Master 1 Data Engineer/Science
Avril 2022

Architecture Distribuée

Natural Language Processing : Analyse de sentiments sur des reviews d'hôtels

F.Hammouch
S.Mahboubi
P.Zozor

Sous la direction de Alex Lima

SOMMAIRE

Introduction

Introduction

- Elaboration d'un process permettant de classifier du texte à l'aide de modèles de Machine Learning en continue

- Big Data :
  Sans API → objectif minimum Fichier plat > 500 000 lignes → Envoi séquentielle ligne/ligne

- Review Hotels : Booking.com un site néerlandais qui propose des hébergements dans différents types d'hotêls

Présentation des données

Présentation des données

- Hotel Reviews (Kaggle): 515 000 lignes
  - Split du fichier .csv Originel en 2 Fichiers .csv (80%-20%)
  - Séparation des review + et - (lignes différentes) → 849 548 lignes
  - Nettoyage du fichier (“;” , suppression de variables, des N/A)
  - Correction des mauvaises interpretations (No Negative ou No Positive)
  - Equilibrage des avis positifs et négatifs

- Dataset Training : 746 400 lignes (55% avis positifs / 45% négatifs)

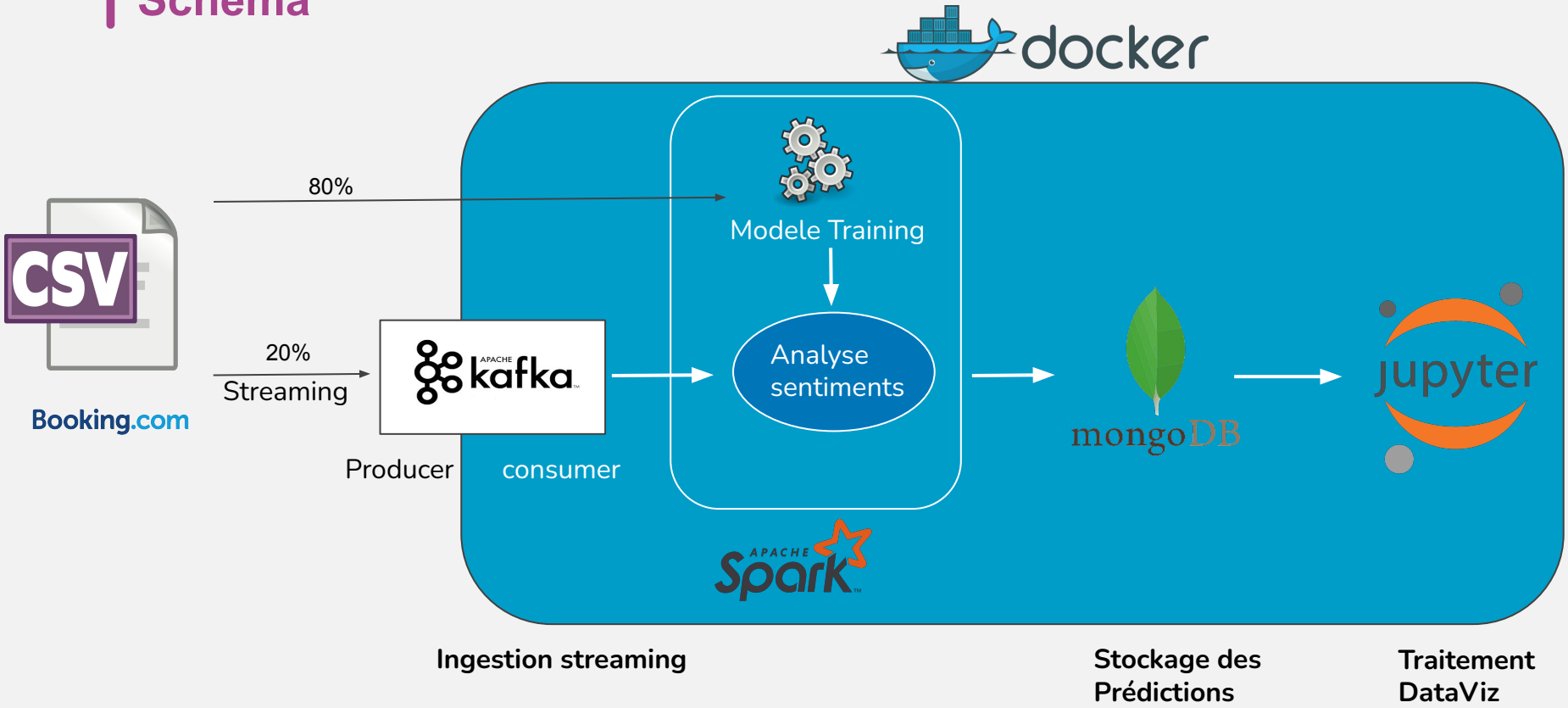
- Dataset Streaming: 103 148 lignes (55% avis positifs / 45% négatifs)

{'Hotel_Name': 'Hotel Arena', 'lat': 52.36057589999999, 'long': 4.9159682999999999, 'Average_Score': 7.7, 'Review': 'No Negative', 'Polarity': 0, 'Word_counts': 0, 'Tags': "[' Leisure trip ', ' Couple ', ' Duplex Double Room ', ' Stayed 4 nights ']"}

Architecture et environnement technique

Architecture et environnement technique

Schéma



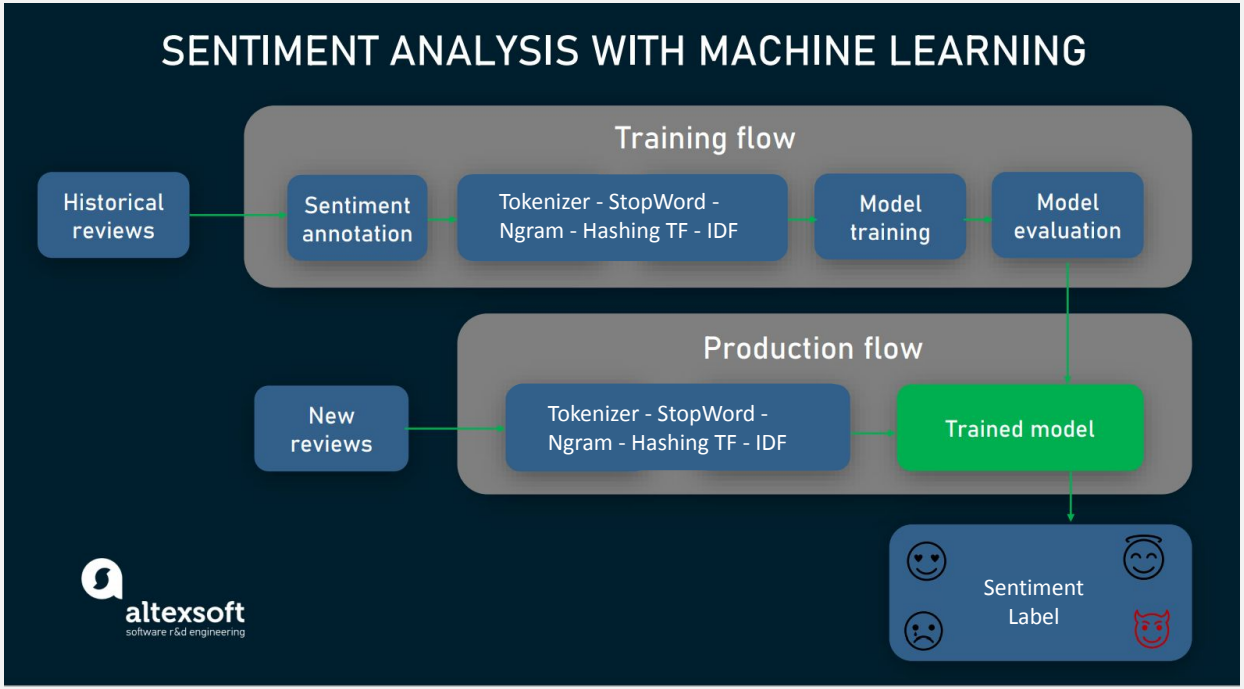
Architecture et environnement technique
Docker Compose

Image Docker:

- Pyspark-Notebook
- Spark Master
- Spark Worker
- Kafka
- Zookeper
- Mongo
- Mongo-express

Pipeline & Modèle

Pipeline et Modèle
Principe



Pipeline & Modèle

- Pipeline: Recherche des meilleurs paramètres
  - Ngram (1, 2, 3)
  - HashingTF NumFeatures

- Evaluation de 4 modèles:
  - Decision Tree Classifier
  - RandomForest Classifier
  - Gradient-Boosted Tree Classifier
  - Logistic Regression

- Optimisation du modèle de Régression Logistique
  - numFeatures à 1 million
  - Evaluation prédiction : 88%

Comparatif de la performance des modèles en fonction du paramètre numFeatures de HashingTF

Out[151]:

	DT	RF	gbt	lr
num=10	0.571381	0.567830	0.572640	0.516648
num=100	0.565484	0.560384	0.616621	0.634291
num=1000	0.538062	0.548387	0.703538	0.765896
num=10000	0.537006	0.553177	0.750337	0.849571

→ Difficultés rencontrées dans la mise en place de Grid search et de la cross validation pour l'optimisation des hyperparamètres

Démonstration

Conclusions et perspectives

Conclusion

- Obtention d’un modèle satisfaisant

- Beaucoup d’apprentissage
  - Analyse de sentiment “à la volée”
  - Architecture sous Docker
  - Kafka - Pyspark
  - PysparK - Mongo
  - Pyspark ML & Processus NLP
  - Modèles et Pipeline (save & load)

- Perspectives
  - Modèle Word2vec
  - Sauvegarde des modèles sous MongoDB
  - Une dataviz plus poussé avec Streamlight
  - Lancer le consumer dans le shell (image alpine de spark trop light)
  - Stream depuis l’API Tripadvisor, Booking (pas de retour)
  - Stemmatisation, Lemmatisation ?