Project 8:Déployez un modèle dans le cloud

Mitra Dadgar _ Data Scientist

Sommaire

Contexte et problématique

"**Fruits!**": start-up de l'AgriTech, cherche à proposer des solutions innovantes pour la récolte des fruits.

- Se faire connaître en mettant 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
- Cette application permettre de:
  - sensibiliser le grand public à la biodiversité des fruits et de mettre en place une première version du moteur de classification des images de fruits
  - Construire une première version de l'architecture Big Data nécessaire

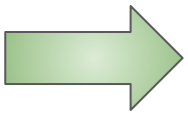
Fruits!

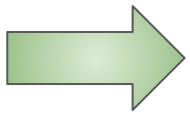
Objectifs

Mise en place d’une application de reconnaissances de fruits

Dans un premier temps :

- Extraire automatiquement des descripteurs à partir des images
- Déploiement dans une infrastructure cloud





Fruit		Benefit				
apples		Protects your heart	prevents constipation	Blocks diarrhea	Improves lung capacity	Cushions joints
apricots		Combats cancer	Controls blood pressure	Saves your eyesight	Shields against Alzheimer's	Slows aging process
artichokes		Aids digestion	Lowers cholesterol	Protects your heart	Stabilizes blood sugar	Guards against liver disease
avocados		Battles diabetes	Lowers cholesterol	Helps stops strokes	Controls blood pressure	Smoothes skin
bananas		Protects your heart	Quiets a cough	Strengthens bones	Controls blood pressure	Blocks diarrhea
beans		Prevents constipation	Helps	Lowers cholesterol	Combats cancer	Stabilizes blood sugar
beets		Controls blood pressure	Combats cancer	Strengthens bones	Protects your heart	Aids weight loss
blueberries		Combats cancer	Protects your heart	Stabilizes blood sugar	Boosts memory	Prevents constipation
broccoli		Strengthens bones	Saves eyesight	Combats cancer	Protects your heart	Controls blood pressure
cabbage		Combats cancer	Prevents constipation	Promotes weight loss	Protects your heart	Helps
cantaloupe		Saves eyesight	Controls blood pressure	Lowers cholesterol	Combats cancer	Supports immune system

Jeu de données

- 90423 images de 131 fruits (600 Mb)
- 2 jeux de données training(67692) et test set (22688)
- Labellisés
- En couleur 100px * 100px * 3 (R,G,B)
- Fond d'image éliminé (en blanc)
- Photographiés à plusieurs angles

Échantillon

- 5 fruit, 10 images
- Minimiser les coûts de stockage et traitement pendant développement

Apple_Golden				
Apricot				
Blueberry				
Kiwi				
Watermelon				

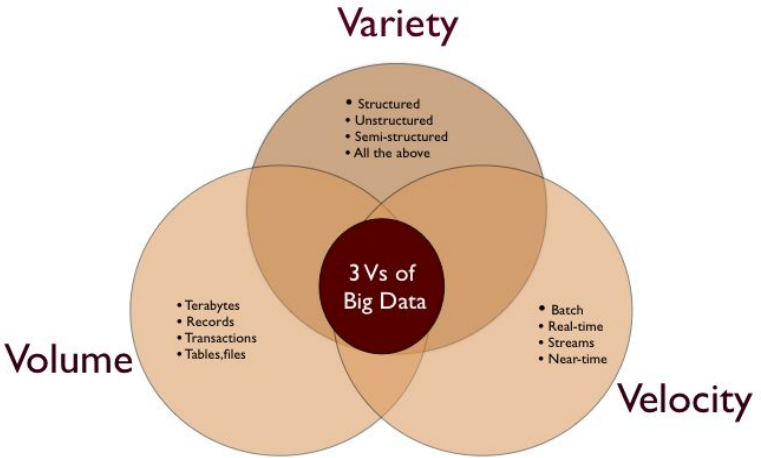
Architecture big data et les outils

Le Big data

Définition : stratégies et technologies mises en œuvre pour rassembler, organiser, stocker et analyser de vastes ensembles de données.

Les 3V du Big Data:

- ➔ **Volume** : énormes quantités de données
- ➔ **Variété** : différents types de données
- ➔ **Vélocité** : vitesse de circulation des données



Choix du prestataire cloud : AWS

Cloud Amazon Web Services:
Le prestataire le plus connu et qui offre à ce jour l'offre la plus large dans le cloud computing . Certaines de leurs offres sont parfaitement adaptées à notre problématique.

L'objectif premier est de pouvoir, grâce à AWS, louer de la puissance de calcul à la demande. L'idée étant de pouvoir, quel que soit la charge de travail, obtenir suffisamment de puissance de calcul pour pouvoir traiter nos images, même si le volume de données venait à fortement augmenter.

De plus, la capacité d'utiliser cette puissance de calcul à la demande permet de diminuer drastiquement les coûts si l'on compare les coûts d'une location de serveur complet sur une durée fixe (1 mois, 1 année par exemple).

Choix des technologies

EC2 : Elastic Compute Cloud

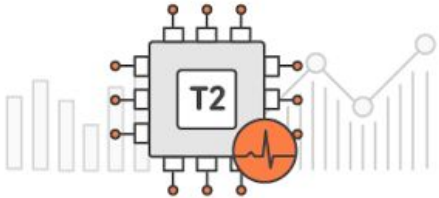
Ce service permet de gérer des serveurs sous forme de machines virtuelles dans le cloud. Serveur de développement:

- Ubuntu
- Anaconda
- Pyspark
- Via connexion SSH

S3 : Simple Storage Service

Amazon S3 (Simple Storage Service) est un service de stockage et de distribution de fichiers. C'est une sorte d'entrepôt de fichiers à très bas coût qui garantit de ne jamais perdre nos données. Utiliser pour stocker des images.

Les instances T2 Amazon EC2 sont des instances de performance à capacité extensibles qui fournissent un niveau de départ en matière de capacités de CPU, avec la possibilité de dépasser le seuil de base.



Utilisateur IAM Identity and Access Management
Création de la clé privée de l'utilisateur pour la connexion SSH au serveur EC2.

Distribution des calculs avec Pyspark

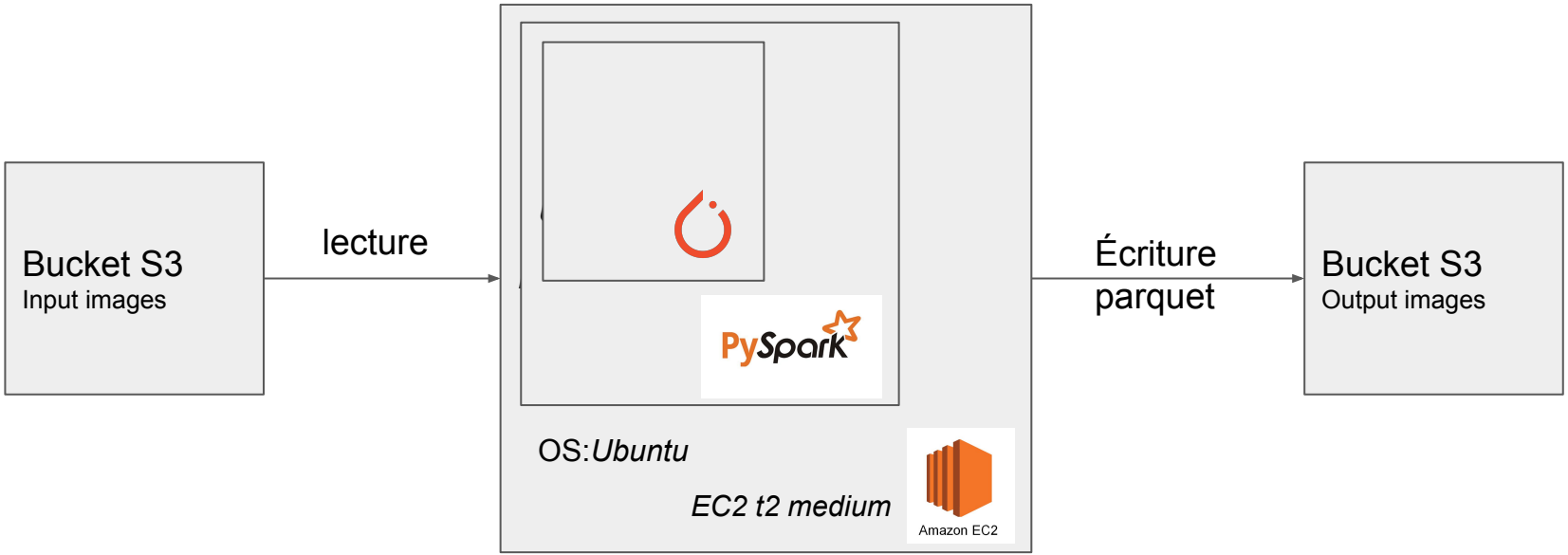
- Mise en place du formalisme Map-Reduce de Hadoop à l'aide de Spark
- Client Python PySpark pour lancer des jobs
- PySpark permet à python de s'interfacer dynamiquement avec des objets JVM à travers la librairie Py4j
- Parallélisation de l'opération de lecture et d'encodage des données
- Ecriture sous format parquet, plus optimisé pour le stockage de vecteurs

SPARK (Pyspark) :Framework open source de calcul distribué pour la parallélisation des calculs Pyspark= API python

SDK pour accéder au bucket S3 afin d'effectuer des opérations de lecture et écriture de fichiers

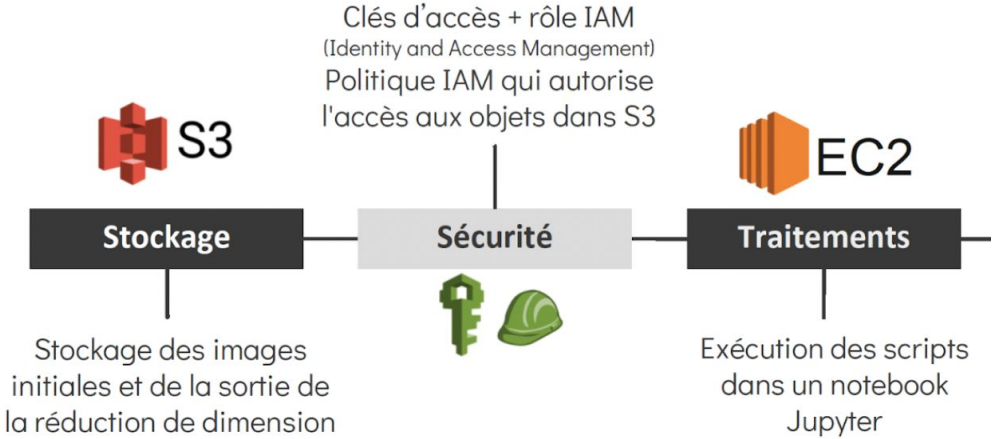
Format de fichier pour une exploitation optimisée en mode distribué conçue pour les données massives

Pipeline de traitement



Mise en place de l'infrastructure

- Configuration de la machine EC2 et bucket S3 à travers l'interface AWS
- Connexion SSH : installation dépendances SPARK
- Accès au notebook depuis l'extérieur
- Au service AWS IAM créer un nouveau rôle avec permission policies: AmazonS3FullAccess



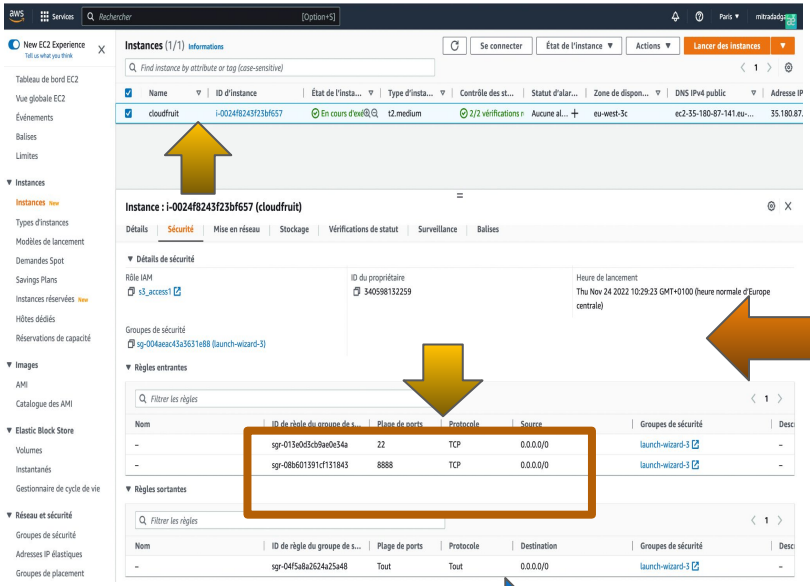
Instance EC2:T2.medium (8GO RAM, 30GO SSD) / OS Ubuntu Server 18.04
Configuration :Python 3.9.7 / Java 8 / Spark 3.2.1 / Pillow
Configuration sur machine distante : accès via SSH
• Chargement clés IAM / AWS
• Installation des logiciels et packages
• Mise en place d'un Notebook Jupyter accessible à distance contenant les scripts en Pyspark Exécutables

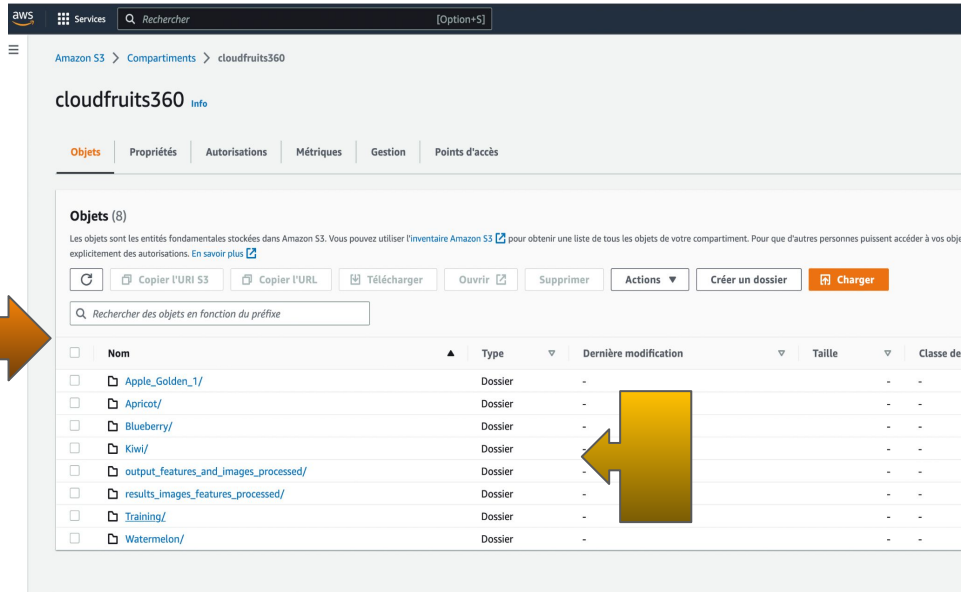
Traitement des images sur le Cloud

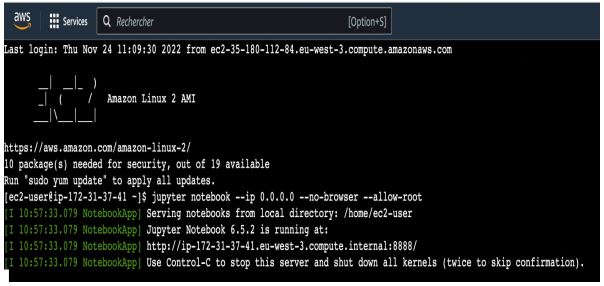
Démonstration

Les étapes à mettre en place

Transfer Learning (VGG 16): Modèle de Réseaux Neuronaux Convolutif
Pré-entraîné sur plus d'un million d'images de 1000 catégories différentes provenant de la base de données ImageNet.
Il comprend 16 couches profondes.







Instance EC2

Instances (1/1) Informations

☑	Name	ID d'instance	État de l'insta...	Type d'insta...	Contrôle des st...	Statut d'alar...	Zone de dispon...	DNS IPv4 public	Adresse IP
☑	cloudfruit	i-0024f8243f23bf657	En cours d'exé	t2.medium	2/2 vérifications r	Aucune al... +	eu-west-3c	ec2-35-180-87-141.eu-...	35.180.87.

Instance : i-0024f8243f23bf657 (cloudfruit)

Détails | **Sécurité** | Mise en réseau | Stockage | Vérifications de statut | Surveillance | Balises

▼ Détails de sécurité

Rôle IAM: s3_access1

ID du propriétaire: 340598132259

Heure de lancement: Thu Nov 24 2022 10:29:23 GMT+0100 (heure normale d'Europe centrale)

Groupes de sécurité: sg-004aeac43a3631e88 (launch-wizard-3)

▼ Règles entrantes

Filtrer les règles

Nom	ID de règle du groupe de s...	Plage de ports	Protocole	Source	Groupes de sécurité	Desc
–	sgr-013e0d3cb9ae0e34a	22	TCP	0.0.0.0/0	launch-wizard-3	–
–	sgr-08b601391cf131843	8888	TCP	0.0.0.0/0	launch-wizard-3	–

▼ Règles sortantes

Filtrer les règles

Nom	ID de règle du groupe de s...	Plage de ports	Protocole	Destination	Groupes de sécurité	Desc
–	sgr-04f5a8a2624a25a48	Tout	Tout	0.0.0.0/0	launch-wizard-3	–

BucketS3

cloudfruits360 Info

Objets (8)

Les objets sont les entités fondamentales stockées dans Amazon S3. Vous pouvez utiliser l'inventaire Amazon S3 pour obtenir une liste de tous les objets de votre compartiment. Pour que d'autres personnes puissent accéder à vos obje explicitement des autorisations. En savoir plus

☐	Nom ▲	Type	Dernière modification	Taille	Classe de
☐	Apple_Golden_1/	Dossier	-	-	-
☐	Apricot/	Dossier	-	-	-
☐	Blueberry/	Dossier	-	-	-
☐	Kiwi/	Dossier	-	-	-
☐	output_features_and_images_processed/	Dossier	-	-	-
☐	results_images_features_processed/	Dossier	-	-	-
☐	Training/	Dossier	-	-	-
☐	Watermelon/	Dossier	-	-	-

Jupyter notebook sur EC2

```
Last login: Thu Nov 24 11:09:30 2022 from ec2-35-180-112-84.eu-west-3.compute.amazonaws.com

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
10 package(s) needed for security, out of 19 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-37-41 ~]$ jupyter notebook --ip 0.0.0.0 --no-browser --allow-root
[I 10:57:33.079 NotebookApp] Serving notebooks from local directory: /home/ec2-user
[I 10:57:33.079 NotebookApp] Jupyter Notebook 6.5.2 is running at:
[I 10:57:33.079 NotebookApp] http://ip-172-31-37-41.eu-west-3.compute.internal:8888/
[I 10:57:33.079 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
```

Accèder aux images stockées sur S3 depuis le notebook Jupyter

```
import boto3
import PIL.Image

s3_bucket_name = "cloudfruits360"
image_key = "Apricot/0_100.jpg"

s3 = boto3.client('s3')
image = s3.get_object(Bucket=s3_bucket_name, Key=image_key)
PIL.Image.open(image['Body'])
```

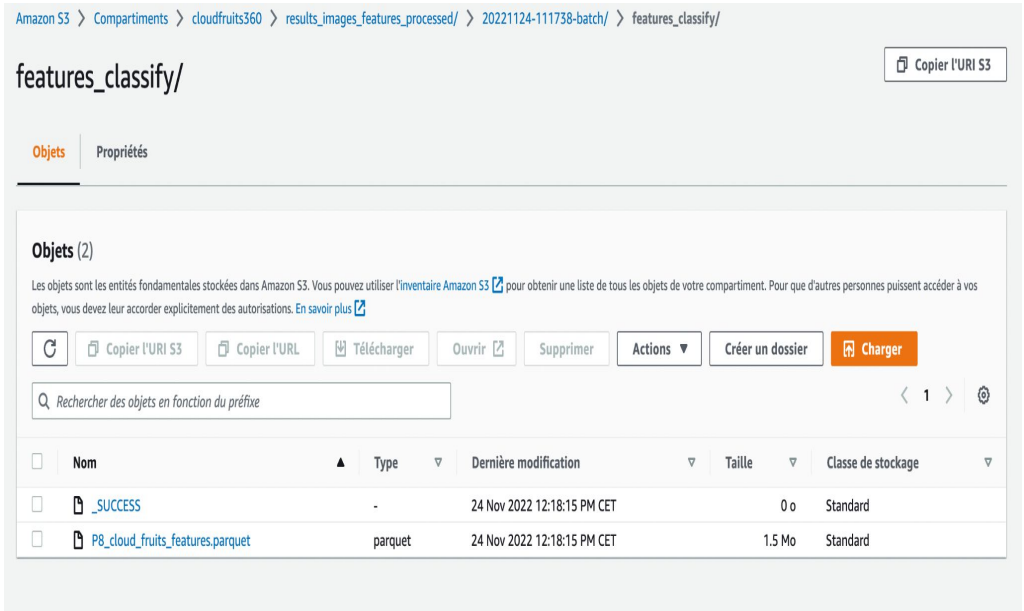
Utiliser PySpark avec accès à S3

```
s3_bucket_name = 'cloudfruits360'
s3_bucket_name_database = 'Training/'
df = spark.read.format('Image').load(f's3a://{s3_bucket_name}/{s3_bucket_name_database}/*.jpg')
df.show()
```

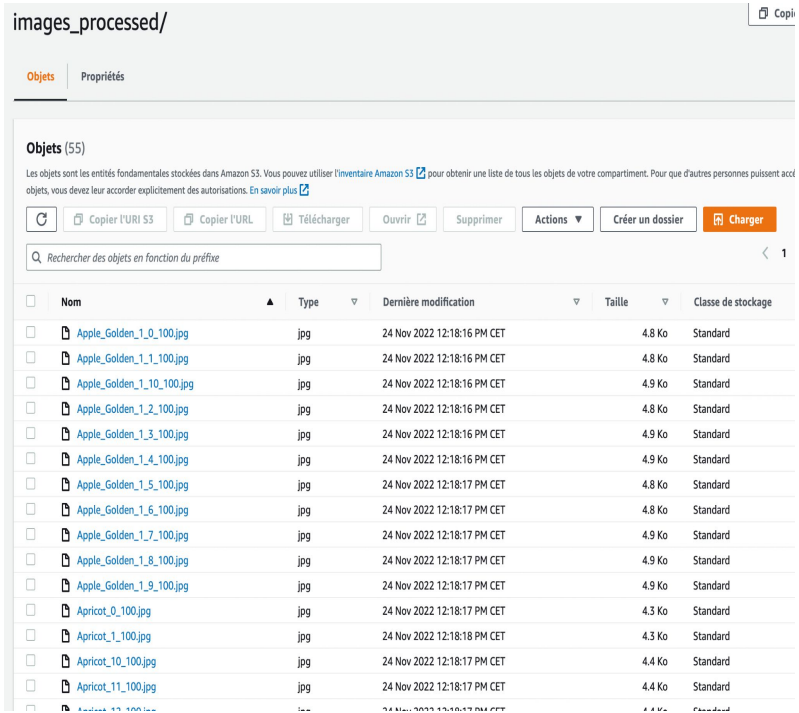
```
+--------------------+
|               image|
+--------------------+
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
|{s3a://cloudfruit...|
+--------------------+
only showing top 20 rows
```

Enregistrement des résultats sur S3

Enregistrer les features



Enregistrer les images



Enregistrement des résultats sur S3

Apple_Golden					
Apricot					
Blueberry					
Kiwi					
Watermelon					

Conclusion

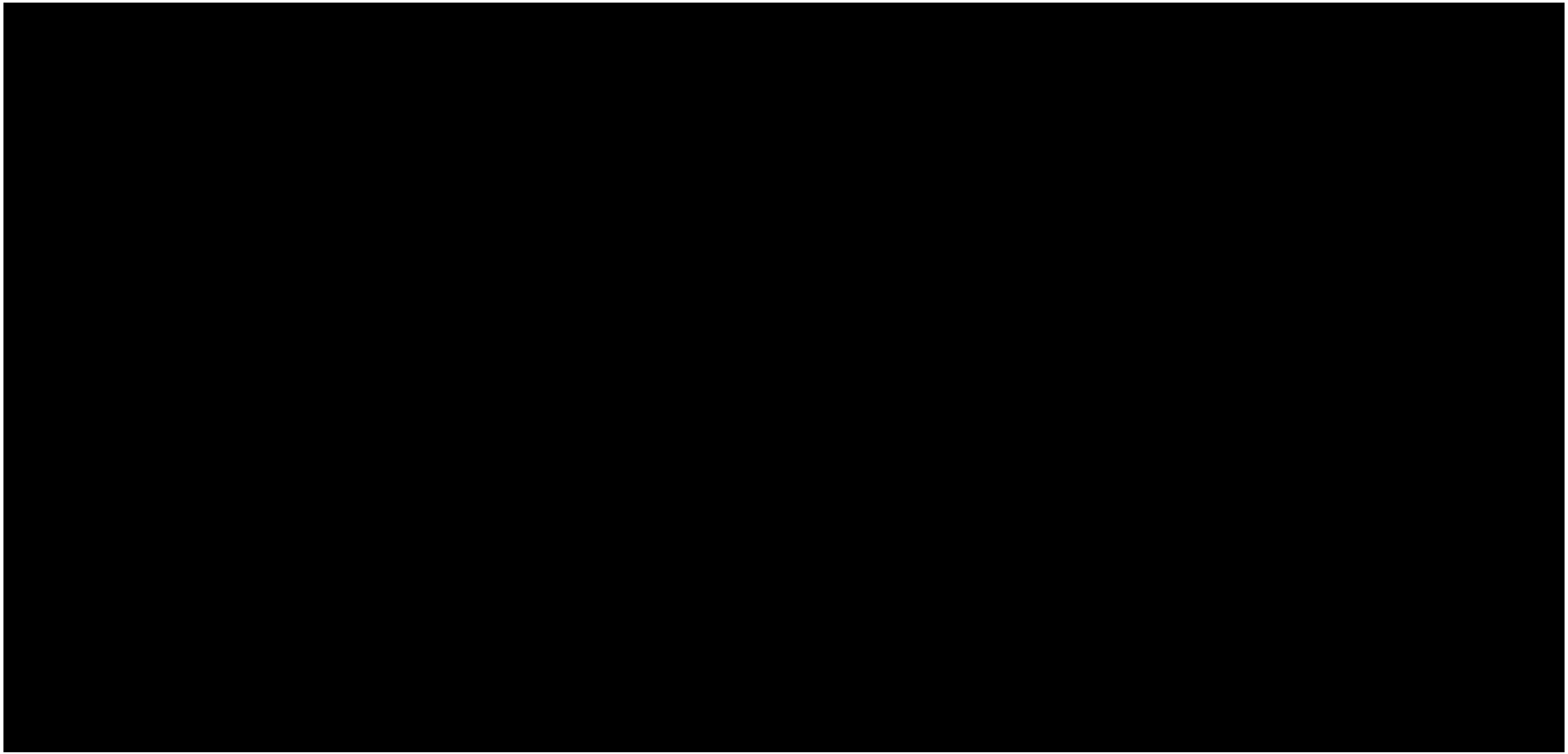
- Mise en place d'une procédure d'extraction de features automatisée
- Utilisation de RDD Map pour paralléliser la fonction d'extraction
- Parallélisable sur un cluster SPARK pour une montée en charge
- Déploiement sur une infrastructure AWS scalable

Limitations, amélioration

- Lancement manuel du job d'extraction des features
  - Mise en place d'une tâche CRON périodique
  - Trigger si des fichiers sont déplacés dans le bucket
- Infrastructure avec un seul worker, pas de distribution
  - Exploiter un cluster
  - Utilisation de Amazon EMR pour simplifier la mise en place de l'infrastructure
- Amélioration des features
  - Utilisation d'un CNN plus récent, ex EfficientNetV2

AWS

Résultats sur S3



- Pourquoi avoir choisi AWS ?
    - Avantage: Plus connu, plus adapté à notre besoin, Business focus Pas cher, l'agilité payer ce qu' on consomme, flexible(les équipements sont élastiques), pas d'engagement pas de frais de résiliation, on peut profiter plus 90 services MV, BDD,Stockage…), la sécurité , c'est programable
    - inconvénients au cloud computing : Un risque d'attaque du réseau informatique, pouvant affecter l'ensemble du système ; Une panne de connexion internet peut impacter le fonctionnement de l'organisation toute entière.
- Expliquer le choix de mon instance EC2 (t2.medium)
- Quelles pourraient être les contraintes/risques d'erreurs lors de la mise en production à l'échelle ? le choix de l'instance , avec une plus grande quantité de données, il faudra sûrement choisir une instance plus "performante" (avec un coût plus élevé...)
- Quelles ont été les difficultés rencontrées lors de la transformation de mon script Python en Pyspark ?
- Qu'est-ce qui m'a pris le plus de temps dans l'apprentissage de Spark ?
- IAM?
- Est-ce que spark peut faire des jobs parallélisés sur une seule machine? oui seulement s'il y a plusieurs processeurs. Si un seul processeur, non. mais en tout cas, ce n'est pas pratique de le faire ainsi sur le cloud car c'est très coûteux
- Comment on automatise le workflow quand le base de données s'agrandit?Amazon Simple Workflow Service (SWF)

- Sur combien d'exécuteurs j'ai fait tourner mon code ? 1 seul mais j'ai justifié en disant que mon travail n'était qu'un début pour la mise en place de l'architecture big data. il m'a dit de regarder comment configurer la session spark pour pouvoir faire tourner sur plusieurs exécuteurs et c'est passé - Comment ce que j'ai fait peut être utilisé dans le cadre d'une application globale ? Une fois qu'un utilisateur a scanné un fruit, les étapes de transfer learning et réduction de dimension sont effectuées puis il faudra rajouter une étape de prédiction et des informations sur les différents fruits et légumes.
-Différence entre spark et hadoop?La principale différence réside dans le fait que Spark est uniquement un moteur d'analyse, tandis qu'Hadoop est un système de gestion et de stockage des données. Spark n'a toutefois pas besoin d'Hadoop pour la gestion et le stockage des données.
-le lancement du notebook Jupyter depuis la ligne de commande (côté serveur): jupyter notebook --ip 0.0.0.0 --no-browser --allow-root
pourquoi j'ai utilisé le format parquet pour sauvegarder mes resultats

-Comment l'extraction des données a été sécurisée (clé utilisateurs IAM) ?
- Qu'est-ce qu'il a été mis en place pour limiter les coûts (Cost Management) ?

- pyspark est une librairie Python qui donne accès aux fonctionnalités de Spark
- Spark lui-même fait partie de l'environnement Apache, qui s'occupe des jeux de données volumineux(Big Data)
- Le principe d'Apache est d'optimiser les calculs en les répartissant entre plusieurs ordinateurs, au lieu de n'en utiliser qu'un seul
- Spark optimise encore plus les calculs en sollicitant la mémoire vive (RAM), sorte d'intermédiaire entre le processeur (qui fait les calculs), et le disque dur (qui stocke les données sous la forme de 0 et de 1)
- En réalité, on peut n'utiliser Spark que sur un seul ordi, en simulant la répartition des calculs.
- En réalité (bis), Spark n'est pas rédigé en Python, mais surtout en Scala et en Java. D'où la galère pour l'utiliser, et les messages d'erreur dégueulasses. pyspark fait la passerelle entre Python et Scala/Java,
- Une AMI est une image contenant un système d'exploitation et parfois des outils préinstallés. Toute instance EC2 est créée à partir d'AMI.
- Un volume EBS est un disque dur que l'on assigne à une instance.
- Un groupe de sécurité permet de configurer les règles firewall (pare-feu), et de limiter l'accès réseau à la machine.

---

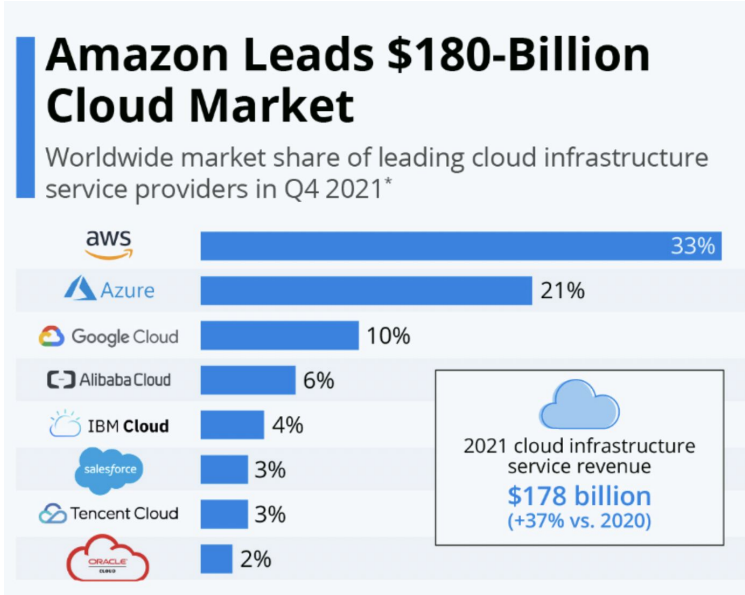
1) les extraJavaOptions semblent concerner des librairies classiques de java :
   - ➔ a) nio : gestion des entrées/sorties
   - ➔ b) lang : "Provides classes that are fundamental to the design of [...] Java"
   - ➔ c) util : "Provides utility methods that can be used [...] to perform common operations."
2) les deux dernières config fournissent des identifiants permettant à la SparkSession de se connecter aux buckets S3:
   - ➔ a) le "fs" désigne "Apache Hadoop file system commands to interact with HDFS", HDFS étant le système de données distribuées
   - ➔ b) le "s3a" fait référence aux "S3A Committers, which can commit work directly to an S3 object store."

```
def start_spark() -> SparkSession:
    spark = SparkSession.builder \
    .master("local") \
    .appName("cloudfruit") \
    .config("spark.sql.parquet.writeLegacyFormat", 'true')\
    .getOrCreate()
    return spark
```

```
spark = (SparkSession.builder
        .appName("Notebook P8")
        .config("spark.driver.extraJavaOptions",
                "--add-opens=java.base/sun.nio.ch=ALL-UNNAMED " \
                "--add-opens=java.base/java.lang=ALL-UNNAMED  " \
                "--add-opens=java.base/java.util=ALL-UNNAMED")
        .config('spark.hadoop.fs.s3a.access.key', ACCESS_KEY)
        .config('spark.hadoop.fs.s3a.secret.key', SECRET_KEY)
```

Choix du fournisseur AWS et S3

- Amazon Web Service (AWS)
- Microsoft Azure
- Google Cloud Platform (GCP)
- Apache Hadoop
- Alibaba Cloud
- IBM Cloud
- Tencent Cloud



S3

- Amazon S3 est un service AWS de stockage non structuré de fichiers bruts, appelés aussi objets.
- Le prix de stockage sur S3 le rend attractif pour y déposer des sauvegardes volumineuses, des images de site web, etc.
- Les données sur Amazon S3 sont stockées dans des compartiments (buckets). Chaque compartiment peut contenir plusieurs répertoires et fichiers.
- Un fichier sur Amazon S3 peut appartenir à différentes classes de stockage selon le niveau d'accès requis. La classe de stockage S3 Glacier Deep Archive coûte 23 moins cher que la classe "Standard", mais nécessite d'attendre quelques heures avant de récupérer le fichier.

AWS

Données client

Plateforme, applications, accès et identités

Système d'exploitation, configuration réseau et pare-feu

Cryptage des données clients et authentification de l'intégralité des données

Cryptage côté serveur (système de fichier et/ou de données)

Protection du trafic réseau (chiffrement, intégrité et identité)

Logiciel

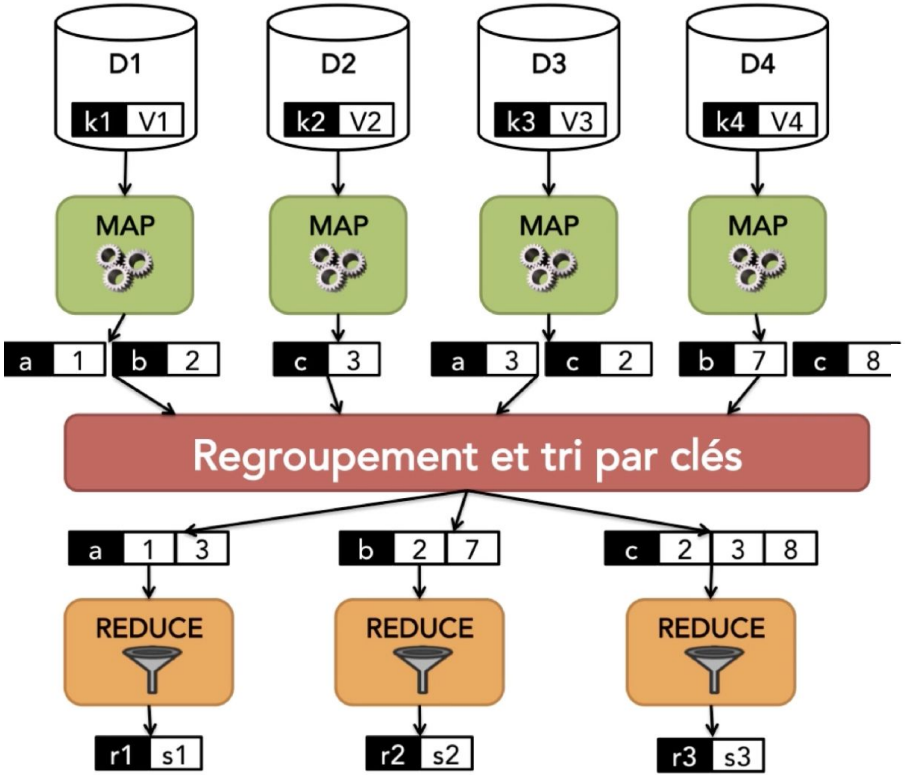
Calcul | Stockage | Base de données | Mise en réseau

Infrastructure globale

Régions | Zones de disponibilité | Localisations à la pointe

MapReduce:

- un modèle de programmation qui fournit un cadre pour **automatiser le calcul parallèle sur des données massives.** ils ont inventé un cadre générique qui permet de distribuer de manière standard un spectre d'applications restreint, mais suffisamment large pour couvrir une grande majorité des cas d'usage.



MapReduce ?

- ✧ Un modèle de conception d'algorithmes
- ✧ Un cadre générique pour la parallélisation de traitements

MapReduce : les étapes

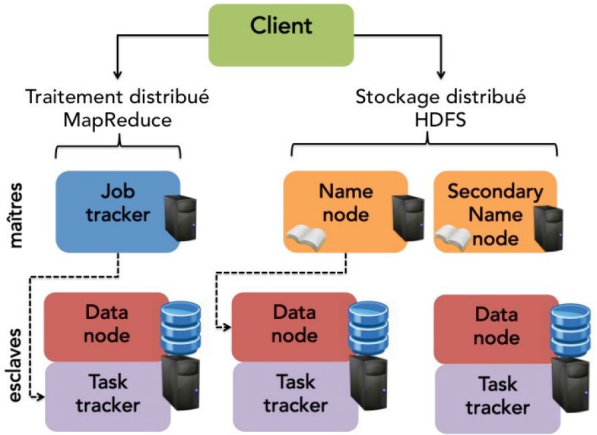
4 étapes distinctes :

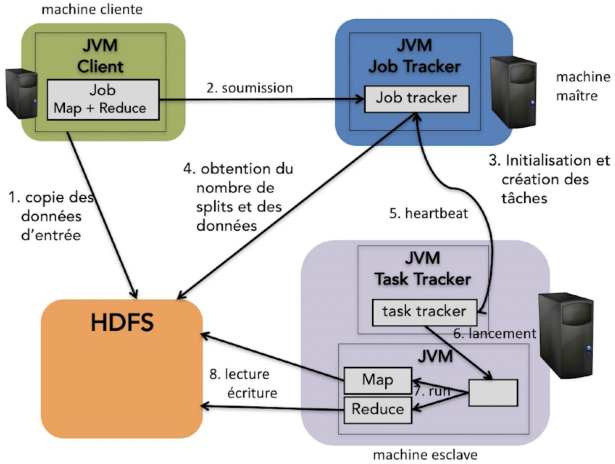
- ✧ Découper (SPLIT)
- ✧ Transformer **(MAP)**
- ✧ Grouper (SHUFFLE)
- ✧ Réduire **(REDUCE)**

Architecture Hadoop

une infrastructure logicielle dédiée qui permette d'exécuter le schéma MapReduce de manière massivement distribuée sur un cluster de machines tout en prenant à sa charge les enjeux du calcul distribué :

- l'optimisation des transfert disques et réseau en limitant les déplacements de données (*data locality*),
- la scalabilité pour permettre d'adapter la puissance au besoin (*scalability*),
- et enfin la tolérance aux pannes (*embracing failure*).
- De toute l'architecture support nécessaire pour l'orchestration de MapReduce, c'est-à-dire :
    a. l'ordonnancement des traitements,
    b. la localisation des fichiers,
    c. la distribution de l'exécution.
- D'un système de fichiers **HDFS** qui est :
    a. Distribué : les données sont réparties sur les machines du cluster.
    b. Répliqué : en cas de panne, aucune donnée n'est perdue.
    c. Optimisé pour la colocalisation des données et des traitements.





Spark et hadoop map Reduce

La principale différence réside dans le fait que Spark est uniquement un moteur d'analyse, tandis qu'Hadoop est un système de gestion et de stockage des données. Spark n'a toutefois pas besoin d'Hadoop pour la gestion et le stockage des données.

Hadoop MapReduce présente deux inconvénients majeurs :

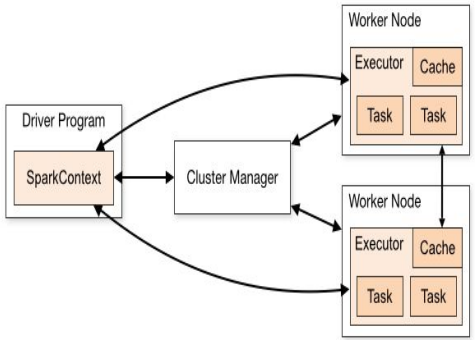
1. Après une opération map ou reduce, le résultat doit être écrit sur disque. Ce sont ces données écrites sur disque qui permettent aux mappers et aux réducteurs de communiquer entre eux. C'est également l'écriture sur disque qui permet une certaine tolérance aux pannes : si une opération map ou reduce échoue, il suffit de lire les données à partir du disque pour reprendre là où on en était. Cependant, ces écritures et lectures sont coûteuses en temps.
2. Le jeu d'expressions composé exclusivement d'opérations map et reduce est très limité et peu expressif. En d'autres termes, il est difficile d'exprimer des opérations complexes en n'utilisant que cet ensemble de deux opérations.

Apache Spark est une alternative à Hadoop MapReduce pour le calcul distribué qui vise à résoudre ces deux problèmes.

La différence fondamentale entre Hadoop MapReduce et Spark est que Spark écrit les données en RAM, et non sur disque. Ceci a plusieurs conséquences importantes sur la rapidité de traitement des calculs ainsi que sur l'architecture globale de Spark.

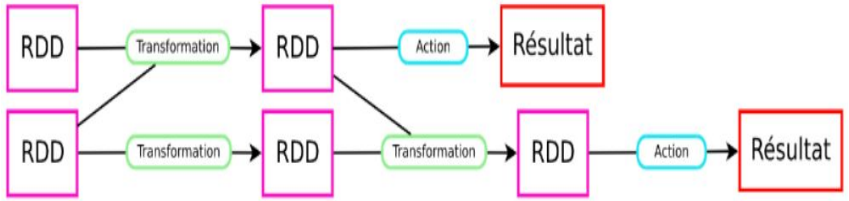
Architecture *Pyspark*

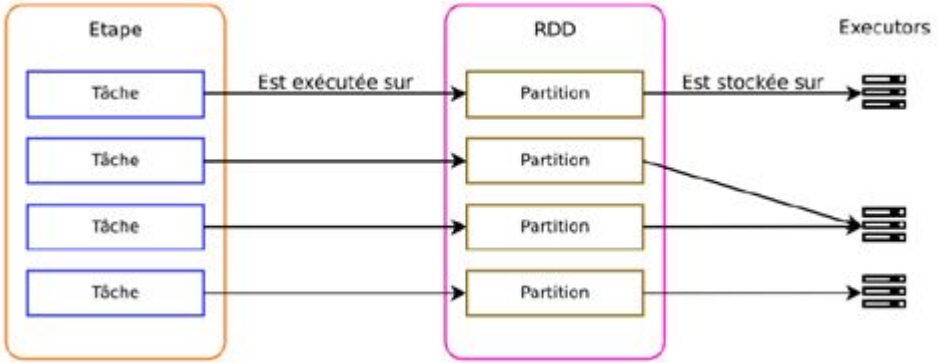
- pyspark est une librairie Python qui donne accès aux fonctionnalités de Spark
- Spark lui-même fait partie de l'environnement Apache, qui s'occupe des jeux de données volumineux(Big Data)
- Le principe d'Apache est d'optimiser les calculs en les répartissant entre plusieurs ordinateurs, au lieu de n'en utiliser qu'un seul
- Spark optimise encore plus les calculs en sollicitant la mémoire vive (RAM), sorte d'intermédiaire entre le processeur (qui fait les calculs), et le disque dur (qui stocke les données sous la forme de 0 et de 1)



Cluster de calcul (fonctionnement)

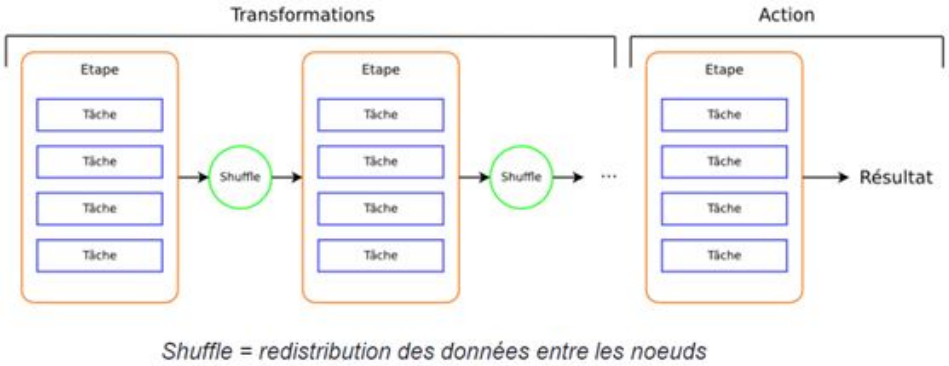
- RDD (ResilientDistributedDatasets) : principale innovation de Spark.
- Permettent d'effectuer des calculs parallèles en mémoire sur un cluster de façon complètement tolérante aux pannes
- Job Spark= ensemble d'étapes et étape = ensemble de tâches
- Chaque tâche s'exécute sur une partition différente des données et ces partitions sont crées par les RDD





Stockage : système de fichier distribué (ex : HDFS)

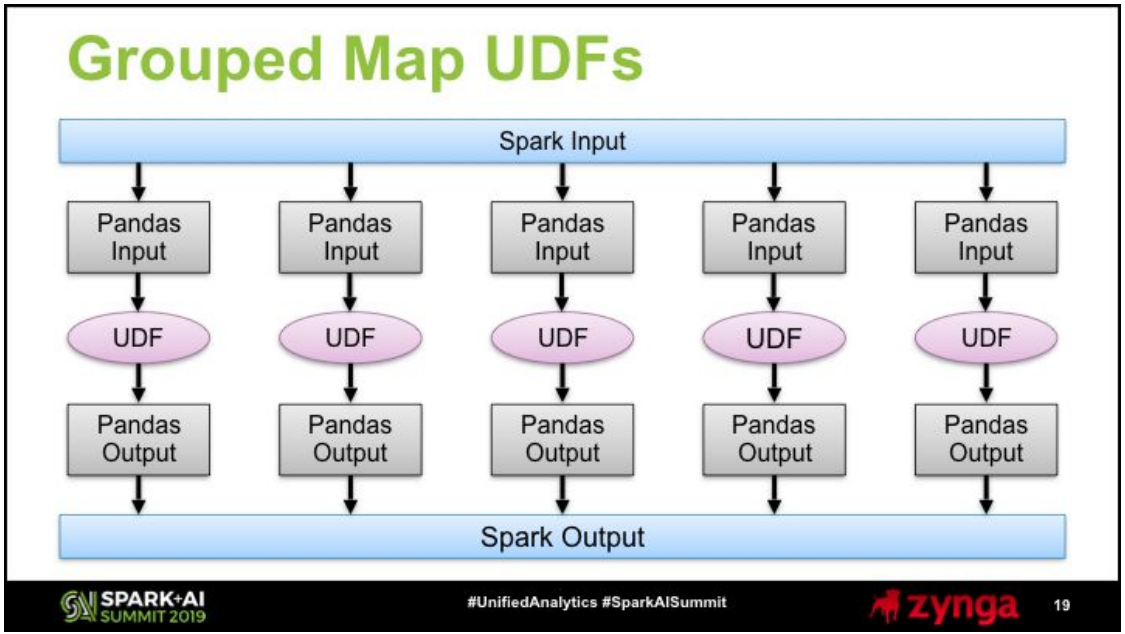
- Tolérance aux pannes
- Utilisation de Resilient Distributed Datasets (RDD)
- Division des données en partitions
- Duplication des données (3 machines par défaut)
- Graphe Acyclique Orienté (DAG) :
- Panne : Régénération à partir des noeuds parents
- Noeuds (RDD ou Résultats) : liés par des actions et transformations



Shuffle = redistribution des données entre les noeuds

UDF(user defined fonction)

Permet de créer et appliquer des fonctions non préexistantes dans spark
Pandas UDF, prendre les colonnes en entier au format pandas.series et retourne le résultat au format pandas.series



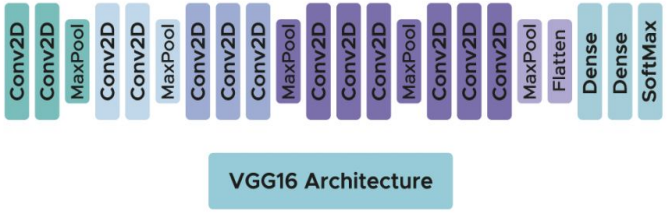
Le modèle ne demande qu'un prétraitement spécifique qui consiste à soustraire la valeur RGB moyenne, calculée sur l'ensemble d'apprentissage, de chaque pixel.

Durant l'apprentissage du modèle, l'input de la première couche de convolution est une image RGB de taille 224 x 224. Pour toutes les couches de convolution, le noyau de convolution est de taille 3×3: la plus petite dimension pour capturer les notions de haut, bas, gauche/droite et centre. C'était une spécificité du modèle au moment de sa publication. Jusqu'à VGG16 beaucoup de modèles s'orientaient vers des noyaux de convolution de plus grande dimension (de taille 11 ou bien de taille 5 par exemple). Rappelons que ces couches ont pour but de filtrer l'image en ne gardant que des informations discriminantes comme des formes géométriques atypiques.

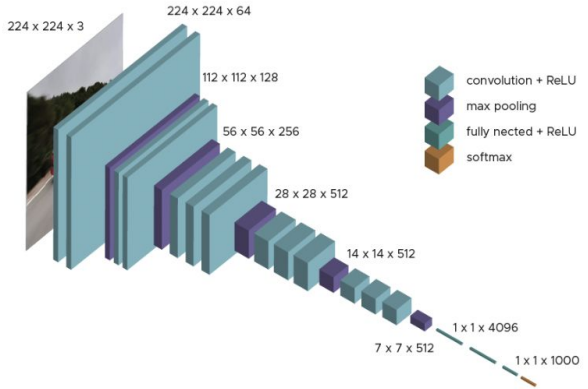
Ces couches de convolution s'accompagnent de couche de Max-Pooling, chacune de taille 2×2, pour réduire la taille des filtres au cours de l'apprentissage.

En sortie des couches de convolution et pooling, nous avons 3 couches de neurones Fully-Connected. Les deux premières sont composées de 4096 neurones et la dernière de 1000 neurones avec une fonction d'activation softmax pour déterminer la classe de l'image.

Comme vous avez pu le constater l'architecture est claire et simple à comprendre ce qui est aussi une force de ce modèle.



Architecture Algoritme VGG16



Structure Algoritme VGG16

CNN : composé de 2 couches de convolution (avec maxpooling), une couche de décrochage pour la régularisation, une couche d'aplatissement et 2 couches denses a été utilisé pour tester rapidement les pipelines de prétraitement et évaluer l'effet de la régularisation (~ 1 million de paramètres, temps d'entraînement de quelques secondes).
Pour ce problème particulier, de meilleurs résultats ont été obtenus à partir de modèles d'apprentissage en profondeur CNN, pré-entraînés sur des millions d'images

VGG16 : Le modèle pré-entraîné VGG-16 (poids ImageNet) a été utilisé pour détecter la probabilité que chaque image appartienne à une catégorie donnée.

Transfer Learning(classification supervisée) : les couches denses ont été supprimées et remplacées par une couche d'aplatissement et de nouvelles couches denses, ainsi qu'une dernière fonction softmax permettant de choisir entre les 7 catégories.

les couches de convolution ont été conservées et leurs poids pré-formés ont été gelés pour éviter de perdre les caractéristiques d'image pré-formées un réglage fin a été appliqué en ajustant uniquement les poids dans les nouvelles couches denses et softmax, tout en gelant les poids pré-formés dans la couche convolutive l'entropie croisée catégorielle a été utilisée comme fonction de perte l'algorithme d'optimisation d'Adam a été utilisé pour une optimisation rapide (une extension de la descente de gradient stochastique)