**Cahier des charges Mercari Challenge**

1. Présentation et contexte du projet:

Il peut être difficile de connaître la valeur réelle d'un produit. De petits détails peuvent entraîner de grandes différences de prix.

La fixation des prix des produits devient encore plus difficile à grande échelle, compte tenu du nombre de produits vendus en ligne. Les vêtements ont de fortes tendances saisonnières en matière de prix et sont fortement influencés par les marques, tandis que les produits électroniques ont des prix fluctuants en fonction des spécifications du produit.

Mercari, la plus grande application d'achat communautaire du Japon, connaît bien ce problème.

Dans le cadre de notre projet, nous allons créer une api dédiée à tous les vendeurs de Mercari.

L’objectif de notre projet est de pouvoir proposer des suggestions de prix aux vendeurs. Le prix proposé pour chaque produit devrait être le bon prix pour guider les vendeurs dans le choix de prix.

2. Choix du modèle:

Nous avons testé les cinq modèles ci-dessous :

* Machine Learning
* Ridge
* SVR
* Random Forest
* LightGBM
* Deeplearning : MLP ( multilayer perceptron)

Définition des métriques et exigences de performances:

Nous avons obtenu la plus faible RMSE avec le modèle de deep learning multilayer perceptron : 0,38. Cependant, nous avons décidé de garder les deux modèles SVR et Lightgbm pour la partie fastapi qui ont un RMSE également faible i.e. 0,45 et qui ont l’avantage d’avoir un temps d’exécution plus rapide que le modèle multilayer perceptron.

3. Api:

Nous allons créer une api avec Fastapi.

Cette api sera uniquement utilisée par les vendeurs sur le site mercari.

L’api contiendra 2 sections:

* Vendors:
* Chaque vendeur devra d’abord s’authentifier en rentrant son username et son mot de passe. Le vendeur devra alors être dans la base utilisateurs sinon il ne pourra pas utiliser l’api.
* Nous pourrons également récupérer la liste des vendeurs
* Une fonction permettra de renvoyer 1 si l’api fonctionne
* Sales:
* Afin de suggérer les bons prix, chaque vendeur devra rentrer les informations suivantes sur son produit:
* name
* Item condition id : 1 à 5
* Item description
* Category name
* Brand name
* Shipping : 1 si shipping fee est payé par le vendeur , 0 si c’est payé par l’acheteur
* Une fois que toutes les informations seront renseignées , il faudra exécuter la requête pour avoir une suggestion de prix.
* Cette requête sera alors enregistrée dans une base de données historique.
* Le vendeur pourra également voir l’historique de ses requêtes.