Spark Search Engine

Done by Artur Samigullin

This Notebook shows how to make indexing with a Spark Search Engine Library on a small use case

Part I. Indexing Dataset

Initialize Contexts

First of all, to work with Spark Search Engine you need to import pyspark library and initialize SparkContext and SQLContext

In [1]:

```
import findspark
findspark.init()

import pyspark
sc = pyspark.SparkContext()

from pyspark.sql import SQLContext
sqlc = SQLContext(sc)
```

Import SearchEngine class

At this step you need to import SearchEngine class from SparkSearchEngineLib.SearchEngine package

In [2]:

```
from SparkSearchEngineLib.SearchEngine import SearchEngine
```

Initialize instance of SearchEngine class

You need to pass two parameters to SearchEngine constructor - SparkContext and SQLContext

In [3]:

```
se = SearchEngine(sc,sqlc)
```

Index your dataset

We assume that you made all preprocessing for your files, and we expect a folder that consists of textual files in format 'Token0 Token1 ... TokenN'

In [4]:

```
se.construct_index('./Dataset/')
```

Part II. Use Search

To use search you need to have an SearchEngine instance with constructed index. You can make it with *search()* method.
search() method has one parameter - preprocessed query string with format 'Token0 Token1 ... TokenN'
Method returns a list of links(filenames) with number of hits.

In [5]:

```
find_Python = se.search('Python')
find_Python.take(10)
```

Out[5]:

```
['file:/Users/deusesx/Projects/P&MP/Dataset/3.txt has number of hi
ts: 2',
 'file:/Users/deusesx/Projects/P&MP/Dataset/1.txt has number of hi
ts: 1']
```

In [6]:

```
find_program = se.search('program')
find_program.take(10)
```

Out[6]:

```
['file:/Users/deusesx/Projects/P&MP/Dataset/1.txt has number of hi
ts: 2',
 'file:/Users/deusesx/Projects/P&MP/Dataset/2.txt has number of hi
ts: 1']
```

In [7]:

```
find_both = se.search('program Python')
find_both.take(10)
```

Out[7]:

```
['file:/Users/deusesx/Projects/P&MP/Dataset/1.txt has number of hi
ts: 3',
 'file:/Users/deusesx/Projects/P&MP/Dataset/3.txt has number of hi
ts: 2',
 'file:/Users/deusesx/Projects/P&MP/Dataset/2.txt has number of hi
ts: 1']
```

Part III. Files Manipulation

You can store your index in a *parquet* format. To make this just use *save_index( )* method.
save_index( ) method has one parameter - string with filename.
Note that if filename is already exists, it will be overwritten by *save_index( )* method.

In [8]:

```
se.save_index('index.parquet')
```

In [9]:

```
se2 = SearchEngine(sc, sqlc)
```

You can load index from *parquet* format with *load_index( )* method.
load_index( ) method takes one parameter - filename.

In [10]:

```
se2.load_index('index.parquet')
```