CA 675 Cloud Technologies Assignment 1

Conall Butler

Student Number:21269599

[conall.butler36@mail.dcu.ie](mailto:conall.butler36@mail.dcu.ie)

Git repository: <https://github.com/ConallButler/CA675-Assignment-1>

Task 1; The Dataset

The acquired dataset used in subsequent tasks consists for 5 CSV files of total 200,000 entries excluding headers. Schema provided by stack exchange1.

The dataset was acquired using the Stack Exchange Data Explorer2. See queries and number of entries yielded are show below. Each row corresponds to a post on the Stack Exchange forum, with each cell containing some piece of metadata, the Title, or the Body of the post. The body of posts appear as HTML in the dataset.

select top 50000 \* from posts where posts.ViewCount >140000 ORDER BY posts.ViewCount;

select top 50000 \* from posts where posts.ViewCount<140001 and posts.ViewCount >80000 ORDER BY posts.ViewCount;

select top 50000 \* from posts where posts.ViewCount<80001 and posts.ViewCount >58000 ORDER BY posts.ViewCount;

select top 50000 \* from posts where posts.ViewCount<58001 and posts.ViewCount >45000 ORDER BY posts.ViewCount;

select top 18724 \* from posts where posts.ViewCount<45001 and posts.ViewCount >41000 ORDER BY posts.ViewCount DESC;

|  |  |  |
| --- | --- | --- |
| **Query** | **Range** | **Posts** |
| 1 | >14000 | 43628 |
| 2 | >80000, <140001 | 47772 |
| 3 | >58000, <80001 | 43732 |
| 4 | >45000, <58001 | 46144 |
| 5 | >41000, <45001 | 18724 |

ViewCount ranges in intervals approaching 50000 posts were determined using the method laid out in the assignment document Data Acquisition section. 18724 posts in descending order were retrieved in query 5 as this was the remaining number required to reach 200,000 posts after summing the first 4 queries.

Output CSVS; <https://github.com/ConallButler/CA675-Assignment-1/tree/main/CSVs>

Task 2&3

Pig was selected for initial ETL tasks as dealing with line-breaks contained in data can be difficult using Hive.

2.2.1, 2.2.2, and 2.2.3 were identified as tasks that could be easily completed using SQL queries; Hive was selected as such

--Load CSVs output by Stack Exchange Queries using schema adjusted for Pig datatypes3

Query1 = LOAD 'CA675-Assignment-1/csvs/query1.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',', 'YES\_MULTILINE', 'WINDOWS', 'SKIP\_INPUT\_HEADER') AS (Appendix 3)

--Filter by required fields to reduce resources required, clean line-breaks from Body to allow easy processing in Hive

Queries = UNION Query1, Query2, Query3, Query4, Query5;

Queries\_Filtered\_Cleaned1 = FOREACH Queries GENERATE Id, Score, OwnerUserId, Title, REPLACE(Body,'\n', '') AS Body;

--Clean HTML tags so as not to interfere with counting of terms in 2.2.3 and 2.3

Queries\_Filtered\_Cleaned2 = FOREACH Queries\_Filtered\_Cleaned1 GENERATE Id, Score, OwnerUserId, Title, REPLACE(Body,'<.\*?.>', '') AS Body;

--Store filtered, cleaned data for use in later tasks.

STORE Queries\_Filtered\_Cleaned2 INTO 'CA675-Assignment-1/csvs/Queries\_Filtered\_Cleaned' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',' ,'NO\_MULTILINE' ,'UNIX', 'SKIP\_OUTPUT\_HEADER');

--Hive was used to complete the remainder of Task 2&3, data loaded as below

CREATE DATABASE sequeries;

USE sequeries;

CREATE EXTERNAL TABLE Queries\_Filtered\_Cleaned

(Id int, Score int, OwnerUserId int, Title string, Body string)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

LOCATION 'CA675-Assignment-1/csvs/Queries\_Filtered\_Cleaned';

**Screenshots**

<https://github.com/ConallButler/CA675-Assignment-1/blob/main/Screenshots/2.%20PIG%20ETL%201.png>

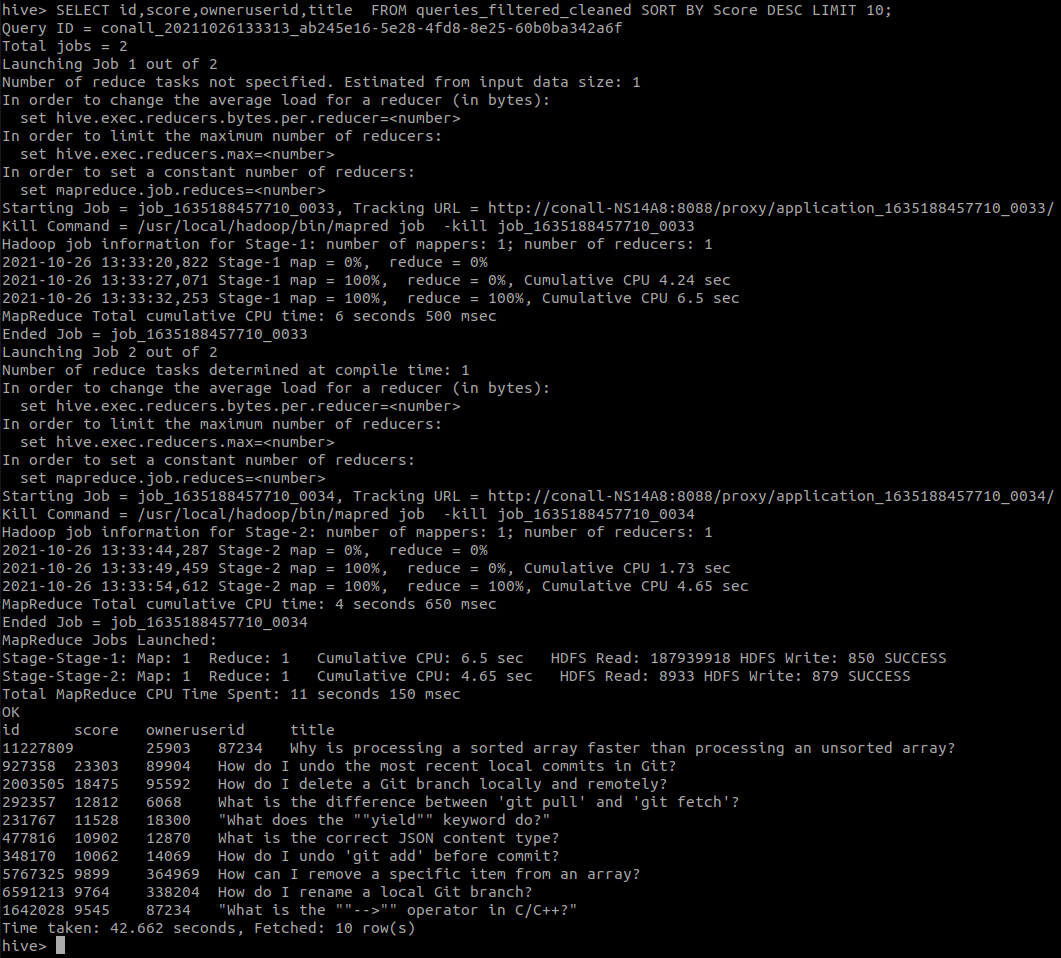
<https://github.com/ConallButler/CA675-Assignment-1/blob/main/Screenshots/3.%20Pig%20ETL%202.png>

<https://github.com/ConallButler/CA675-Assignment-1/blob/main/Screenshots/3.1%20PIG%20ETL%20terminal%20info>

Task 2.2.1

--Display Id, Score, Owneruserid and Title for top 10 posts, by score.

SELECT id,score,owneruserid,title FROM queries\_filtered\_cleaned SORT BY Score DESC LIMIT 10;



Screenshot;

<https://github.com/ConallButler/CA675-Assignment-1/blob/main/Screenshots/5.%20hive%202.2.1.png>

Task 2.2.2

--Sum score for each distinct user, display top 10 by summed score. Null OwnerUserId values ignored, these correspond to blanks in the source data.

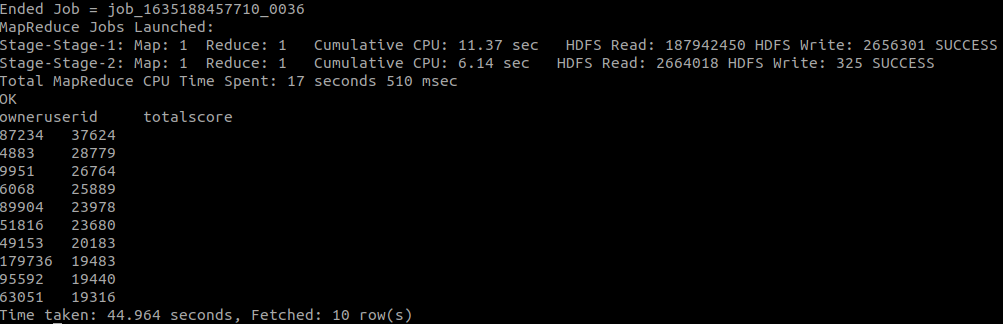
SELECT owneruserId, SUM(score) AS totalScore

FROM queries\_filtered\_cleaned

WHERE owneruserId IS NOT NULL

GROUP BY owneruserId

ORDER BY totalScore DESC LIMIT 10;



Screenshots

<https://github.com/ConallButler/CA675-Assignment-1/blob/main/Screenshots/6.%20hive%202.2.2(1).png>

<https://github.com/ConallButler/CA675-Assignment-1/blob/main/Screenshots/7.%20Hive%202.2.2(2).png>

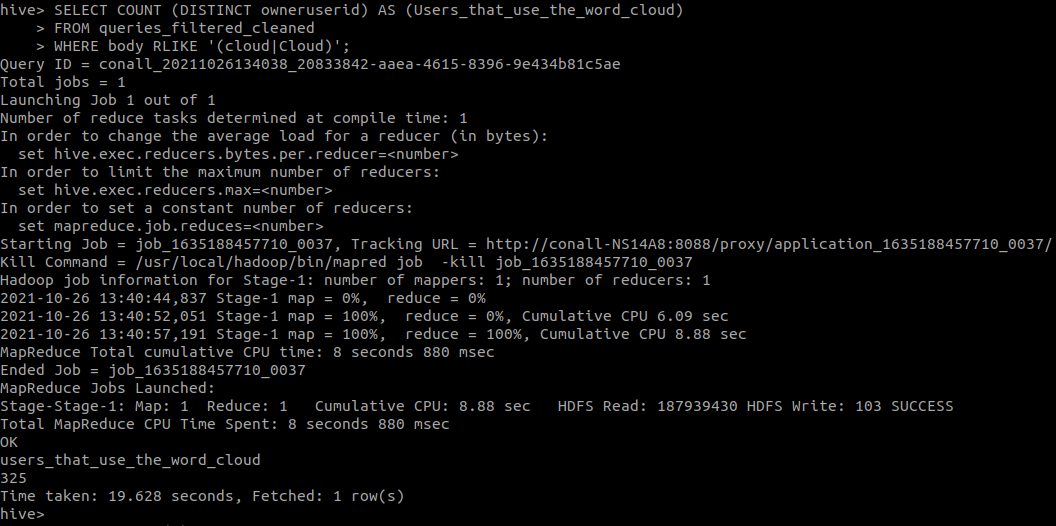
Task 2.2.3

--Count distinct users with “Cloud” or “cloud” in the body of one of their posts.

SELECT COUNT (DISTINCT owneruserid) AS (Users\_that\_use\_the\_word\_cloud)

FROM queries\_filtered\_cleaned

WHERE body RLIKE '(cloud|Cloud)';



Screenshot

https://github.com/ConallButler/CA675-Assignment-1/blob/main/Screenshots/8.%20Hive%202.2.3.png

Hive Outputs Spreadsheet format

<https://github.com/ConallButler/CA675-Assignment-1/blob/main/Outputs/Hive%20Outputs.ods>

Hive Queries Source Code

<https://github.com/ConallButler/CA675-Assignment-1/blob/main/Source%20Code/4.%20Hive%20Queries>

Task 4

Apache Pig was selected for task 4 due to ease of development in Pig Latin, and lack of familiarity with java, required for Hadoop/MadReduce.

Pg33 was referenced for the outline of TF-IDF Pig Latin script <https://courses.cs.ut.ee/MTAT.08.036/2017_fall/uploads/Main/L4_Pig_2017.pdf>

Final outputs were 10 tab delimited text files of Schema (Term, TF-IDF), one for each user. Top 10 were selected and combined in excel (top 10 can be done in Pig using LIMIT 10, all terms generated for completeness);

Pig Text Outputs

<https://github.com/ConallButler/CA675-Assignment-1/tree/main/Outputs/HDFS%20files/CA675-Assignment-1/csvs>

Output xlsx

<https://github.com/ConallButler/CA675-Assignment-1/blob/main/Outputs/Top10Users%20TF-IDF.xlsx>

Source Code

<https://github.com/ConallButler/CA675-Assignment-1/blob/main/Source%20Code/5.%20Pig%20TF-IDF>

Screenshots

<https://github.com/ConallButler/CA675-Assignment-1/blob/main/Screenshots/9.%20TFIDF%201.png>

<https://github.com/ConallButler/CA675-Assignment-1/blob/main/Screenshots/10.%20TFIDF%202.png>

1. Stack Exchange Schema (Id:int, PostType:tinyint, AcceptedAnswerId:int, ParentId:int, CreationDate:datetime, DeletionDate:datetime, Score:int, ViewCount:int, Body: nvarchar (max), OwnerUserId:int, OwnerDisplayName:nvarchar (40), LastEditorUserId:int, LastEditorDisplayName:nvarchar (40), LastEditDate:datetime, LastActivityDate:datetime, Title:nvarchar (250), Tags:nvarchar (250), AnswerCount:int, CommentCount:int, FavoriteCount:int, ClosedDate:datetime, CommunityOwnedDate:datetime, ContentLicense:varchar (12))
2. Stack Exchange Query <https://data.stackexchange.com/stackoverflow/query/new>
3. Pig Schema (Id:int, PostTypeId:int, AcceptedAnswerId:int, ParentId:int, CreationDate:datetime, DeletionDate:datetime, Score:int, ViewCount:int, Body: chararray, OwnerUserId:int, OwnerDisplayName:chararray, LastEditorUserId:int, LastEditorDisplayName:chararray, LastEditDate:datetime, LastActivityDate:datetime, Title:chararray, Tags:chararray, AnswerCount:int, CommentCount:int, FavoriteCount:int, ClosedDate:datetime, CommunityOwnedDate:datetime, ContentLicense:chararray)
4. Screenshots in chronological order <https://github.com/ConallButler/CA675-Assignment-1/tree/main/Screenshots>
5. Source Code <https://github.com/ConallButler/CA675-Assignment-1/tree/main/Source%20Code>
6. HDFS File Content <https://github.com/ConallButler/CA675-Assignment-1/tree/main/Outputs/HDFS%20files/CA675-Assignment-1/csv>