Rapport projet
Big Data

Analyse des données YouTube

2022/2023
IID 2

Réalisé par :
Chaimaa KHALIL & Siham HAFSI

Encadré par :
Mme Nassima SOUSSI

Sommaire

1. Introduction.

2. Objectifs.

3. Data YouTube.

4. Code JAVA.

5. Exécution.

6. Conclusion.

1. Introduction :

Le terme de Big Data désigne de vastes ensembles de données collectées par les entreprises, pouvant être explorées et analysées afin d'en dégager des informations exploitables.

Dans tous les secteurs, les entreprises utilisent le Big Data engrangé dans leurs systèmes à différentes fins. Il peut s'agir d'améliorer les opérations, de proposer un meilleur service client, de créer des campagnes marketing personnalisées basées sur les préférences des consommateurs, ou tout simplement d'augmenter le chiffre d'affaires. Grâce au Big Data, les entreprises peuvent profiter d'un avantage compétitif face à leurs concurrents n'exploitant pas les données. Elles peuvent prendre des décisions plus rapides et plus précises, s'appuyant directement sur les informations. Par exemple, une entreprise peut analyser le Big Data pour découvrir de précieuses informations sur les besoins et les attentes de ses clients. Ces informations peuvent ensuite être exploitées pour créer de nouveaux produits ou des campagnes marketing ciblées afin d'accroître la fidélité client ou d'augmenter le taux de conversion. Une entreprise s'appuyant totalement sur les données pour aiguiller son évolution est qualifiée de « data-driven » (dirigée par les données).

2. Objectifs :

Kaggle fournit un ensemble de statistiques sur les tendances des vidéos sur YouTube. Ce projet analyse les statistiques et utilise un programme Hadoop MapReduce qui fournit la vidéo la plus aimée, la plus vue et la plus détestée dans chaque catégorie sur YouTube.

3. Data YouTube:

Les données se trouvent au lien suivant :
https://www.kaggle.com/datasnaek/youtube

c'est un fichier csv qui contient 11 colonnes et 7999 lignes.

	A	B	C	D	E	F	G	H	I	J	K
1	video_id	title	channel_title	category_id	tags	views	likes	dislikes	comment_total	thumbnail_link	date
2	XpVt6Z1Gjjo	1 YEAR OF VLOGGING	Logan Paul Vlogs	24	logan paul vlog|l	4394029	320053	5931	46245	https://i.ytimg.com/vi/Xp	13.09
3	K4wEI5zhHB0	iPhone X — Introducing	Apple	28	Apple|iPhone 10	7860119	185853	26679	0	https://i.ytimg.com/vi/K4	13.09
4	cLdxuaxaQwc	My Response	PewDiePie	22	[none]	5845909	576597	39774	170708	https://i.ytimg.com/vi/cL	13.09
5	WYYvHb03Eog	Apple iPhone X first look	The Verge	28	apple iphone x h	2642103	24975	4542	12829	https://i.ytimg.com/vi/WY	13.09
6	sjlHnJvXdQs	iPhone X (parody)	jacksfilms	23	jacksfilms|parod	1168130	96666	568	6666	https://i.ytimg.com/vi/sjlH	13.09
7	cMKX2tE5Luk	The Disaster Artist | Offi	A24	1	a24|a24 films|a2	1311445	34507	544	3040	https://i.ytimg.com/vi/cM	13.09
8	8wNr-NQImFg	The Check In: HUD, Ber	Late Night with Seth M	23	Late night|Seth M	666169	9985	297	1071	https://i.ytimg.com/vi/8w	13.09
9	_HTXMhKWqnA	iPhone X Impressions &	Marques Brownlee	28	iPhone X|iphone	1728614	74062	2180	15297	https://i.ytimg.com/vi/_H	13.09
10	_ANP3HR1jsM	ATTACKED BY A POLIC	RomanAtwoodVlogs	22	Roman Atwood|l	1338533	69687	678	5643	https://i.ytimg.com/vi/_A	13.09
11	zgLtEob6X-Q	Honest Trailers - The Mu	Screen Junkies	1	screenjunkies|sc	1056891	29943	878	4046	https://i.ytimg.com/vi/zgL	13.09
12	Ayb_2qbZHm4	Honest College Tour	CollegeHumor	23	Collegehumor|C	859289	34485	726	1914	https://i.ytimg.com/vi/Ayb	13.09
13	CsdzflTXBVQ	Best Floyd Mayweather	Awkward Puppets	23	best floyd mayw	452477	28050	405	2745	https://i.ytimg.com/vi/Cs	13.09
14	l864IBj7cgw	Jennifer Lawrence Chall	The Tonight Show Sta	23	The Tonight Sho	258781	8085	303	726	https://i.ytimg.com/vi/l86	13.09
15	4MkC65emkG4	Hand In Hand A Benefit	MTV	24	mtv|video|online	274358	9215	477	838	https://i.ytimg.com/vi/4M	13.09
16	vu_9muoxT50	Colin Cloud: Mind Read	America's Got Talent	24	America's Got Ta	473691	14740	415	1696	https://i.ytimg.com/vi/vu_	13.09
17	1L7JFN7tQLs	iPhone X Hands on - Ev	Jonathan Morrison	28	Apple|iPhone X|i	514972	18936	641	3817	https://i.ytimg.com/vi/1L7	13.09
18	ZQK1F0wz6z4	What Do You Want to Ea	Wong Fu Productions	24	panda|what shou	282858	14870	300	1398	https://i.ytimg.com/vi/ZQ	13.09
19	T_PuZBdT2iM	getting into a conversati	ProZD	1	skit|korean|langu	1582683	65749	1531	3598	https://i.ytimg.com/vi/T_I	13.09
20	w8fAeIlnPns	Juicy Chicken Breast - Y	You Suck At Cooking	26	how to|cooking|r	479951	23945	640	1941	https://i.ytimg.com/vi/w8	13.09
21	UCrBICYM0yM	Downsizing (2017) - Offi	Paramount Pictures	1	downsizing|previ	2693468	7941	302	1432	https://i.ytimg.com/vi/UC	13.09
22	-lfnaxi2LQg	Fergie - You Already Kn	FergieVEVO	10	Fergie|You|Alrea	815608	66420	3578	5550	https://i.ytimg.com/vi/-lfn	13.09
23	B7YaMkCl3XA	Hurricane Irma death tol	Al Jazeera English	25	55730511420001	382525	1521	270	1168	https://i.ytimg.com/vi/B7	13.09
24	5ywKal6-anc	Gigi Hadid Loses High H	TMZ	24	TMZ2016FS112	703750	2921	2196	1042	https://i.ytimg.com/vi/5yv	13.09
25	4Yue-q9Jdbk	SUPERFRUIT REACTS	SUPERFRUIT	24	superfruit|super	255967	21817	293	2017	https://i.ytimg.com/vi/4Y	13.09
26	JhA1Wi9mrns	Kid Rock - Tennessee M	Kid Rock	10	kid rock|greatest	96872	3498	482	439	https://i.ytimg.com/vi/Jh	13.09
27											

4.Code JAVA:

1- Le programme Mapper supprime tous les enregistrements erronés de cet ensemble d'enregistrements.

il envoie un ensemble des clés, valeurs.
Clé : category_id
valeur : nombre des vues + le nombre des likes + le nombre des dislikes + lien + video_id .

MRdriver.java | MRmapper.java | MRreducer.java

```java
package youtube;

import org.apache.hadoop.mapreduce.Mapper;
import org.apache.log4j.Logger;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.io.LongWritable;
import java.io.IOException;
import java.util.ArrayList;
import java.util.HashMap;

public class MRmapper extends Mapper <LongWritable,Text,LongWritable,Text> {
    static String IFS=",";
    static String OFS=",";
    static int NF=11;
    int bad_record_counter = 0;
    public void map(LongWritable key, Text value, Context context)
                    throws IOException, InterruptedException {
        /** USvideos.csv
        video_id
        title
        channel_title
        category_id
        tags
        views
        likes
        dislikes
        comment_total
        thumbnail_link
        date
        */


        int category_id = 0;
        long num_of_views = 0;
        long num_of_likes = 0;
        long num_of_dislikes = 0;
        String thumbnail = null;
        String video_id = null;
        int bad_record_count = 0;

        if(key.get() == 0 && value.toString().contains("video_id")){
            return;
        }
        else{
            String [] data = value.toString().split(",");
            if(data.length!=11){
                bad_record_count++;
                return;
```

```
            }
            else{
                video_id = data[0];
                category_id = Integer.parseInt(data[3]);
                num_of_views = Long.parseLong(data[5]);
                num_of_likes = Long.parseLong(data[6]);
                num_of_dislikes = Long.parseLong(data[7]);
                thumbnail = data[9];
            }
        }

        String output_string = num_of_views+"_ "+num_of_likes + "_ " + num_of_dislikes + "_ " + thumbnail + "_ " + video_id;
        LongWritable newKey = new LongWritable();
        newKey.set(Long.valueOf(category_id));
        context.write(newKey, new Text(output_string));


    }
}
```

2- Le programme Reducer trouve les vidéos les plus aimées, vues et détestées dans chaque catégorie et les envoie au conducteur avec une clé.

MRdriver.java | MRmapper.java | MRreducer.java

```
package youtube;

import org.apache.hadoop.mapreduce.Reducer;
import org.apache.log4j.Logger;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import java.io.IOException;

public class MRreducer  extends Reducer <LongWritable,Text,Text,Text> {
    public static String IFS=",";
    public static String OFS=",";

    public void reduce(LongWritable key, Iterable<Text> values, Context context)
            throws IOException, InterruptedException {
        final Logger logger = Logger.getLogger("Mylog");
        String temp_string = null;
        long temp_num_views = 0;
        long temp_num_likes = 0;
        long temp_num_dislikes = 0;
        String temp_thumbnail_views = null;
        String temp_thumbnail_likes = null;
        String temp_thumbnail_dislikes = null;
        String video_id_views = null;
        String video_id_likes = null;
        String video_id_dislikes = null;

        for(Text value:values){
            temp_string = value.toString();
            String [] stringArray = temp_string.split("_ ");
            //temp_num_views = Long.parseLong(stringArray[0]);
            //temp_num_likes = Long.parseLong(stringArray[1]);
            //temp_num_dislikes = Long.parseLong(stringArray[2]);
            //temp_thumbnail = stringArray[3];
            if(temp_num_views<Long.parseLong(stringArray[0])){
                temp_num_views = Long.parseLong(stringArray[0]);
                temp_thumbnail_views = stringArray[3];
                video_id_views = stringArray[4];
            }
            if(temp_num_likes<Long.parseLong(stringArray[1])){
                temp_num_likes = Long.parseLong(stringArray[1]);
                temp_thumbnail_likes = stringArray[3];
                video_id_likes = stringArray[4];
            }
            if(temp_num_dislikes<Long.parseLong(stringArray[2])){
                temp_num_dislikes = Long.parseLong(stringArray[2]);
                temp_thumbnail_dislikes = stringArray[3];
                video_id_dislikes = stringArray[4];
```

```
48
49              }
50          }
51          Text keyText = new Text("\nmost views:" + video_id_views + OFS + temp_thumbnail_views + "\nmost likes:" + video_id_likes + OFS + temp_thumbnail_likes
52                  + "\nmost dislikes:" + video_id_dislikes + OFS + temp_thumbnail_dislikes);
53          context.write(new Text("category_id:"+key), keyText);
54          logger.info("Reducer completed");
55
56      }
57  }
```

3- Driver :

Le Driver est la classe principale qui crée et lance le job MapReduce. Elle contient la fonction main du programme et qui va permettre de :

a. Récupérer la configuration générale du cluster.
b. Créer un job, lui indiquer les classes concernées : mapper et reducer.
c. Définir les types de clés et de valeur de notre programme Hadoop.
d. Indiquer où sont les données d'entrée et de sortie dans HDFS.
e. Lancer l'exécution de la tâche.

MRdriver.java | MRmapper.java | MRreducer.java

```
1  package youtube;
2
3  import org.apache.hadoop.conf.Configured;
4  import org.apache.hadoop.conf.Configuration;
5  import org.apache.hadoop.io.FloatWritable;
6  import org.apache.hadoop.io.LongWritable;
7  import org.apache.hadoop.io.Text;
8  import org.apache.hadoop.mapreduce.Job;
9  import org.apache.hadoop.util.Tool;
10 import org.apache.hadoop.util.ToolRunner;
11 import org.apache.hadoop.fs.Path;
12 import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
13 import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
14 import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
15
16 public class MRdriver extends Configured implements Tool {
17     @SuppressWarnings("deprecation")
18     public int run(String[] args) throws Exception {
19         if (args.length != 2) {
20             System.err.printf("usage: %s [generic options] <inputfile> <outputdir>\n", getClass().getSimpleName());
21             System.exit(1);
22         }
23         Job job = new Job(getConf(), "my receipts");
24         job.setJarByClass(MRdriver.class);
25         job.setMapperClass(MRmapper.class);
26         job.setReducerClass(MRreducer.class);
27         job.setInputFormatClass(TextInputFormat.class);
28         job.setOutputKeyClass(LongWritable.class);
29         job.setOutputValueClass(FloatWritable.class);
30         job.setMapOutputValueClass(Text.class);
31         FileInputFormat.addInputPath(job, new Path(args[0]));
32         FileOutputFormat.setOutputPath(job, new Path(args[1]));
33         return job.waitForCompletion(true) ? 0 : 1;
34     }
35
36     public static void main(String[] args) throws Exception {
37         if(args.length != 2) {
38             System.err.println("usage: MRdriver <input-path> <output-path>");
39             System.exit(1);
40         }
41         Configuration conf = new Configuration();
42         System.exit(ToolRunner.run(conf, new MRdriver(), args));
43     }
44 }
45
```

5.Exécution :

On exécute la commande : **hadoop jar projet.jar /USvideos.csv big**
File input: /USvideos.csv
Dossier d'output : big

```
[cloudera@quickstart ~]$ hadoop jar projet.jar /USvideos.csv big
23/01/04 07:31:14 INFO client.RMProxy: Connecting to ResourceManager at quickstart.cloudera/192.168.94.128:8032
23/01/04 07:31:15 INFO input.FileInputFormat: Total input paths to process : 1
23/01/04 07:31:15 WARN hdfs.DFSClient: Caught exception
java.lang.InterruptedException
        at java.lang.Object.wait(Native Method)
        at java.lang.Thread.join(Thread.java:1281)
        at java.lang.Thread.join(Thread.java:1355)
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.closeResponder(DFSOutputStream.java:967)
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.endBlock(DFSOutputStream.java:705)
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.run(DFSOutputStream.java:894)
23/01/04 07:31:16 WARN hdfs.DFSClient: Caught exception
java.lang.InterruptedException
        at java.lang.Object.wait(Native Method)
        at java.lang.Thread.join(Thread.java:1281)
        at java.lang.Thread.join(Thread.java:1355)
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.closeResponder(DFSOutputStream.java:967)
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.endBlock(DFSOutputStream.java:705)
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.run(DFSOutputStream.java:894)
23/01/04 07:31:16 INFO mapreduce.JobSubmitter: number of splits:1
23/01/04 07:31:17 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1672842452454_0006
23/01/04 07:31:17 INFO impl.YarnClientImpl: Submitted application application_1672842452454_0006
23/01/04 07:31:17 INFO mapreduce.Job: The url to track the job: http://quickstart.cloudera:8088/proxy/application_1672
842452454_0006/
23/01/04 07:31:17 INFO mapreduce.Job: Running job: job_1672842452454_0006
23/01/04 07:31:35 INFO mapreduce.Job: Job job_1672842452454_0006 running in uber mode : false
23/01/04 07:31:35 INFO mapreduce.Job:  map 0% reduce 0%
23/01/04 07:31:53 INFO mapreduce.Job:  map 100% reduce 0%
23/01/04 07:32:09 INFO mapreduce.Job:  map 100% reduce 100%
23/01/04 07:32:10 INFO mapreduce.Job: Job job_1672842452454_0006 completed successfully
23/01/04 07:32:10 INFO mapreduce.Job: Counters: 49
        File System Counters
                FILE: Number of bytes read=240204
                FILE: Number of bytes written=775543
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=2982877
                HDFS: Number of bytes written=3662
                HDFS: Number of read operations=6
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
```

File output :

```
[cloudera@quickstart ~]$ hadoop fs -cat /user/cloudera/big/part-r-00000
category_id:1
most views:bu9e410C__I,https://i.ytimg.com/vi/bu9e410C__I/default.jpg
most likes:hkA2a4_tNOs,https://i.ytimg.com/vi/hkA2a4_tNOs/default.jpg
most dislikes:hkA2a4_tNOs,https://i.ytimg.com/vi/hkA2a4_tNOs/default.jpg
category_id:2
most views:Hg6L_7qLIEQ,https://i.ytimg.com/vi/Hg6L_7qLIEQ/default.jpg
most likes:Hg6L_7qLIEQ,https://i.ytimg.com/vi/Hg6L_7qLIEQ/default.jpg
most dislikes:Hg6L_7qLIEQ,https://i.ytimg.com/vi/Hg6L_7qLIEQ/default.jpg
category_id:10
most views:MBdVXkSdhwU,https://i.ytimg.com/vi/MBdVXkSdhwU/default.jpg
most likes:MBdVXkSdhwU,https://i.ytimg.com/vi/MBdVXkSdhwU/default.jpg
most dislikes:1NyMSWqIJDQ,https://i.ytimg.com/vi/1NyMSWqIJDQ/default.jpg
category_id:15
most views:evvVtqmvE5w,https://i.ytimg.com/vi/evvVtqmvE5w/default.jpg
most likes:evvVtqmvE5w,https://i.ytimg.com/vi/evvVtqmvE5w/default.jpg
most dislikes:U2CqZNd6rgM,https://i.ytimg.com/vi/U2CqZNd6rgM/default.jpg
category_id:17
most views:LcZ2AuvxXNA,https://i.ytimg.com/vi/LcZ2AuvxXNA/default.jpg
most likes:LcZ2AuvxXNA,https://i.ytimg.com/vi/LcZ2AuvxXNA/default.jpg
most dislikes:LcZ2AuvxXNA,https://i.ytimg.com/vi/LcZ2AuvxXNA/default.jpg
category_id:19
most views:wGQtrwey-TI,https://i.ytimg.com/vi/wGQtrwey-TI/default.jpg
most likes:wGQtrwey-TI,https://i.ytimg.com/vi/wGQtrwey-TI/default.jpg
most dislikes:wGQtrwey-TI,https://i.ytimg.com/vi/wGQtrwey-TI/default.jpg
category_id:20
most views:SNGWh_-R1VE,https://i.ytimg.com/vi/SNGWh_-R1VE/default.jpg
most likes:SNGWh_-R1VE,https://i.ytimg.com/vi/SNGWh_-R1VE/default.jpg
most dislikes:zHdJwBrT3WA,https://i.ytimg.com/vi/zHdJwBrT3WA/default.jpg
category_id:22
most views:D59v74k5flU,https://i.ytimg.com/vi/D59v74k5flU/default.jpg
most likes:cLdxuaxaQwc,https://i.ytimg.com/vi/cLdxuaxaQwc/default.jpg
most dislikes:I0Se3ce433Q,https://i.ytimg.com/vi/I0Se3ce433Q/default.jpg
category_id:23
```

6. Conclusion :

Ce projet était une opportunité qui nous a poussé à donner le maximum et bien exploiter nos connaissances sur le Big data.

Nous tenons aussi tous à remercier notre prof Mme. Soussi pour ses efforts et son aide ainsi que pour nous avoir donné l'opportunité de pratiquer les outils acquis et de développer nos connaissances en ce module.