26/03/2024

Franklin essono & Nadir Zourdani

ESIEA

Rapport d’analyse

Projet NoSQL

Contents

[1. Introduction 1](#_Toc162288273)

[2. Logique de résolution des questions 2](#_Toc162288274)

[ MongoDB Manager (mongodb\_manager.py) 2](#_Toc162288275)

[ Neo4j Manager (neo4j\_manager.py) 2](#_Toc162288276)

[3. Analyse des résultats obtenus 3](#_Toc162288277)

[ Analyse des données MongoDB 3](#_Toc162288278)

[ Analyse des données Neo4j 5](#_Toc162288279)

[ Intégration des données et influenceurs principaux 5](#_Toc162288280)

[4. Requêtes utilisées 5](#_Toc162288281)

[ Script d'initialisation de la base de données Neo4j 5](#_Toc162288282)

[ Initialisation de la base de données MongoDB 6](#_Toc162288283)

[5. Difficultés rencontrées et solutions adoptées 7](#_Toc162288284)

[6. Conclusion 8](#_Toc162288285)

# Introduction

Avec l’avancement actuel du big data, les bases de données NoSQL sont devenues des outils inestimables pour gérer des ensembles de données vastes et diversifiés. Ce projet explore deux systèmes de gestion de bases de données NoSQL significatifs : MongoDB et Neo4j. MongoDB, une base de données orientée document, offre une grande flexibilité dans le stockage et la récupération de données non structurées, tandis que Neo4j, une base de données orientée graphe, excelle dans la modélisation et l'analyse des relations complexes entre les données.

L'objectif principal de ce projet était de développer une application en Python capable d'interagir efficacement avec ces bases de données NoSQL hébergées dans le cloud. Cette application vise à exécuter une série de tâches spécifiques : se connecter aux bases de données, effectuer diverses interrogations et manipulations de données, et enfin, analyser et visualiser les résultats obtenus pour en tirer des insights pertinents.

Ce rapport présente le processus de développement de l'application, la logique employée pour répondre aux questions posées par les bases de données, et une analyse des résultats obtenus. Enfin, il conclut sur les leçons apprises tout au long de ce projet et les perspectives. En abordant ces points, ce document vise à fournir une vue d'ensemble complète des défis et des accomplissements associés à la manipulation et à l'analyse de données NoSQL dans un environnement cloud.

# Logique de résolution des questions

## MongoDB Manager (mongodb\_manager.py)

Dans le cadre de notre projet, le fichier mongodb\_manager.py joue un rôle crucial dans l'interrogation et la manipulation de données au sein de la base de données MongoDB. Voici comment ce gestionnaire aborde les questions posées :

* **Gestion des entités :** Le gestionnaire établit une connexion avec la base de données MongoDB et utilise différentes méthodes pour comptabiliser les entités telles que les utilisateurs, les tweets et les hashtags. Ces comptages sont fondamentaux pour répondre aux questions concernant les statistiques de base de l'application.
* **Interrogation par hashtags :** Le gestionnaire effectue des requêtes spécifiques pour récupérer le nombre de tweets contenant des hashtags particuliers et le nombre d'utilisateurs uniques ayant utilisé un hashtag donné. Cette fonctionnalité répond aux questions relatives à l'engagement et à la popularité des sujets sur la plateforme.
* **Analyse des discussions :** À travers une série de méthodes, le gestionnaire identifie les tweets qui initient des discussions et évalue les conversations en calculant la plus longue discussion basée sur le nombre de réponses. Cela permet d'aborder des questions sur l'interaction et la portée des discussions entre utilisateurs.
* **Évaluation de popularité :** En triant et en limitant les requêtes, le gestionnaire détermine les tweets et les hashtags les plus populaires. Cette analyse est cruciale pour comprendre les tendances et les intérêts des utilisateurs.
* **Relations entre utilisateurs :** Le gestionnaire explore la dynamique sociale en identifiant les utilisateurs avec un nombre significatif de followers ou ceux qui suivent de nombreux autres utilisateurs, aidant à répondre aux questions sur l'influence et les réseaux sociaux.

## Neo4j Manager (neo4j\_manager.py)

Le fichier neo4j\_manager.py s'attaque aux aspects de la gestion des données orientées graphe avec Neo4j :

* **Connexion et requêtes Cypher :** Le gestionnaire établit une connexion sécurisée avec Neo4j et exécute des requêtes Cypher pour analyser les relations entre les utilisateurs. Cela comprend la récupération du nombre de followers et de suivis pour des utilisateurs spécifiques, comme 'Spinomade'.
* **Analyse des relations :** En employant des requêtes complexes, le gestionnaire dévoile les réseaux de relations, identifiant par exemple les followers mutuels, les utilisateurs suivis et les followers de 'Spinomade'. Cela répond à des questions sur les structures sociales et l'interconnectivité entre les utilisateurs.
* **Intégration MongoDB-Neo4j :** Le gestionnaire utilise l'intégration entre MongoDB et Neo4j pour enrichir les requêtes Neo4j avec des données utilisateur provenant de MongoDB, offrant ainsi une vue complète des interactions et des relations.

En combinant les fonctionnalités de **mongodb\_manager.py** et **neo4j\_manager.py**, notre application peut répondre de manière exhaustive aux questions posées par le projet. Ces gestionnaires facilitent non seulement la récupération et l'analyse des données mais aussi leur visualisation et leur compréhension dans le contexte des réseaux sociaux numériques et de l'engagement des utilisateurs.

# Analyse des résultats obtenus

L'analyse des données recueillies et générées par notre application offre un aperçu significatif du comportement et des interactions des utilisateurs sur la plateforme sociale analysée.

## Analyse des données MongoDB

A graph of blue rectangular bars

Description automatically generated

*Nombre d’utilisateurs, de tweets et de hashtags*

A graph with blue bars and black text

Description automatically generated with medium confidence

*Les 10 hashtags les plus populaires*

La première visualisation met en évidence le nombre d'utilisateurs, de tweets et de hashtags. Avec respectivement environ 1200, 1400 et 1500, les données indiquent une plateforme active où les hashtags légèrement surpassent les tweets et les utilisateurs en termes de quantité. Ce déséquilibre peut suggérer une forte culture de tagging, importante pour le ciblage et la découverte de contenu.

Les 10 hashtags les plus populaires, dominés de manière écrasante par '#socionoel' avec 1225 mentions, révèlent des tendances saisonnières ou des événements spécifiques captivant l'attention des utilisateurs. La chute drastique de mentions des hashtags suivants, avec '#lt' ne recueillant que 33 mentions, indique un intérêt concentré, probablement autour d'un événement unique ou d'une campagne spécifique, laissant peu de place pour des thèmes diversifiés dans la conversation générale.

La plus longue discussion initiée par le tweet mentionné, qui a généré six réponses, suggère que bien que la plateforme puisse ne pas favoriser les longues chaînes de conversation, il existe néanmoins des discussions engagées. Cela peut indiquer une dynamique de communication où des sujets particuliers entraînent une plus grande interaction. **(A CHANGER)**

## Analyse des données Neo4j

A graph with different colored bars

Description automatically generated

*Relations de Spinomade*

Les relations de Spinomade, illustrées par le dernier graphique, offrent un aperçu fascinant des interactions sociales. La distinction entre les followers (bleu), les followings (rouge) et les relations mutuelles (vert) permet de comprendre les dynamiques d'influence et d'engagement. Notamment, certains utilisateurs, bien que suiveurs, ne sont pas suivis en retour, ce qui indique une hiérarchie ou une disparité dans les relations d'influence.

## Intégration des données et influenceurs principaux

En fusionnant les données recueillies à partir de MongoDB et Neo4j, une corrélation significative émerge entre le volume de contenu produit, tel que les tweets et les hashtags, et la structure des réseaux sociaux, notamment les suivis et les followers. Cette relation souligne comment les interactions et les connexions entre utilisateurs sont influencées par le contenu qu'ils génèrent et partagent. En outre, l'analyse révèle que les principaux vecteurs d'influence, marqués par leur large base de followers, jouent un rôle crucial dans l'orientation du dialogue et des tendances au sein de la communauté. Leur capacité à mobiliser, à informer et à influencer est particulièrement visible lorsqu'on considère l'impact significatif de comptes de personnalités publiques et d'entités telles que 'Libération' et 'Manuel Valls'. Ces derniers ne se contentent pas de participer au discours ; ils le façonnent, mettant en évidence le rôle prépondérant que les institutions et les figures politiques jouent dans la sphère des médias sociaux.

# Requêtes utilisées

## Script d'initialisation de la base de données Neo4j

Le script d'initialisation joue un rôle crucial dans la préparation de notre environnement de base de données Neo4j, permettant de structurer efficacement les données pour des analyses ultérieures. Voici une explication détaillée du processus :

A blue screen with text on it

Description automatically generated

*Init\_neo4j.py*

Dans ce script, la connexion à la base de données Neo4j est établie via des identifiants sécurisés. Ensuite, deux séries d'importations de données sont réalisées :

* Utilisateurs et leurs relations de suivi (Follow) : Les données sont importées depuis un fichier CSV accessible en ligne. Chaque ligne représente une relation de suivi entre deux utilisateurs. Les entités User sont créées dans la base de données Neo4j, avec des relations FOLLOWS indiquant qui suit qui.
* Tweets, utilisateurs, et relations de retweet : De manière similaire, les informations concernant les tweets et les interactions entre utilisateurs sont importées. Les entités Tweet sont associées aux utilisateurs qui les ont postés, et les relations RETWEET\_OF connectent les tweets originaux à leurs retweets.

Ce processus d'initialisation garantit que la base de données est bien structurée et prête à être utilisée pour l'analyse. Chaque étape est conçue pour refléter fidèlement les dynamiques réelles des utilisateurs et de leurs interactions sur la plateforme sociale, permettant ainsi des analyses précises des réseaux sociaux et des flux de contenu.

## Initialisation de la base de données MongoDB

Pour la base de données MongoDB, l'organisation s'est appuyée sur la conversion de trois fichiers CSV représentant respectivement les tweets, les hashtags associés et les utilisateurs, en collections distinctes. Cette structuration a facilité des requêtes complexes pour extraire les tendances, les discussions et les habitudes des utilisateurs, en utilisant les méthodes d'aggrégation et de filtrage propre à MongoDB.

A screenshot of a computer

Description automatically generated

*Collections MongoDB*

# Difficultés rencontrées et solutions adoptées

La manipulation des grandes quantités de données a présenté plusieurs défis, notamment en termes de performance et de cohérence des données. Des problèmes tels que la redondance des données et les requêtes longues ont nécessité l'optimisation des indices et l'amélioration des scripts d'interrogation pour une récupération plus efficace des données.

Une autre difficulté a été l'intégration des données entre Neo4j et MongoDB, nécessitant une synchronisation précise pour maintenir l'intégrité et la pertinence des analyses. Des solutions telles que l'emploi de références croisées et de conventions de nommage cohérentes ont permis de lier les données des deux systèmes de manière logique et efficace.

En outre, la complexité des requêtes Cypher et la nécessité d'une compréhension approfondie de la structure de graphe ont été surmontées par une phase d'apprentissage intensif et la consultation de la documentation et des forums spécialisés. Cette approche a permis d'affiner les requêtes pour extraire des informations significatives tout en maintenant des temps de réponse acceptables.

# Conclusion

Ce projet illustre l'efficacité de l'utilisation combinée de MongoDB et Neo4j pour explorer des ensembles de données complexes, reflétant les interactions et les comportements des utilisateurs sur une plateforme sociale spécifique. L'analyse a mis en évidence des compétences essentielles dans la manipulation, l'interrogation et la visualisation de données non structurées, compétences de plus en plus recherchées dans l'ère des big data. La capacité à déchiffrer des réseaux d'interaction et à identifier des tendances de contenu offre des perspectives précieuses pour des applications stratégiques et marketing.

Le projet souligne également l'importance d'une approche agile et de la capacité à apprendre continuellement face aux technologies de données émergentes. Les insights tirés de cette étude démontrent la valeur de la science des données dans la compréhension des dynamiques numériques et soulignent le besoin actuel de professionnels qualifiés dans ce domaine. En fin de compte, cette expérience enrichit notre compréhension dans le domaine du numérique et affine nos compétences en préparation aux défis futurs dans le monde des données.