Mineração de Dados Complexos
Curso de Aperfeiçoamento
INF-0617 - Big Data

Map Reduce Local
INF-0617 - Hadoop Streaming

Rafael Fernando Ribeiro
Thiago Gomes Marçal Pereira

Prof. Lucas Wanner

Outubro de 2018

Apresentação do Problema

Neste trabalho, recebemos os dados da eleição para governador e deputados do estado de São Paulo do ano de 2014.

O objetivo era obter a quantidade de votos que cada candidato a governador obteve no final da votação.

Solução

O projeto consistiu em uma implementação de MapReduce, em Python.

Para o Mapper, a partir dos dados passados à aplicação, foram feitas algumas análises iniciais nos dados fornecidos antes de iniciarmos os scripts. Foi observado que o campo correspondente ao cargo, que está contido no 11 campo da cadeia de caracteres contém o campo igual a “3” quando o cargo diz respeito a governador, este dado foi utilizado no filtro desenvolvido no mapper que apresentaremos no próximo parágrafo.

Um script de map foi criado para ler o arquivo fornecido em formato CSV, o arquivo estava formatado como “Latin 1” e para que fosse possível carregá-lo tivemos que alterar a forma de como a entrada padrão carrega os dados, para que o mesmo suporte a codificação “Latin 1”. Após a leitura dos dados linha a linha, utilizamos a biblioteca csv reader para carregar e fazer a separação dos dados pelo carácter “;”. Seguindo o processamento, fizemos um filtro no campo 11 que corresponde ao cargo pretendido, no caso para governador o filtro foi para o valor igual a “3” e caso a linha contenha esse valor, selecionamos as colunas 13 e 14 que correspondem ao código do partido do candidato a governador e a quantidade de votos obtidos naquela seção eleitoral. Os valores “96” e “95” correspondem aos votos NULOS e BRANCOS, estes dados também foram sumarizados.

O próximo passo foi criar o script de reducer, ele é o responsável por condensar o somatório dos valores pelo número do partido. O script lê cada linha repassada pelo script de map e faz a separação dos valores por “,”. o resultado disto são 2 campos contendo o código do partido do candidato e a quantidade de votos. O código do candidato é utilizado como chave para reduzir os dados. Sendo assim, para cada linha verificamos se o código do candidato já foi processado anteriormente, caso afirmativo, a quantidade de votos que foi lida no reducer é somada a quantidade anterior, caso contrário, criamos uma nova chave no dicionário e adicionamos a quantidade de votos no mesmo.

Ao final o reducer passa por cada chave do dicionário e imprime os valores de cada candidato. Na saída final da nossa execução dos scripts de map e reduce obtivemos o resultado ao lado. O resultado foi conferido com a página do TSE e com a wikipedia.

```
45 12230807
15 4594708
13 3888584
96 2374946
95 2020613
43 260696
50 187487
31 132042
28 22822
21 12958
29 11118
```

Para executar o trabalho siga estes passos:

1 - dentro da sua pasta data do container docker, crie uma pasta chamada **“votes”**

2 - copie o arquivo **“votacao_secao_2014_SP.txt”** dentro da pasta **“votes”**

3 - copie os scripts **“map_votes.py”** e **“reduce_votes.py”** para a pasta data

4 - dentro da instância master to hadoop vá até a pasta **/tmp/data** e execute o comando:

```
$HADOOP_HOME/bin/hadoop  jar \
$HADOOP_HOME/share/hadoop/tools/lib/hadoop-streaming-3.0.0.jar \
-input /tmp/data/votes \
-output /tmp/data/votes_out \
-mapper "python /tmp/data/map_votes.py" \
-reducer "python /tmp/data/reduce_votes.py"
```

5 - para visualizar o resultado:

```
$HADOOP_HOME/bin/hdfs dfs -cat /tmp/data/votes_out/part-00000 | sort -k2nr
```