Cas n°1

Un retailer souhaite faire de la veille de tendances sur les réseaux sociaux afin d'identifier rapidement des tendances de produits ou de lifestyle. Pour cela, il aimerait commencer par récupérer des tweets en temps réel, et identifier les sujets émergents, les utilisateurs influents et accéder facilement à ces agrégations.
Vous proposerez une architecture Cloud permettant de mettre en place cette pipeline de données et aux Data Scientists de pouvoir appliquer des modèles de Machine Learning (NLP, Information Retrieval, Image processing, ...) sur le jeu de données et d'en effectuer le suivi de performances au cours du temps.

Enjeux métiers

La société a besoin d'une infrastructure data pour pouvoir réaliser des analyses en temps réel. Notamment, l'utilisation du *machine learning* lui permettra de mieux cibler ses clients et mais aussi de mieux gérer ses approvisionnements.

Les réseaux sociaux fourniront potentiellement des indicateurs pour identifier de nouveaux clients ou de nouveaux marchés. Il s'agira d'utiliser des données historiques pour faire des analyses prédictives.

D'autre part, la sécurité et la scalabilité de l'analyse et de l'ingestion des données sont importantes, il s'agit ainsi de configurer finement l'accès aux données et d'améliorer les performances des modèles de *machine learning* sous-jacents.

Les *data analysts* et les *data scientists* veulent faire des expérimentations rapidement pour innover pour le client, et suivre en même temps des modèles déployés en production.

Des besoins implicites et liés à la problématique d'analyse sur les réseaux sociaux pourront être de disposer :

- de données d'inventaire sur une échelle de temps équivalent (quotidien),
- de données sur les commandes et les livraisons,
- d'un *datalake* centralisant tous ces agrégations pour analyse.

Besoins techniques

Plusieurs environnements sont nécessaires.

Environnement	Utilisation
Dev / Test	Expérimentations
Staging	Déploiement de nouvelles fonctionnalités
Production	Services au utilisateurs finaux

La sécurité, le transport et stockage des données doivent être pris en compte et sont gérées sur *Google Cloud Platform* par une politique de droits *IAM* adéquats et un réseau performant entre les *clusters* des projets et les environnement de stockage;

Il s'agit de construire des applications de *machine learning* basée sur du big data. La société a besoin de résultats rapides pour un coût raisonnable.

De manière classique, au vu du caractère déjà établi de la société, il sera nécessaire de voir avec lui son architecture SI et de voir comment migrer son environnement existant.

Localisation et distribution Datacenter ?
Bases de données SQL Server, PostgreSQL etc… ?
Serveurs applicatifs VMs, Tomcat, Nginx etc … ?

Systèmes de stockage iSCSI pour les VMs ? SAN sur SQL Server ? NAS pour les logs ?
Systèmes de traitement de données Hadoop / Spark ?
Infrastructure Monitoring, Bastion, Serveur d'intégration continue ?

La réflexion de la refonte sous *Google Cloud Platform*, amène donc *a priori* naturellement à l'utilisation de *Cloud Storage*, *Dataflow, BigQuery, Data Studio*, *AI Platform* et d'*AutoML* pour la

constitution d'un *datalake*, d'*ETL* capables d'extraire les données de réseaux sociaux en batch et temps réel, de visualiser ces données et de développer et livrer des modèles de machine learning en production.

Points clés à surveiller

Des tâches *ETL* supplémentaires doivent-elles être implémentées pour faciliter les travaux des *data scientists* et *analysts* ?
Les données sont-elles structurées ou relationnelles ? Qu'en est-il du volume de données ?
La société commence-t-elle de zéro ? Des jobs Hadoop existent vraisemblablement déjà dans leur datacenter et des investissements importants peuvent avoir été réalisés.

En fonction des réponses à ces questions l'architecture type pourra être modifiée.

Architecture générale

Ingestion de données
Pour les données structurées, *BigQuery* présente l'avantage de proposer une technologie de *datawarehouse* facile à administrer pour des données transactionnelles et relationnelles de systèmes logiques métier.
Pour les données non structurées, comme le texte ou les images issues des évènements de réseaux sociaux, *Cloud Storage* fournit un moyen de stockage distribué facile et rapide d'accès depuis les différentes composantes *big data* du système.

Dataflow permet d'effectuer des opérations de traitement de manière scalable sur une volumétrie importante et des données présentant une grande vélocité que ce soit en lot (*batch*) ou en continu (*streaming*).

Entraînement des modèles
Avec *AI Platform*, les *data scientists* disposent d'une plateforme pour développer des modèles *Tensorflow* permettant de tester différentes approches avec une interface de suivi des expérimentations et de déployer ensuite facilement différentes versions de modèles.

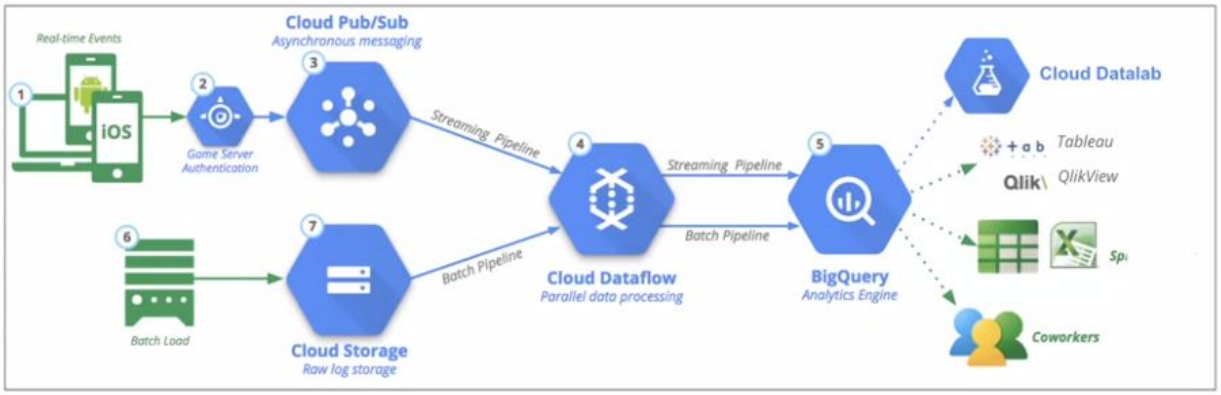
Les problématiques *NLP* et de reconnaissance d'images liées aux données non structurées du datalake pourront être traitée *AutoML* dans un premier temps pour tenter ensuite des approches plus spécifiques et détaillées avec du code *Tensorflow*.

Des dashboards *DataStudio* permettront aussi aux data analystes de réaliser des analyses intéractives.

Suivi opérationnel
Du côté de l'administration des systèmes, *Stackdriver* fournit des outils pour monitorer les performances applicatives et *Tensorboard* des métriques d'entraînement.

Le gain et le coût de migration vers le cloud doivent être évalués pour surmonter d'éventuelles limites business. La scalabilité de la chaîne logistique passe ainsi notamment par des systèmes complémentaires d'inventaire et de tracking temps-réel qui peuvent être traitées avec *Cloud IoT*.

En somme, la manière dont interopère chacun des éléments de ce système peut être résumée par ce schéma classique :



Exemples de code

On trouvera ci-dessous une liste d'éléments intéressants non exhaustive à intégrer dans l'architecture applicative.

Jobs *Dataflow* (Beam)
Exemple de pipeline entre une queue Pub/Sub et BigQuery
StreamingMinuteTrafficPipeline.java

L'API Twitter permet de récupérer ses évènements en mode batch ou temps réel
Twitter - Get batch historical tweets
Twitter - Filter realtime Tweets

Jobs *Dataproc* (Spark)
Analyses NLP depuis des fichiers textes situés sur *Cloud Storage*
Unstructured-ML.ipynb
Cloud Natural Language API : Analyzing Sentiment, Analyzing Entities

Jobs *Tensorflow* ou *AutoML*
Classification de texte à l'aide de réseaux CNN
text_classification.ipynb

Classification d'images à l'aide de réseaux CNN
Convolution model Application - v1.ipynb
Keras - Tutorial - Happy House v1.ipynb

Détection de bord à l'aide de réseaux YOLO
Tensorflow-planespotting

Bibliographie

Google - Preparing for the Google Cloud Professional Data Engineer Exam
Google - Labs and demos for Google Cloud Platform courses
Google - Tensorflow and deep learning without a PhD
Linux Academy - The Data Dossier
Coursera - Labs for Deep learning courses