Homework

Use the user `hadoop`.

HDFS First Hands-On

Get a feeling how to navigate/operate in `hdfs`. To start, here are some useful commands:

```
# creating a folder
hdfs dfs -mkdir /dataset

# list folders
hdfs dfs -ls /

# download The Adventures of Sherlock Holmes
wget https://www.gutenberg.org/files/1661/1661-0.txt -O ~/holmes.txt

# see the content of the first lines of the file
head holmes.txt

# how many lines does the book have?
wc -l holmes.txt

# put the file to hdfs
hdfs dfs -put ~/holmes.txt /dataset/

# list the folder content in hdfs
hdfs dfs -ls /dataset

# show the content of a file
hdfs dfs -cat /dataset/holmes.txt
```

> On your computer, check if you can find the files http://bdlc-XX.el.eee.intern:9870/explorer.html#/

Run a word count on the `holmes.txt` file:

```
~/hadoop/bin/hadoop jar ~/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-examples-3.3.1.jar wordcount /dataset/holmes.txt /test
```

> Can you find the results on the web-interface *and* via the console? Look at the Map Reduce Job in http://bdlc-XX.el.eee.intern:8088/cluster and try to go to the history of the word count job.

Delete the test folder afterwards:

```
hdfs dfs -rm -r /test
```

Own MapReduce with Python

First, we will write a `mapper.py` and a `reducer.py` in python. As you have learned, we will use the `hadoop-streaming` library. You will first implement the code with a small text file and test the implementation with unix commands only. In the next step, you will use `hadoop`'s `MapReduce`.

Setup

As `hadoop` on your machine, create a new folder `/home/hadoop/word_count`. Change to the folder and create a new test text file, called `text.txt`, with the content:

```
Der aus dem englischen Sprachraum stammende Begriff Big Data steht in
engem Zusammenhang mit dem umfassenden Prozess der Datafizierung und
bezeichnet Datenmengen, welche beispielsweise zu gross, zu komplex, zu
schnelllebig oder zu schwach strukturiert sind, um sie mit manuellen und
herkömmlichen Methoden der Datenverarbeitung auszuwerten.
```

Solution Code for the Mapper and the Reducer

I've provided a solution in the folder python_solution. Nevertheless, try to find a solution on your own and use this folder as a backup.

The Mapper

For the word count, we want to produce `<key, 1>` outputs, where the `key`s are the individual words. In the `hadoop-streaming` library the default is to separate the key from the value with a `tab`.

Create a file `mapper.py` and start with the following template:

```
#!/usr/bin/python3
import sys

for line in sys.stdin:
    line = line.strip()
    # your code here
    # emmit each word with word \t 1
```

Make sure that the code has executable permissions by invoking `chmod 755 mapper.py`. Try to run your implementation with:

```
cat text.txt | python mapper.py
```

If the mapper is implemented successfully, you should see:

```
cat text.txt | python mapper.py
Der 1
aus 1
dem 1
englischen  1
...
```

The next step is to simulate the sorting of the map reduce framework. There is a tool in Unix called `sort`:

With:

```
cat text.txt | python mapper.py | sort -k1,1
```

you will see the output sorted by the keys.

The Reducer

Create a file `reducer.py` and start with the following template:

```
#!/usr/bin/python3

import sys

for line in sys.stdin:
    line = line.strip()
    # your code here
    # count the individual words
```

Again, give the executable permission with `chmod 755 reducer.py`.

You can test the whole pipeline with:

```
cat text.txt | python mapper.py | sort -k1,1 | python reducer.py
```

E.g. 'zu' should have a count of four.

Run the Word Count with Map Reduce

Try to run you word count with `MapReduce`.

With the smaller `text.txt`

```
hdfs dfs -put /home/hadoop/word_count/text.txt /dataset/
```

```
hadoop jar ~/hadoop/share/hadoop/tools/lib/hadoop-streaming-3.3.1.jar \
-files /home/hadoop/word_count/ \
-mapper /home/hadoop/word_count/mapper.py \
-reducer /home/hadoop/word_count/reducer.py \
-input /dataset/text.txt \
-output /own_word_count_small_file
```

With `holmes.txt`

```
hadoop jar ~/hadoop/share/hadoop/tools/lib/hadoop-streaming-3.3.1.jar \
-files /home/hadoop/word_count/ \
-mapper /home/hadoop/word_count/mapper.py \
-reducer /home/hadoop/word_count/reducer.py \
-input /dataset/holmes.txt \
-output /own_word_count
```