**Comp9313 Assignment3 Solution**

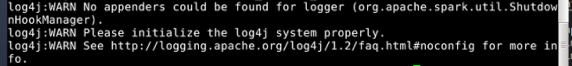
Name: PEIGUO GUAN

zID: z5143964

**Step of test:**

Firstly, I use **sbt console** to run the code, and there is not error occurred. And in order to distinguish from others’ /legal\_idx/cases/, I use /legal\_idxgg11/cases/ to check my result.

And then I use spark-submit to run my code, and change my index and again, although I got this error below(I still do not why and the solution in the forum cannot solve my problem), I still got the same result as expected and run successfully.

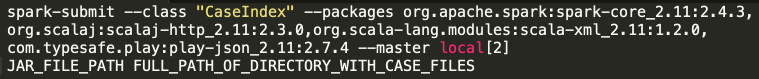


**The explanation of my code and logic:**

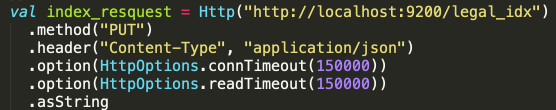
Firstly, get read the file and get all the case test file path by function: ***getAllFiles***, then setting ***index\_request*** and **mapping\_request** to elasticsearch. I use the function ***processAllFile*** function to process all the next steps: 1.load xml file and separate each part 2.pick the sentences part and send to the corenlp to parse sentence and return ***person***, ***location***, and ***organization*** 3.reconstruct, filter, get tokens… to different ***\_list***, and build a ***es\_post\_data***, PUT es\_post\_data to the elasticsearch.

**Running part:**

**Spark-submid command:**



my index design:



my mapping:



(please ignore the \n Part in post data)

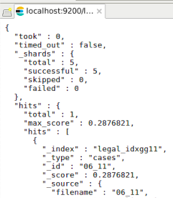
There are 8 parts in my mapping:

1. filename: is for the file name in case\_test
2. name: is the name in file
3. AustLII: is for the AustLII in file
4. catchphrases: for the Catchphrases in test file
5. sentences: for the sentences in test file
6. person: for the person after processing by corenlp
7. location: for the location after processing by corenlp
8. organization: for the organization after processing by corenlp

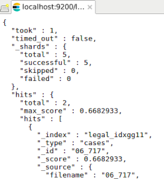
**Result of example queries:(I create my own legal\_idxgg11 to distinguish from others in real test)**

1. queries based on entity type:

curl -X GET <http://localhost:9200/legal_idx/cases/_search?pretty&q=location:Melbourne>



curl -X GET <http://localhost:9200/legal_idx/cases/_search?pretty&q=person:John>



1. query based on general term:

curl -X GET [http://localhost:9200/legal\_idxgg11/cases/\_search?pretty&q=(criminal AND law)](http://localhost:9200/legal_idxgg11/cases/_search?pretty&q=(criminal%20AND%20law))

(which is actually: (criminal%20AND%20law))

