Rendu pour le TP OpenFoodFacts

Introduction:

Ce rapport présente une analyse des données nutritionnelles à l'aide de PySpark, un framework de traitement des données distribuées. L'objectif de cette analyse est de nettoyer et de traiter les données sur les produits alimentaires, les régimes alimentaires et les utilisateurs pour générer un menu hebdomadaire équilibré.

1) Récupération des données

Pour récupérer les données OpenFoodFacts, nous avons exploité la source de données accessible à l'adresse https://fr.openfoodfacts.org/data. À partir de cette source, nous avons téléchargé le fichier CSV contenant les informations sur les produits alimentaires. Ensuite, en utilisant PySpark, nous avons chargé et traité ces données dans notre environnement de développement Eclipse. De plus, nous avons affiché la structure racine du fichier ainsi que le nombre de lignes et de colonnes pour évaluer le volume de donnée.

1.2) Informations sur les données

Le fichier comporte 3085838 lignes et 206 colonnes. Le type de données varie entre type textuelle ou numérique.
On a défini aussi une fonction column_info qui permet de donner les caractéristique(max, min, count,...) d'une colonne spécifique. On a pris la colonne code comme exemple pour la tester.

2) Nettoyage des données

2.1) Sélection des Colonnes Utiles

Initialement, nous avons entrepris de nettoyer les données en éliminant toutes les lignes comportant des valeurs NULL. Cependant, nous avons rapidement constaté que l'intégralité du fichier contenait des valeurs NULL, aboutissant ainsi à un tableau vide. Nous avons donc dû explorer une autre approche. Par conséquent, nous avons opté pour la sélection des colonnes pertinentes. Nous avons identifié et conservé uniquement les colonnes jugées essentielles pour répondre à nos besoins. Celles-ci incluent le code produit, le nom du produit, le pays d'origine, le grade nutriscore et les informations nutritionnelles telles que l'énergie, les lipides, les glucides, les protéines et le sel.

2.2) Suppression des Données Manquantes ou Incomplètes

Nous avons employé des transformations Spark pour filtrer les lignes du DataFrame relatives aux produits alimentaires pour lesquels des informations essentielles étaient absentes ou incomplètes. Cela inclut les produits sans nom ou sans données nutritionnelles significatives. Nous avons choisi de conserver la valeur "unknown" pour le nutriscore, car nous avons jugé que cette information n'était pas indispensable mais pouvait être bénéfique si elle était disponible.

2.3) Filtrage des Valeurs Aberrantes

En exploitant les capacités de traitement distribué d'Apache Spark, nous avons identifié et éliminé les valeurs aberrantes dans les données, en particulier pour les composantes nutritionnelles telles

que les lipides, les glucides, les protéines, etc. Les valeurs extrêmes ou peu plausibles ont été filtrées pour garantir la qualité des données. Après filtrations on retrouve avec 802700 lignes et 9 colonnes.

3) Création des fichiers CSV utilisateurs et régimes

Nous avons créé deux fichiers :

a. regimes_nutritionnels

ce fichier représente une table de référence pour différents régimes alimentaires avec des seuils recommandés pour certaines valeurs nutritionnelles. Chaque ligne correspond à un régime spécifique avec des valeurs maximales recommandées pour les nutriments tels que les glucides, les protéines, les lipides et les calories.

Ce fichier compose les colonnes suivantes:

- régime: Cette colonne représente le nom du régime alimentaire. Chaque ligne correspond à un régime spécifique, par exemple, "FODMAP", "Mediterraneen", "Paleo", etc.

- max_glucides_g: Cette colonne indique la quantité maximale recommandée de glucides (en grammes) par jour pour une personne suivant ce régime alimentaire.

- max_proteines_g: Cette colonne indique la quantité maximale recommandée de protéines (en grammes) par jour pour une personne suivant ce régime alimentaire.

- max_lipides_g: Cette colonne indique la quantité maximale recommandée de lipides (en grammes) par jour pour une personne suivant ce régime alimentaire.

- max_calories: Cette colonne indique la quantité maximale recommandée de calories par jour pour une personne suivant ce régime alimentaire.

Ce fichier est utilisé dans notre application Spark pour définir les critères de seuils lors de la génération d'un menu alimentaire personnalisé en fonction du régime alimentaire d'un utilisateur. Les colonnes de ce fichier sont sélectionnées et utilisées lors de la jointure avec les données des utilisateurs pour créer un menu personnalisé respectant les recommandations nutritionnelles spécifiques à chaque régime.

b. utilisateurs_regimes

Le fichier représente une table d'utilisateurs avec des informations telles que l'identifiant utilisateur, l'âge, le sexe, le poids et le régime alimentaire suivi par chaque utilisateur.

Les colonnes sont les suivantes :

- utilisateur_id: Cette colonne représente un identifiant unique pour chaque utilisateur. Chaque ligne a un identifiant utilisateur différent.
- age: Cette colonne indique l'âge de l'utilisateur.
- sexe: Cette colonne représente le sexe de l'utilisateur, généralement "M" pour masculin et "F" pour féminin.
- poids: Cette colonne indique le poids de l'utilisateur.
- regime_alimentaire: Cette colonne indique le régime alimentaire suivi par l'utilisateur. Les valeurs peuvent être, par exemple, "Vegetarien", "Cetogene", "Vegan", etc.

Ces informations sur les utilisateurs sont utilisées dans notre application Spark pour personnaliser les menus alimentaires en fonction des préférences et des besoins nutritionnels de chaque utilisateur. Lorsque nous générons le menu hebdomadaire, nous pouvons utiliser ces informations pour appliquer des filtres spécifiques basés sur le régime alimentaire de chaque utilisateur, en veillant à respecter les seuils recommandés pour les nutriments spécifiés dans le fichier de régimes alimentaires.

4) Intégration des Informations Utilisateurs avec les Seuils des Régimes Alimentaires

Dans cette section, nous intégrons les informations des utilisateurs avec les seuils des régimes alimentaires. Nous effectuons une jointure entre les données des utilisateurs et celles des régimes alimentaires en utilisant la colonne "regime_alimentaire" comme clé de jointure. Les colonnes sélectionnées incluent l'identifiant de l'utilisateur, le régime alimentaire, ainsi que les seuils maximums de glucides, de protéines, de lipides et de calories.

Génération Aléatoire du Menu Hebdomadaire

Dans cette étape, nous générons aléatoirement un menu équilibré pour chaque jour de la semaine en fonction des produits alimentaires disponibles dans notre ensemble de données OpenFoodFacts. Nous filtrons les données pour nous assurer que les valeurs nutritionnelles essentielles telles que l'énergie, les lipides, les glucides, les protéines et le sel ne sont pas nulles.

Pour chaque jour de la semaine, nous sélectionnons aléatoirement 7 produits alimentaires à partir de notre ensemble de données OpenFoodFacts, en garantissant qu'ils sont représentatifs des besoins nutritionnels des utilisateurs. Nous ajoutons ensuite ces sélections à notre menu hebdomadaire.

Stockage du Menu Hebdomadaire

Une fois le menu hebdomadaire généré, nous le stockons dans un fichier CSV à l'emplacement spécifié sur le système de fichiers local. De plus, nous créons une table dans le Data Warehouse (DWH) pour stocker ce menu. Une autre table est créée pour établir un lien entre le menu hebdomadaire et l'utilisateur final.

Ce rapport résume le processus de génération du menu hebdomadaire à partir des données OpenFoodFacts, des informations sur les utilisateurs et des régimes alimentaires, ainsi que les étapes suivies pour stocker les résultats dans un entrepôt de données.

Conclusion:

Ce rapport présente une analyse complète des données nutritionnelles à l'aide de PySpark. Le processus comprend le chargement, le nettoyage, l'intégration et la génération d'un menu hebdomadaire personnalisé pour chaque utilisateur. Les résultats obtenus peuvent être utilisés pour recommander des menus alimentaires équilibrés aux utilisateurs en fonction de leurs besoins nutritionnels.