
Lab 2 : Wikipedia

Big Data Analysis

Quentin Vaucher, André Neto da Silva, Sylvain Renaud

Hes·SO

Haute Ecole Spécialisée
de Suisse occidentale

Fachhochschule Westschweiz

University of Applied Sciences and Arts
Western Switzerland

April 3, 2019

Contents

1	Raw results	1
2	Interpretations	2
2.1	Naive	2
2.2	Inverted index	2
2.3	Reduce by key	3
3	Wikipedia-based VS RedMonk rankings	3

1 Raw results

The list of retrieved languages is the same independently from the method which is used. This ascertainment seems logic since each attempt tries to compute the same result changing only the way the computations are done.

The following table presents the result obtained, regardless of the attempt.

Table 1: List of languages ranked by number of articles

Rank	Language	# articles
1	JavaScript	1704
2	C#	731
3	Java	700
4	CSS	430
5	Python	409
6	C++	384
7	PHP	333
8	MATLAB	296
9	Perl	176
10	Ruby	161
11	Haskell	65
12	Objective-C	62
13	Scala	53
14	Clojure	29
15	Groovy	29

The analysis becomes more interesting when it targets the time of computations. Here are the comparison of the three different attempts:

Attempt name	Time [s]
Naive	81121
Inverted index	9463
Reduce by key	6359

2 Interpretations

Let's try to understand these results one by one, beginning by the naive implementation.

2.1 Naive

For each language, the code go through the entire RDD and count the number of articles containing its name. Therefore the computations are the followings:

$$N * M$$

where:

- N is the number of languages
- M is the size of the RDD

2.2 Inverted index

For each language, the code has to calculate the size of its inverted index. Therefore, the computations are the followings:

$$\sum_{i=0}^N \text{size of inverted index } i$$

where:

- N is the number of Languages

Compared to the naive implementation, we don't have to go through the entire RDD, and don't ever bother to check which article contains which language because this information is already known.

2.3 Reduce by key

TODO...

3 Wikipedia-based VS RedMonk rankings

Finally, let's see how close is this Wikipedia-based ranking to the popular RedMonk ranking. The list which was given in the Lab has been slightly modified in order to match the one given here: RedMonk ranking - June 2018. Because the list has been "pre-filtered" to match the RedMonk one, only the order relationship is relevant here.

Rank	Wikipedia-based	RedMonk
1	C	JavaScript
2	R	Java
3	Java	Python
4	JavaScript	PHP
5	Go	C#
6	C#	C++
7	CSS	CSS
8	C++	Ruby
9	Python	C
10	PHP	Objective-C
11	Ruby	Swift
12	Scala	Scala
13	Shell	Shell
14	Objective-C	Go
15	Swift	R

- CSS, Scala and Shell are ranked in the same way
- Java, C# and C++ are one or two rank away from the RedMonk ranking

Even though the ranking is not exactly the same, the general idea seems to match pretty well the RedMonk rank.