Projet PAH: Déploiement IA HPML	
Thierry Garcia - Jordan Rey-Jouanchicot	Programmation sur GPU
ING3	Année 2024–2025

1 Description et déroulement du projet

L'objectif de ce projet est d'explorer et expérimenter autour du déploiment de modèles d'intelligence artificielle. Dans cet objectif, les attentes sont autour de l'expérimentation de différentes techniques d'optimisation des modèles d'intelligence artificielle, notamment la quantification, le pruning et d'autres stratégies de compression, présentées rapidement en cours. Ainsi que des tests de performances sur différents moteurs de déploiments.

Ce cours étant orienté sur la programmation GPU, il est attendu une analyse de l'usage GPU, des choix effectués vis-à-vis des architectures cibles, ainsi que de différents moteurs de déploiements. Des comparaisons de perfomances sur des GPUs à architectures différentes seront attendues, de même que des comparaisons des méthodes de déploiements, optimisations, etc.

Ce projet vise à offrir une expérience pratique en optimisation et déploiement de modèles d'IA. Il permettra aux étudiants de mieux comprendre les défis liés à l'efficacité computationnelle et de développer des approches adaptées.

1.1 Liste des sujets:

Chaque groupe travaillera sur l'optimisation d'un modèle d'IA:

- **ResNet** - Classification d'images
- **YOLOv8** - Détection d'objets en temps réel
- **BERT** - Traitement automatique du langage (NLP)
- **EfficientNet** - Classification efficace et scalable
- **ViT-B** - Vision par transformeurs
- **LLM** - Tout Large Language Models
- **Any custom pretrained model** - Tout modèle de réseau de neurones

Les difficultés liées au choix du modèle seront prises en compte.

1.2 Livrables attendus

Chaque groupe devra remettre :

- Un **rapport détaillé** expliquant les techniques appliquées, les choix effectués et les impacts observés sur les performances (*trade-off accuracy/throughput*).
- Le **code source** du projet, documenté.

2 Contexte et problématique

Avec la croissance rapide des modèles d'IA, leur utilisation en production pose des défis majeurs :

— **Consommation de ressources** : Les modèles de machine learning et particulièrement avec l'essor des "Large Language Models" nécessitent des infrastructures coûteuses (GPUs, TPUs).

— **Temps d'inférence** : La latence et le throughput peuvent être des problèmes critiques dans les applications en temps réel, par exemple.

— **Déploiement sur des appareils embarqués** : De nombreuses applications nécessitent des modèles plus légers pouvant tourner sur des smartphones, ou des microcontrôleurs.

3 Techniques d'optimisation

3.1 Pruning (Élagage des réseaux de neurones)

Le pruning consiste à supprimer des poids ou des neurones dans le réseau afin de réduire la complexité du modèle. Il existe plusieurs types de pruning :

— **Pruning non-structuré** : Suppression des poids les plus faibles dans les matrices de poids.

— **Pruning structuré** : Suppression de neurones ou de couches entières.

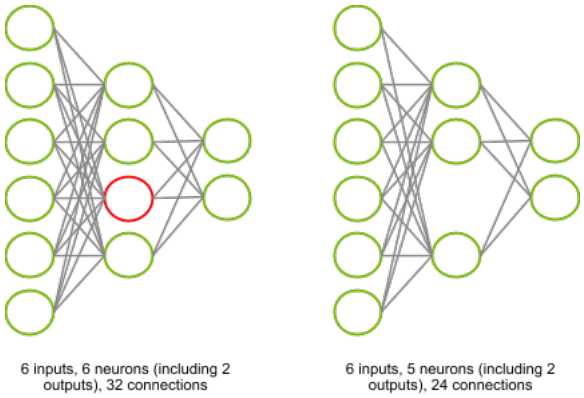

FIGURE 1 – Exemple de pruning dans un réseau de neurones.

3.2 Quantification

La quantification consiste à réduire la précision des poids et des activations pour améliorer l'efficacité computationnelle. Les méthodes principales incluent :

— **Quantification dynamique** : Conversion des poids en valeurs entières lors de l'inférence.

— **Quantification statique** : Conversion des poids et des activations avant l'exécution.

— **Quantification-aware training** : Entraînement du modèle en tenant compte de la quantification.

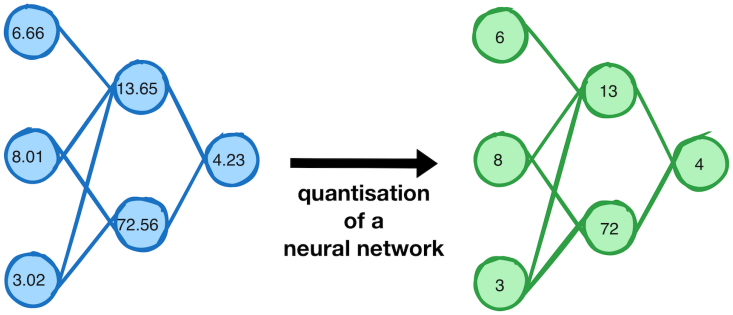

FIGURE 2 – Exemple de modèle non-quantifié, puis après quantification.

4 Méthodes et outils

Les étudiants auront accès à diverses outils et bibliothèques, tels que :

— **PyTorch** : Pruning et quantification avec `torch.quantization`, `torch.prune`.

— **ONNX Runtime** : Exécution optimisée et conversion de modèles.

— **Hugging Face Optimum** : Outils spécifiques aux modèles Transformer.

— **TensorRT** : Accélération sur GPU NVIDIA.

4.1 Proposition de déroulement

— Introduction théorique aux techniques d'optimisation des modèles d'IA. Présentation des sujets et formation des groupes. Implémentation de la version de base du modèle choisi sur GPU et analyse des performances initiales.

— Définition des critères de perfomances à étudier.

— Application de techniques d'optimisations (pruning, quantification, distillation) et analyse de performances.

— Expérimentation autour de divers moteurs de déploiement.

— Finalisation des résultats, rapports complets.

5 Critères d'évaluation

L'évaluation prendra en compte plusieurs critères afin de mesurer la pertinence des optimisations et la qualité de l'analyse. Les principaux seront:

— **Exploration** : Exploration de différentes optimisations et méthode de déploiements.

— **Application optimisations** : Amélioration des performances (vitesse, taille, efficacité énergétique), et aussi adapté à l'architecture cible.

— **Rigueur de l'analyse** : Analyse complète de l'impact sur les performances, choix des métriques.

— **Clarté du rapport** : Explications, justification des choix, méthodologie, résultats et discussion.

En complément, des points intermédiaires seront réalisés sur les trois jours du projet. Ces sessions permettront d'évaluer la progression réelle des groupes, leur compréhension des techniques mises en œuvre et leur capacité à surmonter les obstacles rencontrés. L'implication des étudiants, la pertinence de leurs choix techniques et leurs démarches seront prises en compte dans la notation. L'objectif est d'assurer que chaque groupe mène une véritable expérimentation et ne se limite pas à générer des résultats sans maîtrise du sujet. Ces observations viendront compléter l'évaluation finale.

Bon projet à tous!