

# Optimization des Hyperparamètres appliquée au Fine Tuning de LLM

Basé sur l'article soumis pour OLA2025 : *Bayesian and Partition-Based Optimization for Hyperparameter Optimization of LLM Fine-Tuning*

Nathan Davouse



# Large Language Models

## Point clés

- ▶ Etat de l'art pour le traitement de langage naturel.
- ▶ Réseaux de Neurones avec une architecture basée sur le transformer<sup>1</sup> (annexe 3)
- ▶ Taille : entre 1 et 405 Milliards de neurones

## Fine Tuning

Aspect	Pre-entrainement	Fine Tuning
Objectif	Apprentissage général	Adaptation à un domaine
Données	Larges et diverses	Restreintes et Spécifiques
Ressources	Centaines de GPU	au moins 1 GPU
Durée	Semaine/Mois	Heures/Jours

**Table:** Comparaison entre le Pre-entrainement et le Fine Tuning de LLM

### Parameter-Efficient Fine-Tuning (PEFT)

- Ensemble de méthodes pour réduire le nombre de paramètres à entraîner
- Utilisation de la méthode LoRA (annexe 5)
- Amène des nouveaux hyperparamètres

# Optimisation des Hyperparamètres (OHP)

## Hyperparamètres

Paramètres qui ne sont pas entraînés par le modèle  
(learning rate, dropout ...)

## Objectifs

- Meilleure performance qu'en manuel
- Retirer le besoin d'expertise

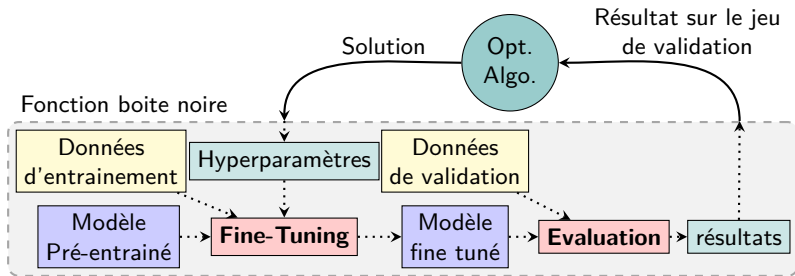


Figure: Fonctionnement général de l'optimisation des hyperparamètres



# Sommaire

1. Introduction

2. Design et Implémentation

3. Résultats et Analyses

4. Conclusion





## Stratégie de Recherche : Optimisation Bayésienne par Process Gaussien

## Principe

Utiliser un substitut moins cher à optimiser pour explorer l'espace de recherche

## Algorithme

- Echantillonnage et evaluation (LHS)

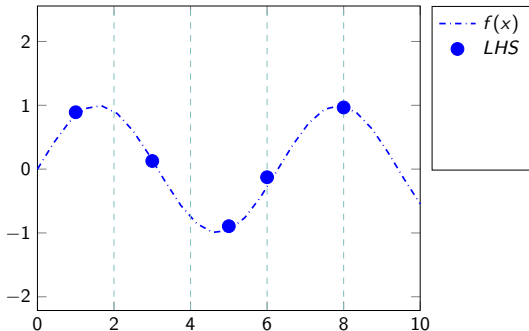


Figure: Exemple d'un surrogate sur une fonction en 1D











## Stratégie de Recherche : Bayesian Multi-Scale Optimistic Optimization (BaMSOO)

Décomposition suivant SOO, mais utilisant des Process Gaussien pour éviter les évaluations non prometteuses.

### Evaluation par BaMSOO : $g(x)$

- ▶ Si  $UCB(x)^2 > f^+$  :
  - $g(x) = f(x)$  évaluation réel
- ▶ Sinon :
  - $g(x) = LCB(x)^3$  remplace  $f(x)$  par  $LCB(x)$

<sup>2</sup>Upper Confidence Bound : Borne Sup. de l'intervalle de confiance

<sup>3</sup>Lower Confidence Bound : Borne Inf. de l'intervalle de confiance

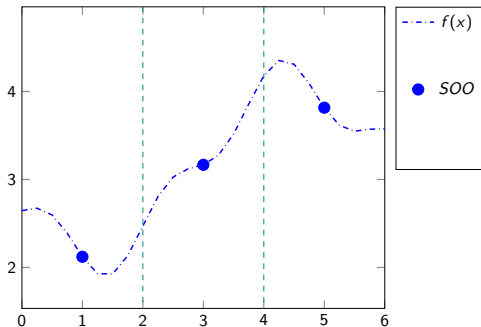


Figure: Illustration de l'Algorithme BaMSOO

## Stratégie de Recherche : Bayesian Multi-Scale Optimistic Optimization (BaMSOO)

Décomposition suivant SOO, mais utilisant des Process Gaussien pour éviter les évaluations non prometteuses.

### Evaluation par BaMSOO : $g(x)$

- ▶ Si  $UCB(x)^2 > f^+$  :
  - $g(x) = f(x)$  évaluation réel
- ▶ Sinon :
  - $g(x) = LCB(x)^3$  remplace  $f(x)$  par  $LCB(x)$

<sup>2</sup>Upper Confidence Bound : Borne Sup. de l'intervalle de confiance

<sup>3</sup>Lower Confidence Bound : Borne Inf. de l'intervalle de confiance

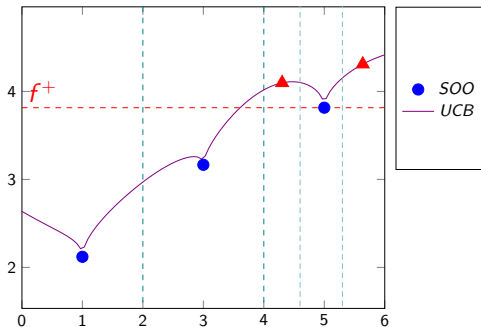


Figure: Illustration de l'Algorithme BaMSOO



## Stratégie de Recherche : Bayesian Multi-Scale Optimistic Optimization (BaMSOO)

Décomposition suivant SOO, mais utilisant des Process Gaussien pour éviter les évaluations non prometteuses.

### Evaluation par BaMSOO : $g(x)$

- ▶ Si  $UCB(x)^2 > f^+$  :
  - $g(x) = f(x)$  évaluation réel
- ▶ Sinon :
  - $g(x) = LCB(x)^3$  remplace  $f(x)$  par  $LCB(x)$

<sup>2</sup>Upper Confidence Bound : Borne Sup. de l'intervalle de confiance

<sup>3</sup>Lower Confidence Bound : Borne Inf. de l'intervalle de confiance

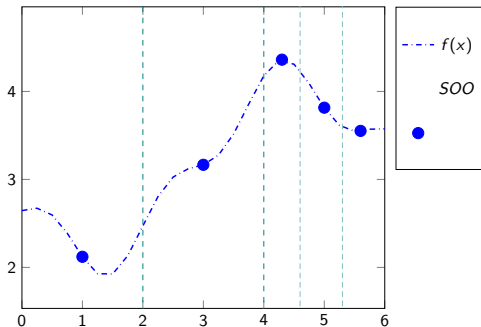


Figure: Illustration de l'Algorithme BaMSOO

## Stratégie de Recherche : Bayesian Multi-Scale Optimistic Optimization (BaMSOO)

Décomposition suivant SOO, mais utilisant des Process Gaussien pour éviter les évaluations non prometteuses.

### Evaluation par BaMSOO : $g(x)$

- Si  $UCB(x)^2 > f^+$  :
  - $g(x) = f(x)$  évaluation réel
- Sinon :
  - $g(x) = LCB(x)^3$  remplace  $f(x)$  par  $LCB(x)$

<sup>2</sup>Upper Confidence Bound : Borne Sup. de l'intervalle de confiance

<sup>3</sup>Lower Confidence Bound : Borne Inf. de l'intervalle de confiance

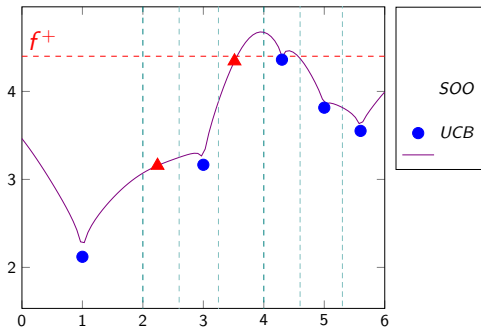


Figure: Illustration de l'Algorithme BaMSOO

## Stratégie de Recherche : Bayesian Multi-Scale Optimistic Optimization (BaMSOO)

Décomposition suivant SOO, mais utilisant des Process Gaussien pour éviter les évaluations non prometteuses.

### Evaluation par BaMSOO : $g(x)$

- ▶ Si  $UCB(x)^2 > f^+$  :
  - $g(x) = f(x)$  évaluation réel
- ▶ Sinon :
  - $g(x) = LCB(x)^3$  remplace  $f(x)$  par  $LCB(x)$

<sup>2</sup>Upper Confidence Bound : Borne Sup. de l'intervalle de confiance

<sup>3</sup>Lower Confidence Bound : Borne Inf. de l'intervalle de confiance

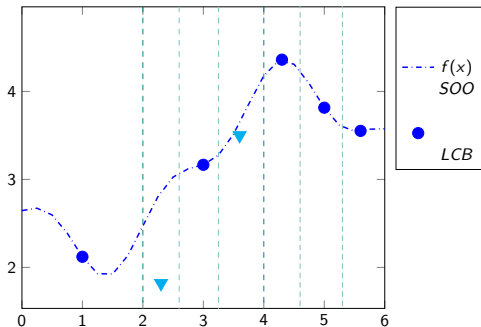


Figure: Illustration de l'Algorithme BaMSOO

## Implémentation

- Programmation Orienté Object en Python
- Travail de documentation : *readme*, indication de type...
- Objectif : permettre le réusage
- Utilisable en ligne de commande pour Grid5000
- Intégralement open-source<sup>4</sup>

<sup>4</sup>[https://github.com/Kiwy3/BO\\_PBO\\_HPO\\_LLM](https://github.com/Kiwy3/BO_PBO_HPO_LLM)

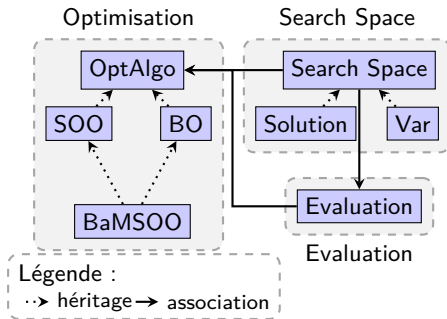


Figure: Diagramme de l'implémentation

# Sommaire

1. Introduction

2. Design et Implémentation

3. Résultats et Analyses

4. Conclusion

## Echantillonnage par Latin Hypercube Sampling (LHS)

Objectif : Explorer l'espace et proposer une borne inférieure

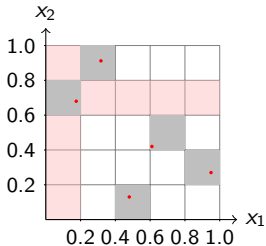


Figure: Illustration du Latin Hypercube Sampling avec  $g = 5$

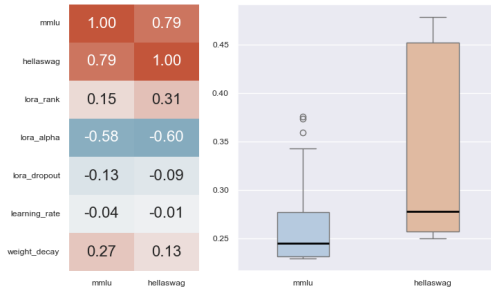


Figure: Résumé des résultats par échantillonnage<sup>5</sup>

<sup>5</sup>A droite : corrélation entre les variables et les métriques / A gauche : distribution des métriques







## Résultats des 3 algorithmes : BO, SOO et BaMSOO

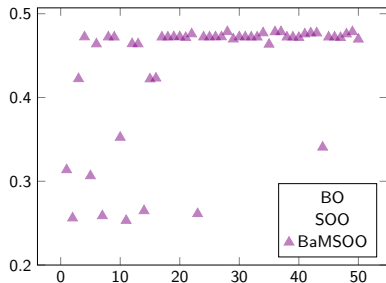


Figure: Résultats sur Hellaswag (Validation)

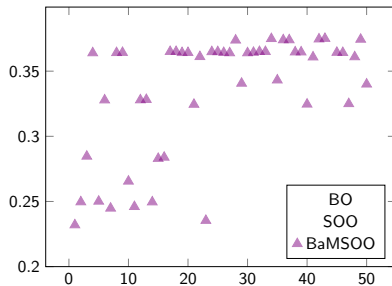


Figure: Résultats sur MMLU (Test)

## Résultats des 3 algorithmes : BO, SOO et BaMSOO

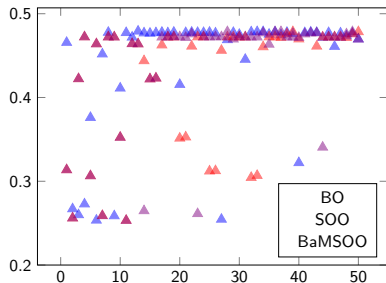


Figure: Résultats sur Hellaswag (Validation)

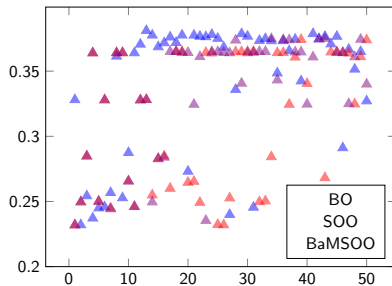


Figure: Résultats sur MMLU (Test)



## Perspectives

### Poursuite du travail

- ▶ Retour sur l'article et présentation en conférence (si validation)
- ▶ Elargissement de l'espace de recherche
- ▶ Diversification sur les modèles/données

### Optimisation fractale parallèle enrichie par approche bayésienne<sup>6</sup>

- ▶ Généralisation de l'hybridation décomposition/bayésien dans un cadre parallèle
  - 5 moyens d'exploiter le *surrogate* pour améliorer l'optimisation fractale

---

<sup>6</sup>Parallel Bayesian-enhanced Fractals Optimization

# Sommaire

## 4. Conclusion

## Résultats du stage

### Implémentation d'OHP pour du Fine Tuning de LLM

Preuve du concept d'utilisation des algorithmes utilisés, ainsi qu'une base pour de futur travaux de l'équipe

### Comparaison entre approche bayésien/décomposition pour fonction couteuse

Dont l'exploration d'une première partie de l'hybridation des deux

Tend vers une généralisation d'une approche par décomposition améliorée par l'utilisation d'un *surrogate*

## Apprentissage

### 1ere expérience de recherche

- ▶ Apprentissage de la rigueur
- ▶ Gestion de la littérature
- ▶ Première écriture d'article

### Programmation pour une démarche de recherche

- ▶ Prototypage et versionnage
- ▶ Transmission pour l'équipe
- ▶ 1ere approche du parallélisme

## Poursuite du projet professionnel

### Poursuite en recherche

Confirmation de l'attrait pour le domaine

### Début d'une thèse en mars

Sujet : Ecological and economic logistics service network design : Models and Decision Support Algorithms

Equipe INOCS, au sein de l'INRIA Lille, dirigé par Frederic Semet



*Merci.*

## Annexe 1 : Travaux connexes

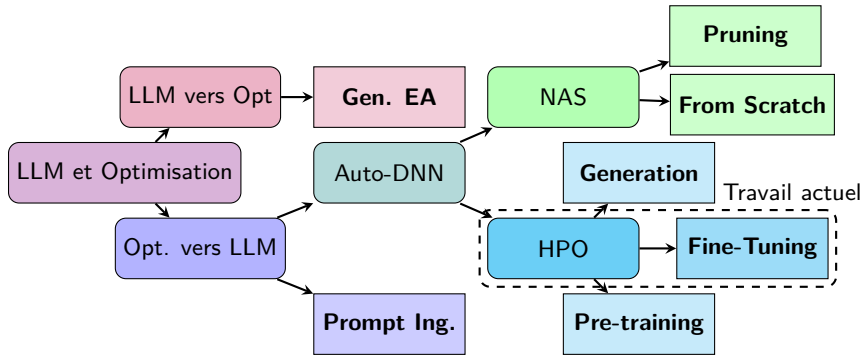


Figure: Classification des travaux similaires

## Annexe 2 : Stratégie d'Evaluation de Solutions

### Implémentation

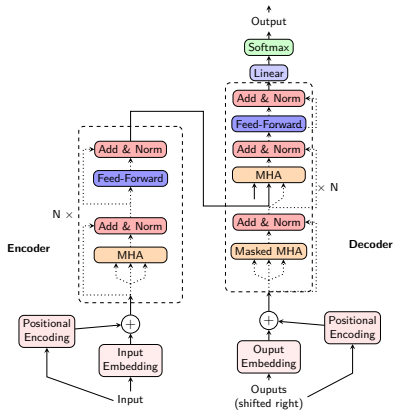
#### ► Fine Tuning

- Modèle : LLaMa-3.2-1B
- Jeu de données d'entraînement : Alpaca
- LitGPT framework : basé sur Pytorch, facilite le Fine Tuning de LLM

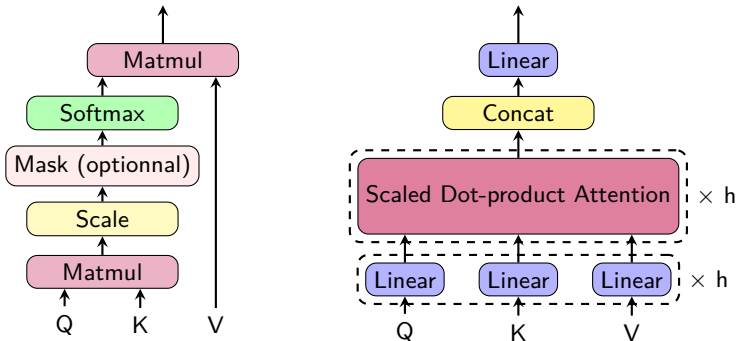
#### ► Evaluation

- librairie lm\_eval : standard pour l'évaluation de LLM
- Evaluation par la précision sur des jeu de données Benchmark : Hellaswag et MMLU

## Annexe 3 : Transformer architecture

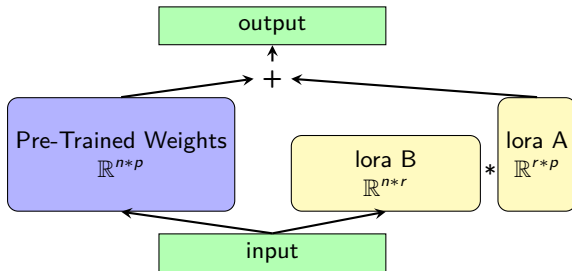


## Annexe 4 : Multi-Head Attention



**Figure:** Illustration du mécanisme d'auto-attention : A droite le mécanisme complet, a gauche le *Scaled Dot-product Attention*

## Annexe 5 : Low Rank Adaptation (LoRA)



**Figure:** Illustration de l'application du Low Rank Adaptation (LoRA)



## Annexe 7 : Résultats pour S00

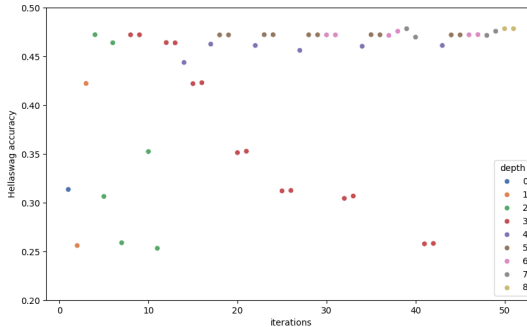


Figure: Evolution des score lors de l'expérience SOO



## Annexe 8 : Résultats pour BaMSOO

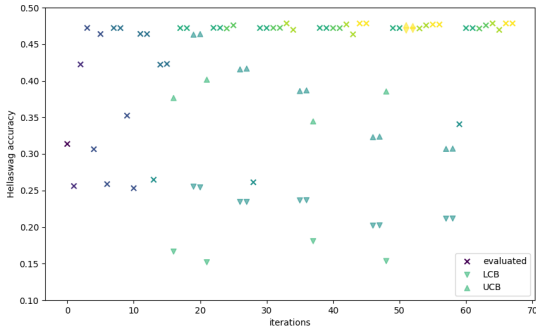


Figure: Evolution des score lors de l'expérience BaMSOO

## Annexe 9 : Aperçu du Bayésien Fractal

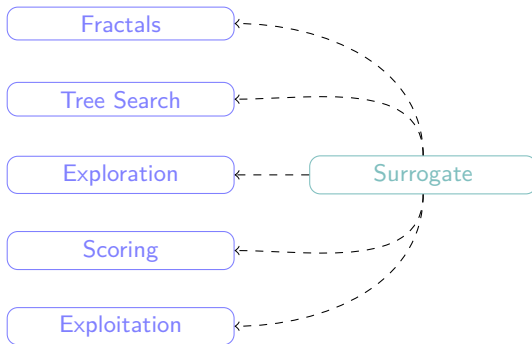


Figure: Interaction du composant Surrogate avec les éléments de base de la décomposition fractale

## Annexe 10 : Modèle Beamer UTT

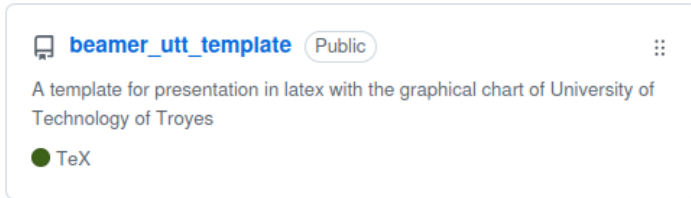


Figure: Presentation du template beamer sur github : lien