DEPLOYEZ UN MODELE DANS LE CLOUD

PROJET N°8
PARCOURS « DATA SCIENTIST »

ETUDIANTE : CATHERINE BRICE

SOUTENANCE DE PROJET

18 MARS 2023

Plan de la présentation

I. Présentation de la problématique

II. Les composants de l'architecture Cloud adoptée et leurs rôles

III. Démarche de mise en œuvre de l'environnement Big Data (EMR)

IV. Les étapes de la chaîne de traitement PySpark

V. Conclusion

I- Rappel de la problématique

SOCIETE "FRUITS"

- Solutions innovantes pour la récolte des fruits en développant des robots cueilleurs intelligents

BESOIN

- Développer une application mobile pour construire une première version de l'architecture Big Data nécessaire

OBJECTIFS

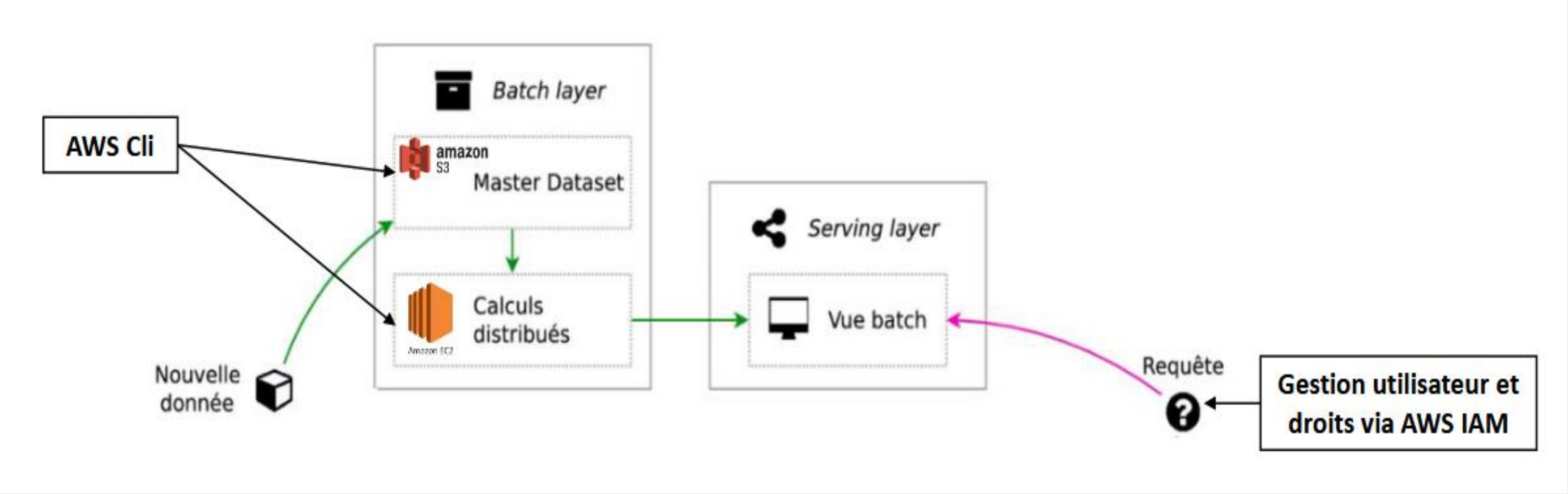
- Sensibiliser le grand public à la biodiversité des fruits et mettre en place une première version du moteur de classification des images de fruits

MISSION

- S’approprier les travaux réalisés par un alternant et compléter la chaîne de traitement
  Mettre en œuvre les premières briques de traitement qui serviront lorsqu’il faudra passer à l’échelle en termes de volume de données

II - Les composants de l'architecture Cloud adoptée et leurs rôles

- Solution PaaS → Amazon EMR
- Choix de région identique pour AWS S3 et AWS EC2 : Francfort (eu-central-1c) → RGPD



II - Les composants de l'architecture Cloud adoptée et leurs rôles

Briques		Rôle(s)
AWS Cli		• Interface de lignes de commande pour interagir avec les différents services d'AWS
AWS Service EMR (PaaS)	Amazon S3	• Gestion et stockage de données
	Instances EC2	• Réalise les calculs issus des données massives stockées sur S3
Interface AWS		• Permet à l'utilisateur de naviguer entre les différents services
AWS IAM		• Gestion utilisateurs et de leurs droits

III - Démarche de mise en œuvre de l'environnement Big Data

① Configuration de l'environnement de travail :

- AWS Cli
- Gestion utilisateur et droits (AWS IAM)
- Création d'une paire de clés EC2

② Upload des datas sur S3 :

- Création d'un bucket sur s3 → s3://p8-data-fruit-cbb
- Copie des dossiers/fichiers nécessaires : images/fichier bootstrap/clefs EC2

③ Configuration du serveur EMR :

- Configuration des logiciels → Version EMR 6.6.0 packages (Spark, Tensorflow et JupyterHub)
- Modification des paramètres du logiciel → Fichier JSON
- Matériel → Instances de type M5xlarge avec 1 instance maître et 2 unités principales
- Modification des paramètres généraux → Bootstrapping
- Sécurité → Paire de clés EC2

III - Démarche de mise en œuvre de l'environnement Big Data

④ Instanciation du serveur

- Création du tunnel SSH à l'instance EC2 Maître

⑧ Résiliation de l'instance EMR / Clonage du cluster EMR

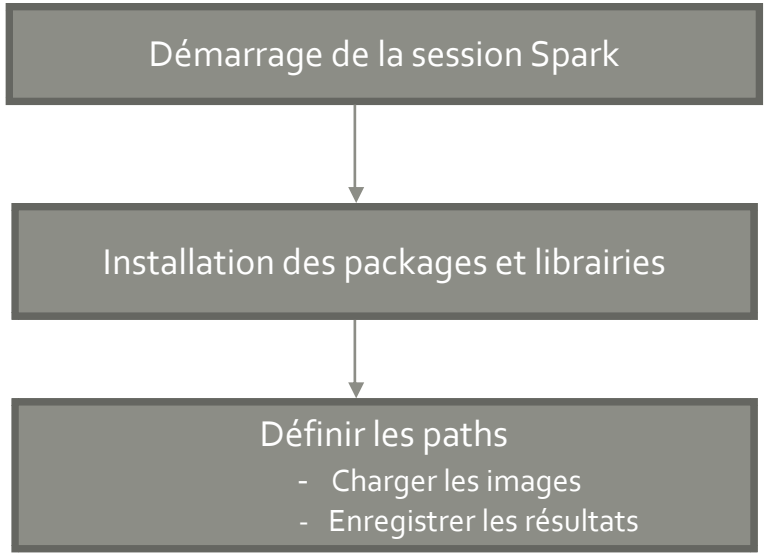
⑤ Connexion au notebook JupyterHub

- Identification et upload du notebook

⑦ Suivi de l'avancement des tâches avec le serveur d'historique Spark

⑥ Exécution du code du notebook

IV - Les étapes de la chaîne de traitement Pyspark




```
+--------------------------------------------------------+--------------+
|path                                                    |label         |
+--------------------------------------------------------+--------------+
|s3://p8-data-fruit-cbb/Test/Pineapple/99_100.jpg        |Pineapple     |
|s3://p8-data-fruit-cbb/Test/Pineapple Mini/99_100.jpg   |Pineapple Mini|
|s3://p8-data-fruit-cbb/Test/Pineapple/143_100.jpg       |Pineapple     |
|s3://p8-data-fruit-cbb/Test/Pineapple Mini/143_100.jpg  |Pineapple Mini|
|s3://p8-data-fruit-cbb/Test/Pineapple/144_100.jpg       |Pineapple     |
+--------------------------------------------------------+--------------+
only showing top 5 rows
```

IV - Les étapes de la chaîne de traitement Pyspark

Traitement des données
Chargement des données images → Path + Label

```
+----------------------------------------------------+--------------+
|path                                                |label         |
+----------------------------------------------------+--------------+
|s3://p8-data-fruit-cbb/Test/Pineapple/99_100.jpg    |Pineapple     |
|s3://p8-data-fruit-cbb/Test/Pineapple Mini/99_100.jpg |Pineapple Mini|
|s3://p8-data-fruit-cbb/Test/Pineapple/143_100.jpg   |Pineapple     |
|s3://p8-data-fruit-cbb/Test/Pineapple Mini/143_100.jpg|Pineapple Mini|
|s3://p8-data-fruit-cbb/Test/Pineapple/144_100.jpg   |Pineapple     |
+----------------------------------------------------+--------------+
only showing top 5 rows
```

Préparation du modèle MobileNetV2 et traitement de diffusion des poids du modèle

• Avant-dernière couche = Vecteur réduit de dimension (1,1,1280)

➢ Première version du moteur pour la classification des images en fruits

```
new_model.summary()
```

block_16_depthwise_relu (ReLU)	(None, 7, 7, 960)	0	block_16_depthwise_BN[0][0]
block_16_project (Conv2D)	(None, 7, 7, 320)	307200	block_16_depthwise_relu[0][0]
block_16_project_BN (BatchNorma	(None, 7, 7, 320)	1280	block_16_project[0][0]
Conv_1 (Conv2D)	(None, 7, 7, 1280)	409600	block_16_project_BN[0][0]
Conv_1_bn (BatchNormalization)	(None, 7, 7, 1280)	5120	Conv_1[0][0]
out_relu (ReLU)	(None, 7, 7, 1280)	0	Conv_1_bn[0][0]
global_average_pooling2d (Globa	(None, 1280)	0	out_relu[0][0]

```
Total params: 2,257,984
Trainable params: 2,223,872
Non-trainable params: 34,112
```

IV - Les étapes de la chaîne de traitement Pyspark

Traitement des données
Chargement des données images → Path + Label

■ Définition du processus de chargement des images et application de leur featurisation avec l'utilisation de pandas UDF
- Prétraitement des octets de l'image brute pour la prédiction
- Featurisation d'une série d'images brutes avec le modèle d'entrée

→ Aplatit les tenseurs de caractéristiques en vecteurs pour faciliter le stockage en dataframe Spark
- Itérateur scalaire qui prend en compte la fonction de featurisation

→ Charge le modèle une fois et permet sa réutilisation pour les lots de données

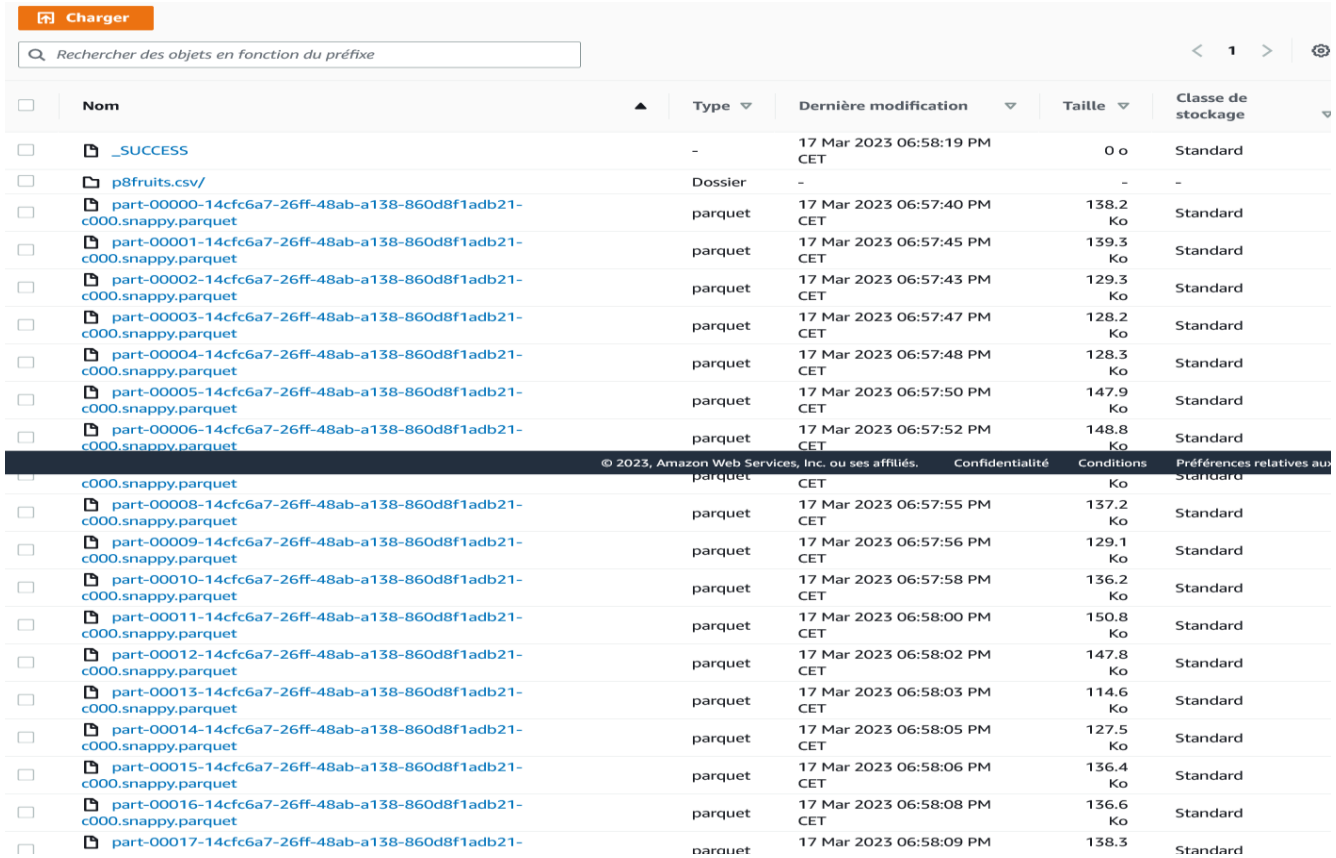
■ Exécution des actions d'extractions de features

```
                                                path  ...                                           features
s3://p8-data-fruit-cbb/Test/Pineapple Mini/25_...  ...  [0.0, 3.8690548, 0.0, 0.0, 0.0, 0.0, 0.0072968...
s3://p8-data-fruit-cbb/Test/Pineapple Mini/17_...  ...  [0.0, 4.6625853, 0.15033168, 0.0, 0.0002589650...
    s3://p8-data-fruit-cbb/Test/Walnut/18_100.jpg  ...  [0.04131617, 0.039828468, 0.0, 0.0, 0.64292824...
    s3://p8-data-fruit-cbb/Test/Walnut/28_100.jpg  ...  [0.15421982, 0.0, 0.0, 0.0, 0.13024212, 0.0, 1...
     s3://p8-data-fruit-cbb/Test/Peach/36_100.jpg  ...  [0.68647873, 0.25965068, 0.0, 0.0, 0.26959205,...

rows x 3 columns]
```

IV - Les étapes de la chaîne de traitement Pyspark

Chargement des données enregistrées et validation du résultat



Charger

Rechercher des objets en fonction du préfixe

1

Nom	Type	Dernière modification	Taille	Classe de stockage
_SUCCESS	-	17 Mar 2023 06:58:19 PM CET	0 o	Standard
p8fruits.csv/	Dossier	-	-	-
part-00000-14cfc6a7-26ff-48ab-a138-860d8f1adb21-c000.snappy.parquet	parquet	17 Mar 2023 06:57:40 PM CET	138.2 Ko	Standard
part-00001-14cfc6a7-26ff-48ab-a138-860d8f1adb21-c000.snappy.parquet	parquet	17 Mar 2023 06:57:45 PM CET	139.3 Ko	Standard
part-00002-14cfc6a7-26ff-48ab-a138-860d8f1adb21-c000.snappy.parquet	parquet	17 Mar 2023 06:57:43 PM CET	129.3 Ko	Standard
part-00003-14cfc6a7-26ff-48ab-a138-860d8f1adb21-c000.snappy.parquet	parquet	17 Mar 2023 06:57:47 PM CET	128.2 Ko	Standard
part-00004-14cfc6a7-26ff-48ab-a138-860d8f1adb21-c000.snappy.parquet	parquet	17 Mar 2023 06:57:48 PM CET	128.3 Ko	Standard
part-00005-14cfc6a7-26ff-48ab-a138-860d8f1adb21-c000.snappy.parquet	parquet	17 Mar 2023 06:57:50 PM CET	147.9 Ko	Standard
part-00006-14cfc6a7-26ff-48ab-a138-860d8f1adb21-c000.snappy.parquet	parquet	17 Mar 2023 06:57:52 PM CET	148.8 Ko	Standard
part-00007-14cfc6a7-26ff-48ab-a138-860d8f1adb21-c000.snappy.parquet	parquet	17 Mar 2023 06:57:53 PM CET	137.2 Ko	Standard
part-00008-14cfc6a7-26ff-48ab-a138-860d8f1adb21-c000.snappy.parquet	parquet	17 Mar 2023 06:57:55 PM CET	137.2 Ko	Standard
part-00009-14cfc6a7-26ff-48ab-a138-860d8f1adb21-c000.snappy.parquet	parquet	17 Mar 2023 06:57:56 PM CET	129.1 Ko	Standard
part-00010-14cfc6a7-26ff-48ab-a138-860d8f1adb21-c000.snappy.parquet	parquet	17 Mar 2023 06:57:58 PM CET	136.2 Ko	Standard
part-00011-14cfc6a7-26ff-48ab-a138-860d8f1adb21-c000.snappy.parquet	parquet	17 Mar 2023 06:58:00 PM CET	150.8 Ko	Standard
part-00012-14cfc6a7-26ff-48ab-a138-860d8f1adb21-c000.snappy.parquet	parquet	17 Mar 2023 06:58:02 PM CET	147.8 Ko	Standard
part-00013-14cfc6a7-26ff-48ab-a138-860d8f1adb21-c000.snappy.parquet	parquet	17 Mar 2023 06:58:03 PM CET	114.6 Ko	Standard
part-00014-14cfc6a7-26ff-48ab-a138-860d8f1adb21-c000.snappy.parquet	parquet	17 Mar 2023 06:58:05 PM CET	127.5 Ko	Standard
part-00015-14cfc6a7-26ff-48ab-a138-860d8f1adb21-c000.snappy.parquet	parquet	17 Mar 2023 06:58:06 PM CET	136.4 Ko	Standard
part-00016-14cfc6a7-26ff-48ab-a138-860d8f1adb21-c000.snappy.parquet	parquet	17 Mar 2023 06:58:08 PM CET	136.6 Ko	Standard
part-00017-14cfc6a7-26ff-48ab-a138-860d8f1adb21-	parquet	17 Mar 2023 06:58:09 PM	138.3	Standard

IV - Les étapes de la chaîne de traitement Pyspark

ACP de réduction de dimensions

- Transformation de la matrice en vecteurs
- Réalisation de l'ACP
- Affichage des résultats
- Transformation inverse de vecteurs à array

```
+--------------------+-------------+--------------------+--------------------+--------------------+
|                path|        label|            features|       netV2_vectors|         pca_vectors|
+--------------------+-------------+--------------------+--------------------+--------------------+
|s3://p8-data-frui...|Pineapple Mini|[0.0, 3.8690548, ...|[0.0,3.8690547943...|[15.2829704533087...|
|s3://p8-data-frui...|Pineapple Mini|[0.0, 4.6625853, ...|[0.0,4.6625852584...|[14.9522592809902...|
|s3://p8-data-frui...|       Walnut|[0.04131617, 0.03...|[0.04131617024540...|[1.55715939618987...|
|s3://p8-data-frui...|       Walnut|[0.15421982, 0.0,...|[0.15421982109546...|[0.77984547046876...|
|s3://p8-data-frui...|        Peach|[0.68647873, 0.25...|[0.68647873401641...|[-3.8481653083153...|
+--------------------+-------------+--------------------+--------------------+--------------------+
only showing top 5 rows
```

Stockage des résultats

- Ecriture/Lecture des résultats sur le fichier parquet

Export des résultats

- Export des résultats dans un fichier en format .csv sur S3

V- Synthèse

Contraintes	Solutions
• Volume de données qui va augmenter très rapidement après la livraison de ce projet	• Développement des scripts en Pyspark à utiliser le cloud AWS pour profiter d'une architecture Big Data (EMR, S3, IAM)
• Nécessité de compléter une chaîne de traitement de données images en Pyspark	• Mise en place d'une instance EMR opérationnelle et chaîne de traitement complétée : o Traitement de diffusion des poids du modèle Tensorflow sur les clusters (broadcast des "weights" du modèle o Etape de réduction de dimensiosn de type PCA
• Respect des contraintes du RGPD	• Paramétrage de l'installation afin d'utiliser des serveurs situés sur le territoire européen

Conclusion

- Fruits! → Mettre à disposition du grand public une application mobile qui permettrait aux utilisateurs de **prendre en photo un fruit** et **d'obtenir des informations** sur ce fruit.
- Mettre en place une **première version du moteur de classification des images de fruits**.
- Développement de l'application mobile : Construire une **première version de l'architecture Big Data** nécessaire.

Fruits!

Logo de la start-up « Fruits! »

- S'approprier les travaux réalisés par l'alternant et compléter la chaîne de traitement
- Mettre en place les premières briques de traitement qui serviront lorsqu'il faudra passer à l'échelle en termes de volume de données

OPENCLASSROOMS

MERCI DE VOTRE ATTENTION