Spécifications techniques et fonctionnelles

Projet : Analyse des avis produits

Auteur : Ismaël SYLLA

A propos de ce document

Ce document reprend les spécifications fonctionnelles et techniques du projet « Analyse des avis produits ». Il documente le périmètre, les contraintes et l'architecture de la solution afin d'assurer une compréhension commune entre les besoins fonctionnels et les choix techniques.

Les spécifications citées dans ce document se base sur un prototype qui s'est inscrit dans une démarche exploratoire visant à démontrer la faisabilité technique de ce projet.

Table des matières

1. Contexte du projet

Le projet Amazon « Analyse des avis produits » vise à développer une solution technique permettant d'analyser la qualité et la pertinence des avis produits publiés sur la plateforme e-commerce d'Amazon.

Les données utilisées proviennent d'un dataset d'avis Amazon contenant pour chaque produit des informations telles que la note, le texte du commentaire et l'identifiant utilisateur et d'autres informations.

Les spécifications fonctionnelles et techniques développées dans ce document se base sur un prototype réalisé afin d'évaluer la faisabilité du projet.

2. Objectifs et périmètre du projet

2.1. Objectifs

Les objectifs de ce projet sont :

- Améliorer la qualité, la fiabilité et la valorisation des données issues des avis produits Amazon, en garantissant leur intégrité et leur pertinence pour les analyses métiers.
- Collecter, stocker et traiter automatiquement les données des avis produits Amazon puis les transformer et les enrichir afin de mettre en avant les avis les plus pertinents et cela en appliquant un algorithme de scoring et des fonctions de pondération.
- Identifier les avis des clients à forte valeur ajoutée.
- Offrir une visibilité claire et automatisée sur les indicateurs de qualité et de performance via des tableaux de bord, permettant un suivi continu et une prise de décision éclairée.
- Mettre en place un cadre de gouvernance et de gestion de qualité de données, évolutif et maintenable et aligné sur les de bonnes pratiques.
- Construire une solution qui respectent les réglementations en vigueur.

2.2. Périmètre du projet

Dans cette section nous abordons le périmètre du projet, pour rappel la solution qui est proposée se base sur une analyse faite grâce à un prototype.

La solution sera développée de façon à pouvoir permettre de futures évolutions facilement, ce qui est exclus aujourd'hui du périmètre peut être ajouté dans une future version de l'application.

2.2.1. Inclus dans le périmètre

- Extraction et importation d'un jeu de données des avis produits Amazon.
- Profilage, nettoyage et validation des données afin d'assurer leur qualité et leur cohérence.
- Définition et exécution de règles de qualité pour détecter les anomalies et incohérences.
- Transformation et stockage des données traitées.
- Analyse, catégorisation et calcul d'un score pondéré de confiance sur les avis clients pour déterminer leur pertinence.
- Intégration d'un système de suivi et d'alertes sur les indicateurs clés de qualité.
- Restitution des indicateurs et résultats via des rapports visuels et interactifs.

2.2.2. Exclus du périmètre

- Développement d'une interface web ou d'une API utilisateur.
- Intégration en temps réel des données (traitement uniquement en batch).
- Contrôle qualité en continu (le suivi se limite à des traitement planifiés).

2.2.3. Contraintes clés

- Temps : projet limité à une période académique (2 semaines par projet) pour réaliser, tester et documenter le projet.
- Budget : aucun budget alloué.
- Ressources humaines : équipe junior. Pas d'experts métiers/techniques dans l'équipe.
- Techniques : favorisation d'outils gratuit/open source.

3. Contraintes, risques et éthiques

Cette section présente les principales contraintes du projet, les risques identifiés ainsi que les aspects éthiques et réglementaires à prendre en compte dans sa mise en œuvre.

3.1. Contraintes techniques

- Jeu de données : étant donné que nous sommes dans un cadre académique, les données sont statistiques et non évolutives, et limité en volume. Ce qui peut restreindre la représentativité des analyses.
- Infrastructure : le projet s'exécute en local, sans recours à un environnement cloud (sauf pour la partie stockage bucket S3).
- Performance : le traitement reste manuel et dépend directement des capacités de la machine locale utilisée.
- Outils : S'assurer à chaque étape de la réalisation du projet que les composants utilisés sont compatibles entre eux (ex : librairies compatibles avec la version python utilisée).

3.2. Contraintes humaines

- Ressources : les acteurs qui travaillent sur le projet sont des apprentis développeurs, ce qui implique une phase d'adaptations aux concepts conséquente.
- Compétences : une montée en compétence est nécessaire à chaque phase du projet pour mener à bien certaines analyses.
- Temps : le projet s'inscrit dans un cadre académique limité dans la durée, nécessitant une priorisation rigoureuse des tâches.

3.3. Risques et mitigation

Risque	Impact	Probabilité	Mesure d'atténuation
Données incompletes ou biaisées	Élevé	Élevé	Nettoyage rigoureux, validation manuelle sur des échantillons.
Mauvaise pondérations	Élevé	Moyen	Ajustement des coefficients après tests.
Problème de compatibilité PostgreSQL–Python	Moyen	Moyen	Test des versions et recherche de la documentation.
Non-respect RGPD (informations personnelles non anonymisées)	Critique	Moyen	Hash les informations à caractères personnelles. Documentation et transparence.
Score biaisé par le modèle de pondération	Moyen	Moyen	Validation manuelle sur échantillon

3.4. Considérations réglementaires et éthiques

Le projet s'inscrit dans un cadre où la collecte et le traitement des données en ligne doivent respecter les réglementations internationales en matière de protection de la vie privée et éthique.

3.4.1. Protection des données personnelles

- Conformité réglementaire : assurer le respect des principales législations internationales de protection des données : RGPD (Union européenne), le CCPA (Californie, États-Unis) et LGPD (Brésil), selon l'origine des utilisateurs et des données collectées.
- Anonymisation : suppression ou masquage systématique des informations susceptibles d'identifier directement ou indirectement un individu (nom, identifiant client, coordonnées, …) avant tout traitement analytique.

- Limitation des finalités : les données collectées sont utilisées uniquement à des fins d'analyse de qualité des avis produits et non à des fins commerciales ou de profilage des utilisateurs.

3.4.2. Transparence

- Documentation du modèle : les algorithmes utilisés sont entièrement documentés, incluant la description des variables utilisées, les choix méthodologiques et les justifications associées.
- Traçabilité : chaque transformation ou traitement appliqué aux données est traçable afin d'assurer une auditabilité complète du pipeline.

3.4.3. Éthique et équité algorithmique

- Neutralité des critères : les scores et pondérations sont fondés sur des critères objectifs (note, longueur du commentaire, analyse de sentiment) sans discrimination fondée sur la langue, le pays, la culture ou tout autre facteur non pertinent.
- Prévention des biais : un contrôle régulier est prévu pour identifier et corriger d'éventuels biais dans les données ou le modèle de pondération.
- Responsabilité humaine : les décisions ou interprétations issues des analyses demeurent assistées par l'humain, garantissant que les résultats ne soient pas utilisés sans validation ou contextualisation.

4. Matrice RACI

Fonctionnalités / Activités	Chef de projet	Data Engineer	Data Analyst	Data Scientist	Responsable Qualité / Sécurité
Sauvegarder/récupération des données	C	R/A	I	I	I
Analyse des données brutes	I	R	A	C	I
Traitement des données (nettoyage, transformation)	I	R/A	C	I	I
Implémentation des algorithmes de classification	I	C	I	R/A	I
Implémentation de l'algorithme de pondération	I	C	C	R/A	I
Création de la visualisation des résultats	I	C	R/A	C	I
Rédaction des tests (unitaires, validation des règles)	C	R/A	C	I	I
Rédaction de la documentation technique	A	R	C	C	I
Rédaction du rapport final	A	C	R	C	I
Mise en production	A	R	I	C	C
Définir les règles de qualité	C	R	A	I	C
Gérer la sécurité et les accès	I	C	I	I	R/A
Gérer la partie conformité et sécurité réglementaire	I	I	I	I	R/A

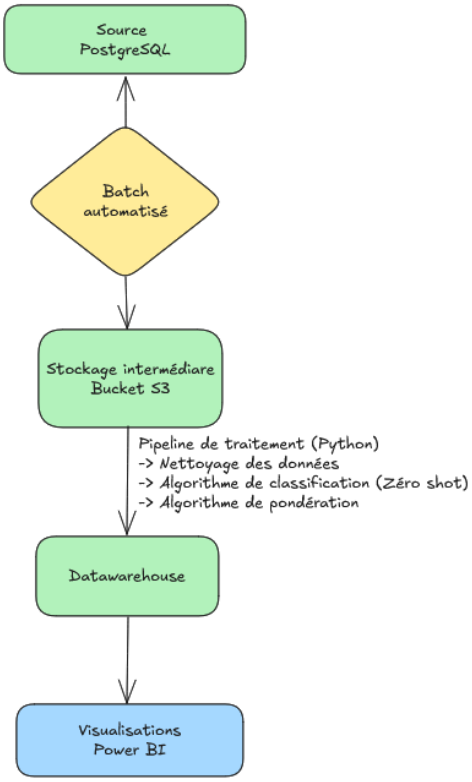
Légende :

- R (Responsible) : réalise la tâche.
- A (Accountable) : porte la responsabilité finale de la tâche et valide le résultat.
- C (Consulted) : Consulté pour avis ou expertise.
- I (Informed) : Informé du déroulement ou du résultat de la tâche.

5. Architecture technique et design de la solution

5.1. Diagramme d'architecture

L'architecture imaginée pour ce projet repose sur un flux de traitement batch quotidien permettant de collecter, transformer et analyser les avis produits Amazon avant de les restituer sous forme de rapports visuels.



5.2. Flux de données

5.2.1. Source PostgreSQL

- Les données brutes dont nous avons besoin pour notre projet sont stockées dans une base de données PostgreSQL. Cette base constitue la source principale d'alimentation de notre pipeline.

5.2.2. Bacth automatisé

- Chaque jour, un batch automatique extrait les nouvelles données depuis PostgreSQL et les sauvegarde dans un bucket S3.

5.2.3. Stockage intermédiare Bucket S3

- Ce stockage intermédiaire sert de zone d'atterissage avant les traitements.

5.2.4. Pipeline de traitement Python

- Les données sont ensuite récupérées depuis S3 pour être nettoyées et préparées.
- Un algorithme de classification est appliqué pour catégoriser les avis et donner un score de confiance.
- Un algorithme de pondération calcule ensuite un score de pertinence pour chaque avis.

5.2.5. Data warehouse

- Les données traitées sont chargées dans un data warehouse, cette couche centralise les données propres et prêtes à l'analyse.

5.2.6. Visualisation et reporting Power BI

- Les indicateurs clés et les scores calculés sont visualisés dans Power BI à travers des tableaux de bord automatisés.

5.3. Prés-requis d'intégration

Le système s'intègre avec la base de données (PostgreSQL), un stockage cloud (S3), un environnement de traitement Python, et un outil de BI (Power BI).
Aucune API externe n'est utilisée dans cette version, mais la structure reste compatible avec une intégration future (API Amazon ou API internes).

5.4. Mesures de sécurité

La sécurité est assurée par un contrôle d'accès par rôle, le chiffrement des données en transit et au repos, et la suppression des informations sensibles avant analyse.

- Authentification et autorisation : accès à PostgreSQL, S3 et Power BI restreint par identifiants sécurisés login/password et clés d'accès.
- Les utilisateurs ont des rôles de sécurités différenciés, lecture seule pour Power BI et écriture pour les développeurs.
- Les données sont chiffrées.
- Les données à caractères personnelles sont anonysimées avant traitement.
- La solution est adaptée aux différentes réglementations selon le territoire de provenance des données.

5.5. Accessibilité et facilité d'utilisation

Cette section concerne la partie visualisation Power BI. Les rapports sont conçus pour être :

- Clairs et intuitifs avec une navigation simple.
- Accessibles à différents profils (techniques ou pas).
- Compatible avec les bonnes pratiques d'accessibilités (respect de la charte et palettes de couleurs Amazon).

5.6. Maintenabilité et scalabilité

La solution est conçue pour être modulaire, versionnée et documentée. Elle peut évoluer progressivement vers une architecture cloud plus automisatée si les besoins augmentent.

- Versioning : Github.
- Documentation : toutes les étapes du pipeline sont documentées et le code inclut des commentaires clairs.
- Modularité : chaque étape de la solution peut être remplacée ou améliorée sans impacter tout le système existant.
- Scalabilité : l'architecture peut avoluer vers une infrastructure cloud complète au besoin.

6. Outils, langages et infrastructure

L'ensemble des outils et technologies a été sélectionné pour leur simplicité d'intégration, leur adéquation avec les besoins du projet et leur accessibilité dans un cadre académique. Cette stack technique permet de couvrir l'ensemble du cycle de vie des données : collecte, nettoyage, analyse, visualisation et documentation.

Catégorie	Technologie/outil	Justification
Environnement d'analyse	Notebook Jupyter	Environnement interactif idéal pour le prototypage, la visualisation et le partage des analyses de données.
Langage de programmation	Python	Langage polyvalent pour la manipulation de données, le NLP et l'implémentation d'algorithmes de classification et de pondération. Large écosystème de bibliothèques.
Base de données	PostgreSQL	Base relationnelle open source robuste, adaptée au stockage structuré et à la gestion des données nettoyées et transformées.
Stockage	Amazon S3	Stockage des fichiers extraits avant traitement. Permet de gérer des données brutes et d'assurer la séparation entre les zones "raw" et "clean".
Visualisation	Power BI	Outil de BI puissant et ergonomique pour la création de tableaux de bord dynamiques et la restitution des indicateurs qualité.
Contrôle de versionning	Git / GitHub	Suivi des versions du code, collaboration et gestion des différentes itérations du projet.
Batch automatisé	Apache Airflow	Il a été choisi pour sa capacité à gérer des workflows complexes et à orchestrer les tâches de traitement de données de manière modulaire. Sa compatibilité avec Python, PostgreSQL et AWS S3 en fait un choix cohérent avec le reste de la stack technique.
Documentation	Word/PDF	Rédaction du rapport technique et de la documentation projet de manière claire et structurée.

6.1. Choix des algorithms

Dans le cadre de notre projet d'analyse des avis produits, plusieurs expérimentations ont été menées afin de sélectionner les modèles et les approches les plus adaptés à notre contexte.

6.1.1. Zero-Shot Learning

Nous avons choisi de partir une approche de zero-shot classification, permettant de classer les avis. Cette méthode présente l'avantage d'être flexible et multilingue, ce qui correspondait parfaitement à notre cas d'usage, puisque les avis Amazon analysés sont rédigés dans plusieurs langues.

model="MoritzLaurer/mDeBERTa-v3-base-xnli-multilingual-nli-2mil7"

Ce modèle, entraîné sur des données multilingues, s'est révélé particulièrement performant pour notre tâche, car il évite l'ajout d'une étape de traduction automatique tout en maintenant une bonne cohérence sémantique entre les langues.

Nous avons également testé différentes combinaisons de labels candidats avant de retenir ceux offrant les meilleurs résultats en termes de précision et de pertinence :

candidate_labels = ["positive feedback", "negative feedback", "delivery problem", "customer service issue"]

6.1.2. Calcul du score de pertinence (Pondération des critères)

Afin d'évaluer la pertinence de chaque avis, nous avons mis en place un algorithme de pondération combinant plusieurs critères qualitatifs et quantitatifs. Après plusieurs itérations et ajustements, nous avons défini les poids suivants, en fonction de l'importance de chaque variable dans la détermination de la valeur d'un avis :

- 25% pour la taille de la description.
- 20% pour la présence d'une image avec l'avis.
- 15% pour un utilisateur qui a passé une commande.
- 25% pour la valeur des mots clés trouvés dans la description.

Cette pondération met un accent particulier sur la richesse du texte et la présence de mots-clés pertinents, indicateurs forts de la valeur analytique d'un avis. Les autres critères, tels que la présence d'une image ou la note extrême, contribuent également à affiner le score global.

Cette approche empirique, fondée sur des tests successifs, nous a permis d'obtenir un équilibre satisfaisant entre qualité sémantique, représentativité et pertinence opérationnelle des résultats.

6.2. Librairies Python

Librairie	Description
psycopg2	Permet la connexion et l'interaction avec la base de données PostgreSQL (exécution de requêtes SQL, insertion, lecture de données, etc.).
pandas	Librairie essentielle pour la manipulation et l'analyse de données. Elle offre des structures de données comme les DataFrames, très pratiques pour nettoyer, transformer et explorer les jeux de données.
numpy	Fournit des outils performants pour le calcul numérique et les opérations sur les tableaux multidimensionnels. Utilisée en complément de pandas pour les calculs statistiques.
matplotlib.pyplot	Module de visualisation permettant de créer des graphiques (histogrammes, courbes, barres, etc.) pour représenter les données de manière claire et visuelle.
seaborn	Librairie basée sur matplotlib, facilitant la création de visualisations statistiques avancées avec un design plus esthétique et intuitif.
datetime	Module standard Python pour la gestion des dates et heures, utile notamment pour analyser les tendances temporelles des avis ou calculer des durées.
transformers	Librairie développée par Hugging Face, utilisée pour exploiter des modèles de langage pré-entraînés (BERT, RoBERTa, etc.). Dans notre projet, elle a servi à effectuer la classification zero-shot via le pipeline NLP.

7. Annexes

7.1. Glossaire

- Batch : Traitement automatique de données exécuté à intervalles réguliers (ex. quotidien).
- ETL : Processus d'Extraction, Transformation et Chargement des données.
- Pipeline : Ensemble des étapes successives de traitement de la donnée.
- Data Warehouse : Base de données optimisée pour l'analyse et la visualisation de grandes quantités de données.
- Pondération : Calcul d'un score de confiance basé sur plusieurs critères (note, sentiment, longueur du texte…).

7.2. Références

- Documentation officielle PostgreSQL : https://www.postgresql.org/docs/
- Documentation AWS S3 : https://docs.aws.amazon.com/s3/
- Documentation Apache Airflow : https://learn.microsoft.com/power-bi/
- Cours / ressources internes Jedha sur la data pipeline et la data quality.

7.3. Documents associés

- Rapport d'analyse stratègiques.
- Document de veille reglementaire et technologique.
- Rapport sur les opportunités en matière de données.
- Prototype fonctionnel pour la catègorisation des avis produits.
- Documents sur les exigences du projet.

7.4. Informations de contact

- Apprentie Data Engineer (école Jedha) : Ismael SYLLA