**EXL-Data Engineering Bootcamp Capstone Project-1**

**By – Mohammad Tabish Siddiqui**

**Business objective:**

* To create end to end data pipeline.
* To perform exploratory data analysis.
* To build a machine learning model for factors affecting the employees leaving the organisation.

**Data used & description:**

* Total six datasets are used.
* All the datasets are in csv format.

1. Employees (employees.csv):

emp\_no – Employee Id – Integer – Not Null

emp\_titles\_id – designation id – Not Null

birth\_date – Date of Birth – Date Time – Not Null

first\_name – First Name – Character – Not Null

last\_name – Last Name – Character – Not Null sex – Gender – Character – Not Null

hire\_date – Employee Hire date –Date Time -Not Null

no\_of\_projects – Number of projects worked on – Integer – Not Null Last\_performance\_rating – Last year performance rating – Character – Not Null

left – Employee left the organization – Boolean – Not Null

Last\_date - Last date of employment (Exit Date) – Date Time

1. Titles (titles.csv):

title\_id – Unique id of type of employee (designation id) – Character – Not Null

title – Designation – Character – Not Null

1. Departments (departments.csv)

dept\_no - Unique id for each department – character – Not Null

dept\_name – Department Name – Character – Not Null

1. Salaries (salaries.csv):

emp\_no – Employee id – Integer – Not Null

Salary – Employee’s Salary – Integer – Not Null

1. Department Managers (dept\_manager.csv)

dept\_no - Unique id for each department – character – Not Null

emp\_no – Employee number (head of the department ) – Integer – Not Null

1. Department Employees (dept\_emp.csv)

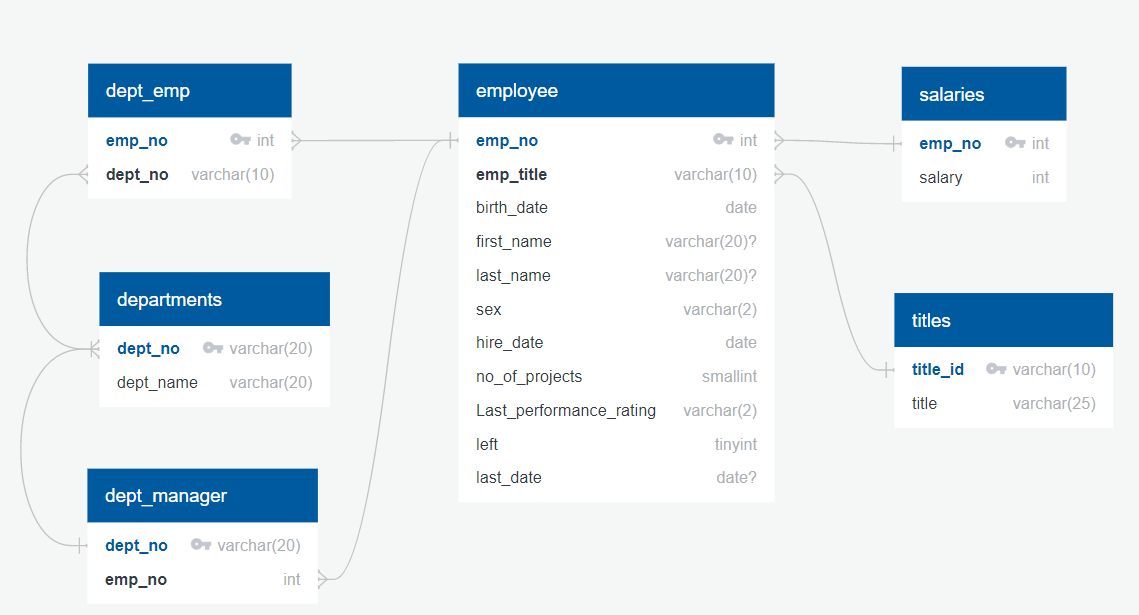
emp\_no – Employee id – Integer – Not Null

dept\_no - Unique id for each department – character – Not Null

**Technology stack used:**

* MySQL (to create database)
* Linux Commands
* Sqoop (Transfer data from MySQL Server to HDFS/Hive)
* HDFS (to store the data)
* Hive (to create database)
* Impala (to perform the EDA)
* SparkSQL (to perform the EDA)
* SparkML (to perform model building)

**ER Diagram:**

****

**Architecture of pipeline:**

* Step 1: Creation of MySQL database based on the data files.

File name: step\_1.sql

* Step 2: Importing the tables from MySQL(RDBMS) to Hive as avro files. Schema files copied from local machine to hdfs.

File name: step\_2.sh

* Step 3: Creating tables in hive to fetch the data from the imported files.

File name: step\_3.sql

* Step 4: Using Impala to perform exploratory data analysis.

File name: step\_4.sql

* Step 5: Importing tables from Hive to Spark. Then performing exploratory data analysis from these tables.

File name: step\_5.py

* Step 6: Building a machine learning model.

File name: step\_6.py

**How to use the files to run the pipeline:**

* Log in to cloudera
* Open web shell
* Run the commands in sequence:

1. mysql -u anabig11429 -pBigdata123 "anabig11429" < “step\_1.sql”
2. sh step\_2.sh
3. hive -f step\_3.sql
4. impala-shell -f step\_4.sql
5. spark-submit step\_5.py
6. spark-submit step\_6.py

**Outputs for different analysis:**

* A large majority of employees earn just about 40000 as salary.
* Over 90000 employees have had a tenure of more than 12 years.
* A large majority of the employees are in middle age group (36-48 years).
* The development department has had the most expenditure on salary till date more than 4 billion. The Human Resource department has had the least expenditure on salary less than 800million.
* 15105 employees have been given PIP rating till date.
* ML model:

Accuracy = 90%

Precision = 81%

f1 = 85%

**Challenges you faced:**

* Importing the tables from MySQL to HDFS was a challenge using sqoop import-all-tables command of sqoop. There was an error regarding splitting. As a solution I imported the tables individually using sqoop import command.
* Creating the tables in Hive was a challenge to fetch data from avro tables. I copied the avro schema files(.avsc) from local machine to hdfs and then pointed the hive table schema to that location in hdfs.
* Importing hive tables to spark was a challenge. I used thrift server to point the spark session object metastore location to hive metastore and retrieved the tables.
* Building the ML pipeline was a challenge regarding feature engineering.
* The pipeline was not able to fit the data in form of list. So multiple stages had to be included. Individual stages for every string indexer and one hot encoder respectively were used.