

L'API

Dans le dossier API se trouve de quoi construire l'image docker qui permettra au lancement du container

D'exécuter l'interface Backend accessible aux administrateurs pour retourner leurs requêtes. Le dossier contient :

- Un dockerfile.api
- Un requierment
- Le fichier api_sentiment_analysis.py

Streamlit

Dans le dossier Streamlit se trouve de quoi construire l'image docker qui permettra au lancement du container

D'exécuter l'interface frontend accessible aux utilisateurs pour exécuter leurs requêtes. Le dossier contient :

- Un dockerfile.streamlit
- Un requierment
- Le fichier Streamlit_satisfaction.py qui contient le code Streamlit
 - o Le streamlit interrogera l'API par le biais de requête

L'ouverture et la fermeture du container Streamlit sera lié à celle du container api par la commande dependsOn dans le docker compose

MySQL

Dans le dossier MySQL se trouve de quoi construire l'image docker qui permettra au lancement du container

'de mettre à disposition une BDD mySQL qui stockera les users et mot de passe utilisateurs. Le container sera directement relié à l'API.

Le container sera relié à un volume local

Scrapping

Dans le dossier scrapp/scrapping on trouve de quoi construire l'image docker qui contient :

- Un dockerfile.scrapp
- Un requierment
- Un fichier last30days.py
- Les clefs d'accès à AWS S3

Et qui permettra au lancement du container de :

- Charger la base de données trustpilot_comment.csv stockée dans AWS stocker en tant que bucket S3
- Scraper les donner de 7 entreprises : ubaldi, habitatjardins, cdiscount, menzzo, fnac, darty, temu
- Les preprocesser
- Ne garder dans ce fichier nommé last30days que les avis ayant une date supérieure à celle de la base de données
- Les ajouter à la base de données trustpilot_comment.csv
- Enregistrer ce fichier dans le bucket S3 en remplacement de celui existant
- Enregistrer le fichier last30days dans le bucket S3 en remplacement de celui existant

Preprocessing

Qui permet de transformer les données scrapper en un format utilisable pour appliquer les modèles de NLP

Contenu dans le dossier scrapp/preprocessing on trouve de quoi construire l'image docker qui contient :

- Un dockerfile.preprocessing
- Un requierment
- Un fichier last30days_retired.py
- Les clefs d'accès à AWS S3

Et qui permettra au lancement du container de :

- Charger les fichiers trustpilot_comment_retired.csv, trustpilot_comment_retired_0_1.csv et last30days.csv depuis AWS S3
- Les processor
- Traiter à l'aide de nltk les 3 fichiers
- Ajouter last30days_retired à trustpilot_comment_retired.csv
- Puis Ajouter last30days_retired_0_1 à trustpilot_comment_retired_0_1.csv
- Enregistrer les fichiers trustpilot_comment_retired.csv, trustpilot_comment_retired_0_1.csv dans le bucket S3 en remplacement de ceux existant.

Models_reload-2

Dans le dossier Models_reload-2 sont présents les fichiers pythons pour :

- Le réentraînement des 2 modèles utilisés dans l'API LSMT et RNN
 - o Pour les notes
 - o Pour les sentiments négatifs et positifs
- Les prédictions pour les 2 types de modèles issus de ces réentraînements utilisés dans l'API LSMT et RNN
 - o Pour les notes
 - o Pour les sentiments négatifs et positifs