TP Map-Reduce

Nadia Benmoussa

Novembre 2019

1 Prise en main

1.1 Exécution locale

1.1.1 Question 1

- ***Map input records*** signifie le nombre de couples (cle,value) en entrée de map ce qui correspond au nombre de ligne entrées dans le Mapper.
- ***Map output records*** signifie le nombre de couples (cle,valeur) en sortie de map ce qui correspond au nombre de mots total trouvé dans le fichier d'entrée.
- le lien entre Map output records et Reduce input records, c'est que le Map output records est égale au Reduce input records
- ***Reduce input groups*** signifie le nombre de clés différentes. Dons Mon cas c'est le nombre de mots différents dans le fichier .txt, c'est à dire si par exemple mon fichier contient : a b c d a alors ***Reduce input groups*** sera égal à 4.

1.2 Premier contact avec HDFS

1.2.1 Question 1

Le chemin, dans HDFS pour mon répertoire personel est : /user/benmousn/ et pour afficher son contenu, on tape : **hdfs dfs -ls /user/benmousn/**

1.3 Exécution sur le cluster

```
2019-11-25 11:52:18,895 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/st
aging/benmousn/.staging/job_1573555016705_0179
2019-11-25 11:52:19,125 INFO input.FileInputFormat: Total input files to process : 5
2019-11-25 11:52:19,248 INFO mapreduce.JobSubmitter: number of splits:5
2019-11-25 11:52:19,403 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1573555016705_0179
2019-11-25 11:52:19,405 INFO mapreduce.JobSubmitter: Executing with tokens: []
```

Sur la ligne 4 de la trace on a :

2019-11-25 11:52:19,248 INFO mapreduce.JobSubmitter: number of splits:5

Donc le nombre de splits est égal à 5, ce compteur correspond au nombre de fichiers lus 5 en input (5 tomes des Misérables).

1.4 Combiner

1.4.1 Question

- Les compteurs qui permettent de vérifier que le combiner fonctionnent sont: Combine input records" et "Combine output records"
- Les compteurs qui permettent d'éstimer le gain effectivement apporté par le Combiner sont Map output materialized bytes, Combine input records, Combine output records ,Reduce shuffle bytes, Spilled Records. Les valeurs trouvés sans le combiner sont plus grandes que celles avec le combiner par exemple pour le compteur Map output materialized bytes sans le combiner est égal : 4977046 et avec le combiner, il est égal à 1303807.
- Le mot le plus utilisé dans les Misérables est : *de* avec 16757 occurrences, afin de connaître cette valeur j'ai exécuté la commande suivante : sort -k 2,2 -n fichierAtester cette commande permet de trier le fichier selon la deuxième colonne.

1.5 Nombre de reducers

1.5.1 Question

La différence entre le répertoire des résultats obtenu ici et celui de la partie 1.3 est : Dans cette section on a fixé le nombre de reducers à 3 donc on retrouve 3 fichiers résultats dans le répertoire. Une clé est présente dans un seul fichier c'est à dire si la clé "bonjour" est présente dans le fichier part-r-00000, on va pas la retrouver dans un autre fichier. L'ensemble des 3 fichiers corresponds au fichier obtenue avec un seul reducer.

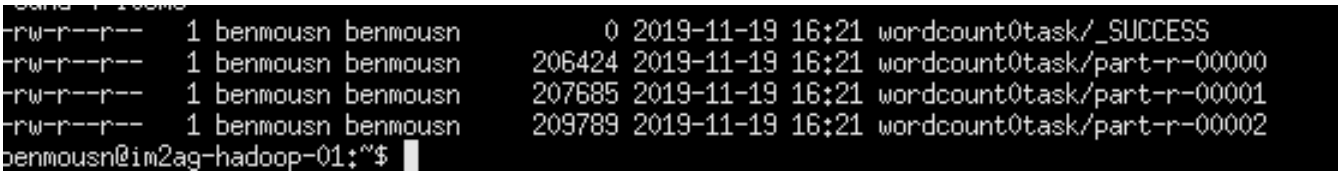

Figure 1: fichier résultat

wc -l part-r-00000 – 17401 part-r-00000
wc -l part-r-00001 – 17490 part-r-00001
wc -l part-r-00002 – 17663 part-r-00002
wc -l part-rn-00000 – 52554 part-rn-00000

2 Top-tags Flickr par pays, avec tri en mémoire

2.1 Map et Reduce

2.2 Combiner

- Pour pouvoir utiliser un combiner, il faudrait utiliser le type de données suivant : Une variable tag qui est de type Text et Une variable nombre d'occurence "occur" de type int. Le map écrit un couple clé/valeur qui représente pays/StringAndInt(tag,1)
- Concernant l'exécution sur le fichier /data/flicker, les 5 tags les plus utilisés en France sont:
  - france 35392
  - paris 24254
  - barcelona 12468
  - spain 9779
  - europe 6170
- Dans le reducer, on a une structure en mémoire dont la taille dépend du nombre de tags distincts, cette structure est la hashmap, le problème ca sera un débordement de mémoire, si par exemple on a des un certain nombre de tags très grand, un débordement de mémoire est généré .

3 Top-tags Flickr par pays, avec mémoire limitée

Les deux jobs nécessaires sont job1 qui compte le nombre d'occurence des tags par pays et le job2 trie le resultat et récupère les K tags les plus populaires .

Note : J'ai commencé à implémenter une version mais elle marche pas correctement.

4 Description et Explication des codes sources

4.1 Word Count

Le texte du fichier texto.txt en entrée est converti en mots pour former une paire clé-valeur avec tous les mots présents dans le fichier texte en entrée.Donc pour cela on récupère ce texte de la manière suivante "String lignes= value.toString()", "value" est le text d'entrée qui est de type TEXT. Une clé est formée d'un mot du fichier d'entrée et une valeur qui est à '1'. Par exemple si notre texte c'est "bonjour bonjour hello hi ohayo", le Mapper divise cette chaîne de caractères en mots (Tokens).

Dans ce cas, la phrase entière sera divisée en 5 mot (token) avec la valeur 1 comme suit: (bonjour,1) , (bonjour,1) , (hello,1) , (hi,1) (ohayo,1).

Le Reducer prend en entrée la sortie de la phase mapper , puis réduit les paires clé-valeur en clés uniques en faisant l'addition des valeurs de clés similaire. Dans l'exemple précédent, pour la clés "bonjour" on voit qu'elle est répétée deux fois donc le mapper va construire une paire (bonjour , 2).

Après l'exécution de la phase de réduction du programme sur l'exemple précédant on aura un ensemble de paires comme suit: (bonjour , 2) , (hello,1) , (hi,1) (ohayo,1).

4.2 Top-tags flickr par pay, avec tri en mémoire : Map et reduce

Dans le mapper (fichier Question21.java), on envoie le couple clé/valeur: pays/tag, puis dans le reducer on récupère le K donné en entrée (pour calculer les K tags utilisés par pays), on remplie une structure de donnée hashmap en associant pour chaque tag, son ocuurence, on utilise la classe MinMaxPriorityQueue afin de ne conserver que les K tags les plus uilisés. Enfin le reduer renvoie (pays,listeTagsOccurences) avec listeTagsOccurences est une chaine de caractère qui contient la liste des tags suivies de leurs occurences.

4.3 Top-tags flickr par pay, avec tri en mémoire : Combiner

Dans cette version on a implémenté une classe combiner ou on envoie tous les couples pays/StringAndInt(tag, occurence) puis dans le reducer on envoie les couples pays/listeTagsOccurences tel que les tags envoyés sont les K tags les plus populaires