

Data Mining 21/22 Homework 2

Marco Calamo

November 2021

All work is tracked on Github at https://github.com/I1Kaiser/DM22_HW

1 Exercise 1 - Search Engine

In order to implement a Cosine Similarity-based search engine, I did the following steps:

1. Using *nltk* , I pre-processed, tokenized (using white space as separator) and normalized the text descriptions and the titles of the jobs by:
 - generating the tokens
 - converting all tokens to lowercase
 - removing punctuation
 - filtering out Italian stop words (since the dataset is in Italian)
 - stemming the tokens
2. I created a simple *inverted index* as a python dictionary, where every pre-processed token is the key and the value is a list of document where the token occurs with the number of total occurrences in the corpus
3. Using the inverted index as a basis, I created the **TF,IDF** and **TFIDF** indexes as python dictionaries:
TF has a token per key and a dictionary (where document index is the key) with $\frac{\text{tokensPerDoc}}{\text{allTokens}}$ as a value
IDF has a token as key and $\frac{|N|}{\text{tokenFreq}}$ as value
TFIDF has a token as key and a dictionary (where document index) with tf-idf product per document as value
4. I saved all computed indexes to files in JSON format as requested
5. I computed Cosine Similarity by pre-processing with the same steps described before, and by comparing the resultant tf-idf query vector with all tf-idf vectors in dataset
6. I printed on screen the top 10 most matching result
7. I used the **Tensorflow** library with *tf-idf vectorizer* as a benchmark

1.1 Results

Figure 1: "Programmatore Java" as query

```

query: programmatore java con molta esperienza

#####
# MY RESULTS #####
title: Programmatore Java full stack score(similarity): 0.301397689836685
title: Analista Programmatore Java / 2m score(similarity): 0.15659315628137
title: Cocco programmatore Java / 2m score(similarity): 0.15659315628137
title: Seo con esperienza score(similarity): 0.11218127032735047
title: Analista programmatore Java / 2m score(similarity): 0.10884738495245
title: Consulente Informatico SAP Stag score(similarity): 0.10884738495245
title: Cocco score(similarity): 0.093493050804859
title: Esperto programmatore prestando score(similarity): 0.0933772787298603
title: Esperto programmatore prestando score(similarity): 0.0933772787298603
title: Analista Programmatore Java / 2m score(similarity): 0.088486073972568
title: Programmatore Analista / Categorie Profette score(similarity): 0.088486073972568
title: Analista Programmatore Java score(similarity): 0.079298522995330
title: Analista programmatore Java / 2m score(similarity): 0.079298522995330
title: Analista programmatore Java score(similarity): 0.07938717415836988
title: Analista programmatore Java / 2m score(similarity): 0.07938717415836988
title: Aziende di modo cerca varie figure score(similarity): 0.0732821738427496
title: Analista Programmatore Java / 2m score(similarity): 0.071819797407035

#####
# TENSORFLOW RESULTS #####
title: Analista Programmatore Java / 2m score(similarity): 0.1516978774305997
title: Analista Programmatore Java / 2m score(similarity): 0.1516978774305997
title: Analista Programmatore Java / 2m score(similarity): 0.1495428515805673
title: Analista Programmatore Java / 2m score(similarity): 0.1398265539889883
title: Analista Programmatore Java / 2m score(similarity): 0.138713380378776
title: Analista Programmatore Java / 2m score(similarity): 0.138713380378776
title: Analista Programmatore Java / 2m score(similarity): 0.137844571238429
title: Analista Programmatore Java / 2m score(similarity): 0.13512268989468
title: Analista Programmatore Java / 2m score(similarity): 0.1348646650857668
title: Analista Programmatore Java / 2m score(similarity): 0.13477805648322367
title: Analista Programmatore Java / 2m score(similarity): 0.13477805648322367
title: Analista Programmatore Java / 2m score(similarity): 0.1337579965584392
title: Analista Programmatore Java / 2m score(similarity): 0.1337579965584392
title: Analista Programmatore Java / 2m score(similarity): 0.14256556520564

```

Figure 2: complex query

```

query: programmatore

#####
# MY RESULTS #####
title: Programmatore Java / 2m score(similarity): 0.237224857598166738
title: Cocco programmatore Java / 2m score(similarity): 0.232440150593745
title: Cocco score(similarity): 0.232440150593745
title: Cocco score(similarity): 0.19235208462084
title: Cocco score(similarity): 0.19235208462084
title: Cocco score(similarity): 0.237224857598166738
title: Programmatore / net e ul score(similarity): 0.1881286426545476
title: Analista programmatore / net e ul score(similarity): 0.1881286426545476
title: Analista programmatore score(similarity): 0.1758036956496118
title: Analista programmatore / net e ul score(similarity): 0.1758036956496118
title: Programmatore score(similarity): 0.1647404045740325
title: Sviluppatore Software, Analista Programmatore / net e ul score(similarity): 0.14581597405863253
title: Cocco net e ul programmatore per siti web score(similarity): 0.1364212569505825
title: Analista programmatore / analista programmatore score(similarity): 0.1364212569505825
title: Social programmatore / analista score(similarity): 0.1294267210454888
title: Programmatore / analista Senior score(similarity): 0.1294267210454888
title: Programmatore PHP wordress wocommerce a Torino score(similarity): 0.22874308615681847
title: Programmatore PHP wordress wocommerce a Torino score(similarity): 0.22874308615681847

#####
# TENSORFLOW RESULTS #####
title: Programmatore PHP wordress wocommerce a Torino score(similarity): 0.351773234425687
title: Programmatore PHP wordress wocommerce a Torino score(similarity): 0.3509383678644124
title: Programmatore PHP wordress wocommerce a Torino score(similarity): 0.2631762685176754
title: Programmatore PHP wordress wocommerce a Torino score(similarity): 0.2510371596123687
title: Programmatore PHP wordress wocommerce a Torino score(similarity): 0.24279367577598417
title: Programmatore PHP wordress wocommerce a Torino score(similarity): 0.2226462935154635
title: Programmatore PHP wordress wocommerce a Torino score(similarity): 0.2226462935154635
title: Programmatore PHP wordress wocommerce a Torino score(similarity): 0.2287586239628743
title: Programmatore PHP wordress wocommerce a Torino score(similarity): 0.2287586239628743
title: Programmatore PHP wordress wocommerce a Torino score(similarity): 0.2186856880222058
title: Programmatore PHP wordress wocommerce a Torino score(similarity): 0.2027025181050563
title: Programmatore PHP wordress wocommerce a Torino score(similarity): 0.1921996987466472
title: Programmatore PHP wordress wocommerce a Torino score(similarity): 0.1775210454888
title: Programmatore PHP wordress wocommerce a Torino score(similarity): 0.1775210454888

```

As we can see my implementations filters out duplicates and does even a better job when it is required to match an entire description.

For more details see *search.py*

Figure 3: whole job as a query

```
Fai ESR View Bookmarks Plugins Settings Help H2 zh — Konsole
```

New tab Split View LeftRight Split View TopBottom Load a new tab with layout 2x2 terminals Load a new tab with layout 2x1 terminals Load a new tab with layout 2x1 terminals

File Edit View Bookmarks Plugins Settings Help

Dataset loaded. 100% 100%

Editor. Editor video con con approfondimenti e documentata esperienza si ricerca per postazioni interne alla struttura; si richiede 1 conoscenza approfondita di Premiere o After Effects, basata sulla conoscenza di Davinci, residenza entro 15 km da Novara Se interessati e in possesso di tutti i requisiti inviare un curriculum a job AT glissepegallo.it e link a propria show reel.

titolo: Editor Pratica - videografia-a-After Effects score(similarity): 0.95182120812286
titolo: Editor video con approfondimenti e documentata esperienza score(similarity): 0.95182120812286
titolo: Grafico score(similarity): 0.15844389204945
titolo: Grafico score(similarity): 0.15844389204945
titolo: Grafico score(similarity): 0.15844389204945
titolo: Grafico score(similarity): 0.092320993277237
titolo: Grafico/a (similarity): 0.092320993277237
titolo: Adattato a ospedale alti digitalizzazione degli archivi score(similarity): 0.97139718164982
titolo: Meccanico In Informatico score(similarity): 0.97139718164982
titolo: Meccanico In Informatico score(similarity): 0.97139718164982
titolo: Silvopasto (Julian) score(similarity): 0.8675752468056251
titolo: Web master score(similarity): 0.86463941838388
titolo: Produzione e montaggio video con Pinnacle CAMVA ecc. score(similarity): 0.863956804263534
titolo: Ricercare per programmare informatico (lavorato) score(similarity): 0.86176082234541845
titolo: Ricercare per programmare informatico (lavorato) score(similarity): 0.86176082234541845

titolo: Senior graphic designer score(similarity): 0.92716271318317
titolo: Senior graphic designer score(similarity): 0.92716271318317
titolo: Senior graphic designer score(similarity): 0.165762354938405
titolo: Senior graphic designer score(similarity): 0.165762354938405
titolo: Senior graphic designer score(similarity): 0.015390125412087
titolo: Senior graphic designer score(similarity): 0.015390125412087
titolo: Senior graphic designer score(similarity): 0.15597540913393
titolo: Senior graphic designer score(similarity): 0.15597540913393
titolo: Senior graphic designer score(similarity): 0.1558256138089268
titolo: Senior graphic designer score(similarity): 0.1558256138089268
titolo: Senior graphic designer score(similarity): 0.15489891120454
titolo: Senior graphic designer score(similarity): 0.15489891120454
titolo: Senior graphic designer score(similarity): 0.154871524938532
titolo: Senior graphic designer score(similarity): 0.154871524938532
titolo: Senior graphic designer score(similarity): 0.152752730618647
titolo: Senior graphic designer score(similarity): 0.152752730618647
titolo: Senior graphic designer score(similarity): 0.1507690523795588

Figure 4: Something not present in dataset as a query

2 Exercise 2 - Locality Sensitive Hash

2.1 Shingles

In order to create k-shingles I used a simple straightforward function in python that takes the text to shingle and k as parameters.

The function iterates over the string and outputs a k-shingles list.

An option to create 4-bytes hash signatures to save space is present as a stand-alone function.

2.2 Min Hashing

Minimum Hash signature matrix is computed for every document (job announcement) in the corpus, by applying the desired number of hash functions generated with the provided *hashFamily* utility and computing the minimum hash values per shingle among every function.

2.3 LSH

The LSH is computed with r (rows per band) and b (number of bands) as parameters on the signature matrix. On the LSH candidate pair –different documents that were a hit at some point in LSH algorithm – is computed the Jaccard Similarity on Shingles.

Finally the program outputs the top 20 most similar documents.

2.4 Results

The resulting program does a good job in duplicates and near-duplicate detection on dataset. Using 10 as r for LSH (and $\frac{|dataset|}{r}$ as b) i obtained the best results, with a good speed to correct match ratio. My program resulted being about 20x faster than brute force similarity computing, while finding about a third of all correct similarities.

Figure 5: Final report

For more details see *lsh.py*

3 Exercise 3 - Locality Sensitive Hash on Spark

The challenge of this exercise was in porting the code I wrote for the previous exercise to Spark platform using **pyspark** library.

The steps are the same form 2, but every time i needed to alter the initial dataframe I used the map function: for shingles, min-hash signatures and LSH.

3.1 Results

As expected –despite it is not accurate the way of measuring Spark execution time on python– the final results are the same compared to 2, with Spark that outperforms simple python on computing shingles, min-hash signatures and LSH candidates.

Spark, however, despite having the dataset cached for faster memory access, is slower on outputting the final results on screen since, for the way my program works, it is necessary to do many subsequent access (that result quite slow) on the dataset in order to retrieve all fields corresponding to the similar jobs.

Finally Spark was even slower than simple python in computing brute force similarity, done by joining two sets of shingles with different IDs (by avoiding to create duplicate rows) and mapping the shingles to their Jaccard Similarity. The computation took more than 20 mins, and thus its execution is discouraged; the code is available at the bottom of the file but it is commented out for fast execution of the program.

For more details see *spark_lsh.py*

4 Dataset

The file used as a dataset for benchmarking the results of my work is ***jobs.txt*** (included in the zip file), generated by *crawler.py* from previous homework in date 02/11/2021, and has the full job description available on
<https://www.kijiji.it/offerte-di-lavoro/offerta/informatica-e-web/>