiques est donnée par :

$$(X * W)_{i,j} = \sum_m \sum_n X_{i+m, j+n} \cdot W_{m,n} + b \quad (1)$$

Cette opération permet un **partage de paramètres**. Le même filtre est utilisé pour toute l'image, réduisant drastiquement le nombre de poids par rapport à un ANN classique.

### 2.2 Rétropropagation spatiale

Le gradient par rapport aux poids du filtre ( $\frac{\partial L}{\partial W}$ ) est lui-même une convolution entre l'entrée et le gradient de la couche suivante, ce qui assure une mise à jour cohérente des détecteurs de traits (bords, textures).

## 3 Analyse Mathématique du RNN

### 3.1 La Récurrence et l'État Caché

Contrairement aux CNN, les RNN traitent les entrées  $x_t$  séquentiellement en maintenant un état caché  $h_t$ . L'équation de transition est :

$$h_t = \sigma(W_{hh}h_{t-1} + W_{xh}x_t + b_h) \quad (2)$$

Ici, la mémoire est stockée dans  $h_{t-1}$ . La complexité réside dans le fait que  $W_{hh}$  est réutilisé à chaque étape temporelle.

### 3.2 Backpropagation Through Time (BPTT)

La dérivation du gradient dans un RNN est plus complexe à cause de la dépendance temporelle :

$$\frac{\partial L}{\partial W_{hh}} = \sum_{t=1}^T \frac{\partial L_t}{\partial h_t} \left( \prod_{k=1}^t \frac{\partial h_k}{\partial h_{k-1}} \right) \frac{\partial h_t}{\partial W_{hh}} \quad (3)$$

## 4 Comparaison : CNN vs RNN

Caractéristique	CNN	RNN
Dépendance	Spatiale (Voisinage)	Temporelle (Séquence)
Poids	Partagés spatialement	Partagés temporellement
Problématique	Explosion du calcul (Stride)	Disparition du gradient
Flux de données	Forward (Unidirectionnel)	Boucle de rétroaction

TABLE 1 – Synthèse des différences structurelles.

## 5 Conclusion

L'implémentation "from scratch" révèle que le CNN est un extracteur de caractéristiques locales extrêmement efficace, tandis que le RNN agit comme une machine à états dont la stabilité numérique dépend fortement de l'initialisation de  $W_{hh}$  pour éviter l'évanouissement du gradient.