Rapport de mise en production

Projet : Amazon Review Analysis

Auteur : Dyhia TOUAHRI

Date : 10 décembre 2025

Version 1.0

Table des matières

1. Introduction

1.1. Contexte & Objectif

Le projet a pour objectif de déployer en environnement de production un système complet d'analyse d'avis clients Amazon, incluant l'extraction, la transformation, le stockage, l'analyse NLP, et la mise à disposition des résultats pour des applications analytiques. L'objectif de cette mise en production est de garantir que le pipeline fonctionne de façon automatisée, fiable et scalable, avec des garanties de qualité des données, de sécurité, et de performance.

1.2. Portée de la mise en production

La mise en production couvre l'ensemble des composants backend (bases de données, entrepôts, stockage), du pipeline ETL, des composants ML/NLP, du mécanisme d'anonymisation, et de l'infrastructure nécessaire. Elle ne couvre pas, dans un premier temps, des extensions (comme le traitement temps réel des avis, ou le traitement d'images).

1.3. Conclusion et recommandation

Après validation des tests, de l'architecture et des performances, le passage en production est recommandé. Le système répond aux exigences fonctionnelles, techniques et de qualité définie, avec des marges sur la scalabilité. Le présent rapport documente les bases pour cette décision, les risques identifiés, et les conditions d'activation.

2. Périmètre et objectifs de la mise en production

2.1. Périmètre fonctionnel

- Extraction automatisée des données sources (PostgreSQL).
- Anonymisation des données sensibles (buyer_id).
- Stockage brute (AWS S3).
- Chargement des données transformées vers le data warehouse (Snowflake).
- Stockage des logs et des rejets dans la base documentaire (MongoDB).
- Orchestration des tâches ETL via Apache Airflow.
- Exécution des traitements NLP & scoring de pertinence des avis (classification thématique, sentiment, scoring).
- Génération de données enrichies dans Snowflake.
- Interface de visualisation (dashboard Streamlit App dans Snowflake).
- Mis à niveau de l'application Streamlit App des Business Analystes.
- Gestion automatisée des jobs via Docker / Docker-Compose.

2.2. Objectifs de qualité & de performance

- Qualité des données : respect des règles de validation (types, non-nullité, cohérence, absence de doublons).
- Sécurité : anonymisation des données sensibles, gestion sécurisée des crédenciales, séparation des environnements.
- Automatisation complète : déclenchement journalier (ou périodique) du pipeline sans intervention manuelle.
- Scalabilité : capacité à traiter des volumes de données plus importants que la volumétrie actuelle.
- Observabilité & maintenance : logs, monitoring, traçabilité des rejets et des erreurs.

3. Inventaire des composants techniques

Couche	Composant / Technologie	Version / Informations
Source des données	PostgreSQL (docker)	PostgreSQL 17
Stockage « brut »	AWS S3 (objet)	Bucket configurable via variables d'environnement
Data Warehouse analytique	Snowflake	Instance Cloud, schéma dédié
Stockage des logs / rejets	MongoDB (docker)	MongoDB 7.0
Orchestration	Apache Airflow	2.8.3 (Docker)
Conteneurisation & Orchestration d'infrastructure	Docker & Docker-Compose	Version Docker compatible 24+ / Compose
Traitements & transformations	Python 3.11+, pandas, boto3, snowflake-connector	Bibliothèques Python standard
NLP / ML / Scoring	Transformers (mDeBERTa), NLTK	Versions compatibles (transformers ≥ 4.x, torch ≥ 2.x) + d'autres librairies (pandas, numpy, seaborn…)
Visualisation	Streamlit App Snowflake	Instance Cloud, schéma dédié
Configuration & credentials	Variables d'environnement (.env)	AWS, Snowflake, autres services

4. Architecture technique et flux de données

- Les données sources sont dans PostgreSQL.
- Via Airflow, un job ETL est déclenché :
  - Extraction des données → anonymisation →transfert vers S3 (data lake).
- Les données sont ensuite transformées puis chargées dans Snowflake pour constituer l'entrepôt analytique.
- Les logs, rejets et erreurs sont stockés dans MongoDB pour audit et monitoring.
- Pour l'analyse des avis :
  - Extraction depuis Snowflake → application des modèles NLP (classification, sentiment) → calcul de score de pertinence → enrichissement des données dans Snowflake.
- L'ensemble s'appuie sur des containers Docker, assurant portabilité et isolation.
- Le pipeline est automatisé, modulable, et peut être re-joué à volonté.

5. Procédure d'installation et de configuration

5.1. Pré-requis & préparation

- Serveur (VM) compatible Docker / Docker Compose, avec ressources suffisantes (CPU, RAM, stockage).
- Accès réseau / droits vers AWS S3 et Snowflake.
- Fichier .env correctement configuré avec les credentials AWS, Snowflake, et autres variables de comptes de service.

5.2. Démarrage des services

1. Lancer PostgreSQL (docker-compose) pour la base source.
2. Lancer MongoDB (docker-compose) pour les logs/rejets.
3. Lancer Airflow (docker-compose) pour l'orchestration.

5.3. Déclenchement du pipeline

- Via l'interface Airflow ou commande CLI, déclencher le DAG principal pour exécution complète.
- Vérifier les logs pour chaque étape (extraction, anonymisation, stockage S3, transformation, chargement, logs).

5.4. Configuration des accès & sécurité

- Stockage sécurisé des credentials (fichier .env, accès restreint).
- Réglage des permissions d'accès base de données, S3, Snowflake, MongoDB selon RBAC.

- Mise en place des bonnes pratiques Docker (réseau, volumes, isolation).

6. Rôles et habilitations des utilisateurs

Dans le cadre de la mise en production, les rôles utilisateurs suivants ont été définis conformément aux besoins fonctionnels et techniques :

Rôle	Description	Habilitations principales
Admin Data Platform	Supervision complète de la plateforme	Accès complet à Airflow, Snowflake, S3, MongoDB
Data Engineer	Exécution et maintenance du pipeline	Exécution DAGs, écriture S3, chargement Snowflake
Data Scientist	Analyse NLP et enrichissement	Lecture Snowflake, exécution traitements NLP
Business Analyst	Consultation des tableaux de bord	Lecture seule dans Snowflake + accès Streamlit
Ops / Exploitation	Supervision et monitoring	Accès aux logs MongoDB + lecture Airflow

Les habilitations respectent le principe du moindre privilège (Least Privilege Access) et ont été configurées dans :

- Snowflake (rôles : SYSADMIN, ETL_ROLE, DATA_SCIENTIST, BUSINESS_ANALYST)
- Airflow (profils : Admin, User, Viewer)
- AWS S3 (permissions IAM adaptées par rôle)
- MongoDB (droits lecture/écriture par profil)

7. Tests et validation en pré-production

Avant la mise en production, les phases suivantes ont été menées :

- Tests de connectivité : vérification des connexions à PostgreSQL, S3, Snowflake, MongoDB.
- Tests d'intégrité des données : validation des contraintes (non-null, types, valeurs, absence de doublons).
- Tests de bout en bout du pipeline : exécution complète — extraction → anonymisation → stockage → transformation → chargement → logs.
- Tests de sécurité : anonymisation effective des données sensibles, sécurisation des credentials.
- Tests de stabilité : exécution multiple, redémarrage, gestion des erreurs.

Les résultats ont validé le bon fonctionnement du pipeline, l'intégrité des données, et la robustesse de l'installation.

8. Activation des interfaces avec les outils du SI

Les interfaces suivantes ont été activées et testées :

- PostgreSQL → Airflow : connexion configurée et extraction validée
- Airflow → AWS S3 : droits IAM opérationnels et export anonymisé validé
- Airflow → Snowflake : chargement automatique des données validé
- Airflow → MongoDB : stockage des logs et rejets validé
- Snowflake → Snowflake Streamlit App: lecture des données enrichies validée
- Snowflake →Application Streamlit : lecture des données enrichies validée

Chaque interface a été vérifiée via :

- Tests de connectivité
- Tests de transfert
- Logs d'exécution
- Validation finale des données

Attestation de bon fonctionnement de l'architecture

À l'issue de l'ensemble des tests, il est attesté que :

- Le pipeline ETL complet fonctionne sans erreur
- Les interfaces inter-systèmes sont actives et stables
- Les données sont correctement anonymisées
- Les modèles ML tournent correctement

- Snowflake reçoit et expose les données enrichies
- Le dashboard est opérationnel
- L'application Streamlit est opérationnelle
- La sécurité et les accès sont conformes

L'architecture est donc opérationnelle et prête pour la mise en production.

9. Analyse de performance et montée en charge

Sur la volumétrie actuelle, les indicateurs sont les suivants :

- Temps d'exécution complet du pipeline (extraction + transformation + chargement) : acceptable moins de 5 mins sur ~111K de
- Utilisation CPU / mémoire raisonnable sur l'infrastructure Docker.
- Temps d'accès aux données dans Snowflake conforme aux attentes pour les requêtes analytiques.
- Capacité de montée en charge : réserve suffisante pour accroître le volume d'avis sans impact majeur — l'architecture est modulaire et scalable.

Des tests de montée en charge (scaling data volume ×5 et ×10) sont planifiés pour valider la capacité future.

10. Risques identifiés et mesures d'atténuation

Risque	Impact potentiel	Mesure d'atténuation
Erreur de configuration des credentials	Échec du pipeline	Validation des .env, contrôle des accès, rotation régulière des clés
Anonymisation insuffisante	Violation de la vie privée / conformité	Vérification des scripts, audits, tests PII
Surcharge infrastructure (CPU, I/O) en cas de montée en volume	Dégradation performance / échec pipeline	Plan de montée en charge, ressources réservées, monitoring système
Perte de données / logs	Perte d'historique, difficulté debug	Sauvegarde régulière, politique de backup, redondance
Bugs dans les transformations / ML	Données incorrectes, classification erronée	Tests automatisés, validation qualité, revue des résultats

11. Procédure de Rollback

La procédure de retour arrière a été définie afin de garantir la capacité de rétablir l'environnement dans un état stable en cas d'incident critique survenant après le déploiement en production. Cette procédure couvre l'ensemble des composants techniques impliqués dans le pipeline de traitement des avis Amazon.

CAS 1 : première mise en production

Etant donné qu'il s'agit une nouvelle solution qui n'a pas encore été mise en production, la partie impactée est juste l'application Streamlit, une version antérieure de l'application avec l'ancienne configuration a été mise dans une branche github « rollback_streamlit » dans le cas où nous rencontrons des problèmes, nous aurons la possibilité de récupérer la dernière version. Ceci garantit que l'application reste opérationnelle pour les utilisateurs.

CAS 2 : mise à jour en production

Pour toute prochaine évolution de la solution analytique (cas où elle est déjà en production), voici le plan d'action :

1. **Snowflake :**

Le rétablissement des objets Snowflake repose sur la fonctionnalité **Time Travel**, permettant de restaurer tables, schémas, entrepôts et données dans un état antérieur.

Étapes :

1. Identifie l'heure précise de déclenchement de l'incident.
2. Restauration des tables via commande sql.
3. Vérification de l'intégrité des données restaurées.
4. Remplacement contrôlé des tables en production.

2. **S3**

En cas d'erreur dans les données transférées ou anomalies dans les fichiers bruts :

1. Consultation de la liste de versions des objets dans le bucket.
2. Restauration de la version précédente des fichiers concernés.
3. Validation de la cohérence avec Snowflake ou le pipeline amont.

3. **Airflow**

Si un DAG déployé introduit une régression :

1. Suppression du DAG problématique depuis le scheduler.
2. Restauration de la version précédente du fichier DAG depuis le dépôt Git.
3. Redémarrage du scheduler et validation des dépendances.

4. **Docker**

En cas d'anomalie liée à un container, une bibliothèque ou une dépendance :

1. Arrêt des services via docker-compose.
2. Relance avec l'image stable précédente :
   docker compose -f docker-compose.yml --env-file .env up -d –build

11.1. Conditions de déclenchement

Un rollback peut être déclenché en cas de :

- ✓ Corruption de données détectée dans Snowflake,
- ✓ Rupture du pipeline rendant impossible la production des données,
- ✓ Incapacité à accéder aux données depuis l'application Streamlit,
- ✓ Problèmes de performance majeurs et persistants.

11.2. Durée estimée

Le retour arrière complet peut être réalisé en 1 heure, sous réserve d'une disponibilité de l'équipe technique.

12. Politique de sécurité et conformité

La mise en production du pipeline doit respecter les règles de sécurité informatique, les bonnes pratiques cloud, ainsi que les exigences de conformité, notamment vis-à-vis de la protection des données personnelles.

12.1. Séparation des environnements

Le système est structuré en trois environnements :

- Développement : tests unitaires et évolutions des DAGs.
- Pré-production : tests finaux, jeux de données complets, validations fonctionnelles.
- Production : données réelles + accès restreints.

Chaque environnement dispose :

- De ses propres identifiants,
- D'un bucket S3 dédié,
- D'un schéma Snowflake distinct.
- D'une application Streamlit selon l'environnement.

12.2. Gestion des accès (IAM & RBAC)

L'ensemble des droits utilisateurs repose sur une méthodologie RBAC (Role-Based Access Control).
Les permissions sont limitées au strict nécessaire selon le principe du moindre privilège.

AWS S3

- Rôles distincts pour Compte de service, Data Engineer, Data Scientist.
- Permissions limitées : s3:GetObject, s3:PutObject, s3:ListBucket.
- Accès chiffrés en transit et au repos.

Snowflake

Tous les utilisateurs appartiennent à des rôles hiérarchisés :

- SYSADMIN : administration des schémas.
- ETL_ROLE : chargement et transformation.
- DATA_SCIENTIST : accès lecture/écriture sur tables du schéma ANALYTIC.
- BUSINESS_ANALYST : lecture seule.

Chaque action sur Snowflake est tracée via les audits logs internes.

MongoDB

Les accès sont segmentés :

- Un compte lecture pour Ops.
- Un compte écriture utilisé uniquement pour le compte de service.
- Un compte administrateur isolé.

12.3. Conformité RGPD

Le seul identifiant sensible présent dans la donnée source est le buyer_id.
Pour répondre aux exigences RGPD :

- Le champ est anonymisé via un hash cryptographique non réversible (SHA-256 + salt),
- Aucun identifiant nominatif n'est stocké dans S3,
- Snowflake ne contient aucune donnée permettant d'identifier un utilisateur Amazon.

Des audits réguliers sont réalisés pour garantir l'absence de données à caractère personnel.

12.4. Sécurité des communications

Toutes les communications inter-services utilisent TLS :

- Connexion Airflow ↔ Snowflake via le connecteur sécurisé,
- Connexion Python ↔ S3 via protocole HTTPS signé AWS,
- Connexion MongoDB protégée par mot de passe fort et port non exposé publiquement.

14.5. Gestion des credentials

Les identifiants nécessaires au fonctionnement du pipeline sont stockés dans :

- un fichier .env non versionné,
- des volumes non exposés,
- avec des permissions Linux restreintes.

Une rotation trimestrielle des clés AWS et Snowflake est effectuée.

13. Go/No-Go : Checklist de mise en production

Critères indispensables pour donner le GO

- ✓ Pipeline complet validé avec succès sans erreur critique.
- ✓ Données anonymisées correctement, conformité RGPD respectée.
- ✓ Performances acceptables sur la volumétrie de production.
- ✓ Infrastructure stable, scalable, avec ressources suffisantes.
- ✓ Accès et sécurité correctement configurés (credentials, permissions).
- ✓ Documentation et procédures d'exploitation disponibles.

Conditions de refus (NO-GO)

- ✓ Échec de tests critiques (connectivité, anonymisation, intégrité).
- ✓ Performance insuffisante (temps d'exécution trop long, surcharge).
- ✓ Risque de sécurité ou de conformité non traité.
- ✓ Absence de plan de maintenance / backup / monitoring.