CS1002 W05 Practical 160026526 Tutor: Ruth Letham 21 Feb 2018

- Overview:

The aim of the practical was to allow for the refinement of data manipulation skills which includes choosing appropriate representation of data in memory, identifying possible errors, dealing with the errors and testing. The practical required a program that must first be able to accept two command line arguments, the first being the text file while the second is a query string. The program should then be able to read the text file and split it into sentences while reverting all capitalizations, by sanitisation. The query string is also be sanitised and then a similarity score is calculated between the query sentence and each sentence in the file using the Jaccard Index. The program will lastly print 50 sentences with the highest similarity scores into the output along with the similarity scores in descending order.

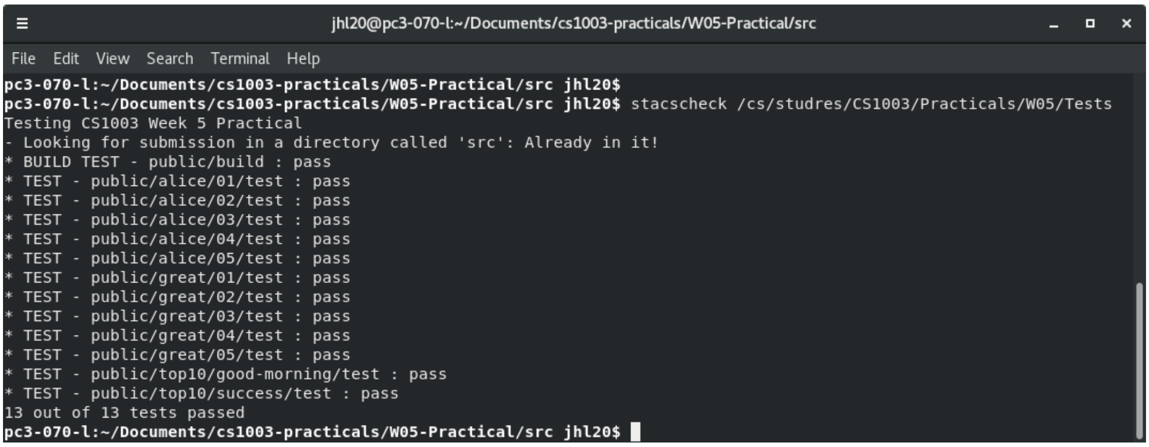
- Design:

The practical starts off with two classes provided by the documents which are ScoredResult and SentenceReader. The program must first be able to read text files therefore the SentenceReader should be edited first as the readAllSentences method is empty. An array list called allSentences is first created which will be used later to store all the sentences after they have been split and sanitised. The readAllSentences will take in a String called filepath which will later be used to take in an argument. A try block is created which converts the filepath to a Path which will be read into bytes and converts all the text into a string and initialises it into input. An array is then created to store all the sentences after splitting the sentences. Each sentence must then be sanitised therefore a for loop is created that loops around the length of the amount of sentences and will perform the sanitise method on each sentence in the array and finally add them into the allSentences array. Two exceptions should then be catched which are the FileNotFoundException and the IOException. A list must be returned therefore the allSentences array list is returned.

Moving on, the W05Practical class is created which will be used to run the program. A new sentence reader is created with the name reader and a list called sentences is created which uses the readAllSentences method to take in an argument, args[0]. The argument taken will be an input text file according to the specification. The list will contain all the sentences which will be sanitised after using the readAllSentences method therefore do no to be sanitised again and the sentences have been splitted by the same method. The next method should be able to turn the sentences into bigrams which will be used in Jaccard Index calculations. A method called bigramize is created to take String sentences and is set to static as the method should constantly perform the same task as it is a utility. An array list called bigram is created in the method which will be used to hold all the bigrams. The method should read all the sentences and change them into bigrams therefore a for loop is created to loop around sentence.length() – 1 which ensures that they array will stay in the boundaries. Substring is used to get a subset of a string and therefore i is used to get the first character and i+2 is used to get the second character as the endIndex is exclusive where else the startIndex is inclusive. The method will then return the array list of bigrams. Now that there is a bigram converting method we proceed back to the main method. Two array lists are created which will store the bigrams of the candidate sentences from the text file and the bigrams of the query string. These are called bigramCandidate and bigramQuery respectively. A double called jaccardIndex is set at 0 which will later be used to temporarily store the Jaccard Indexes, According to the ScoredResult method, the scores will be stored in an array list and therefore the array list called score is made. The sentences should be unique therefore two HashSets are created for the candidate sentences and the query sentences. To ensure all the sentences are read, a for loop is created to loop around the size of the sentences. An int called intersection is initialised which will later be used in calculating the Jaccard Index. The query string which is provided when running the program is sanitised using the sanitiseSentence method and is bigramized and stored in the bigramQuery array list. To store the bigrams of the candidate sentences, a string is created to contain the individual sentences from the text. The sentences are then bigramized and added into the bigramCandidate array list. The lists containing the bigramized query and bigramized candidates are then transferred into the respective hashsets to ensure there are no duplicateds. To calculate the intersection between the query and candidate sentences, a for loop is created to loop through the candidate bigrams in the hashset and if the query contains a candidate bigram, the loops increments the intersection by 1 and the loop is closed. The Jaccard Index also requires the union therefore union is initialized with candidate.size + query.size. The Jaccard Index is then calculated by first subtracting the intersection from the union and using the result to divide by the intersection, This should have the double type. The score array list is then updated to contain the sentences and the corresponding Jaccard Index. The first for loop is then closed and the scores are then sorted using Java Collections and are then looped to print out the 50 highest scores. It should also be mentioned that another override is written in the ScoredResult class to ensure that the score is printed to 4 decimal places. After a try catch loop was written in the main method to prompt the user on the usage of the arguments and an if statement is added to ensure the args.length is less than or equal 2 and if it is not, it notifies the user.

- Testing:

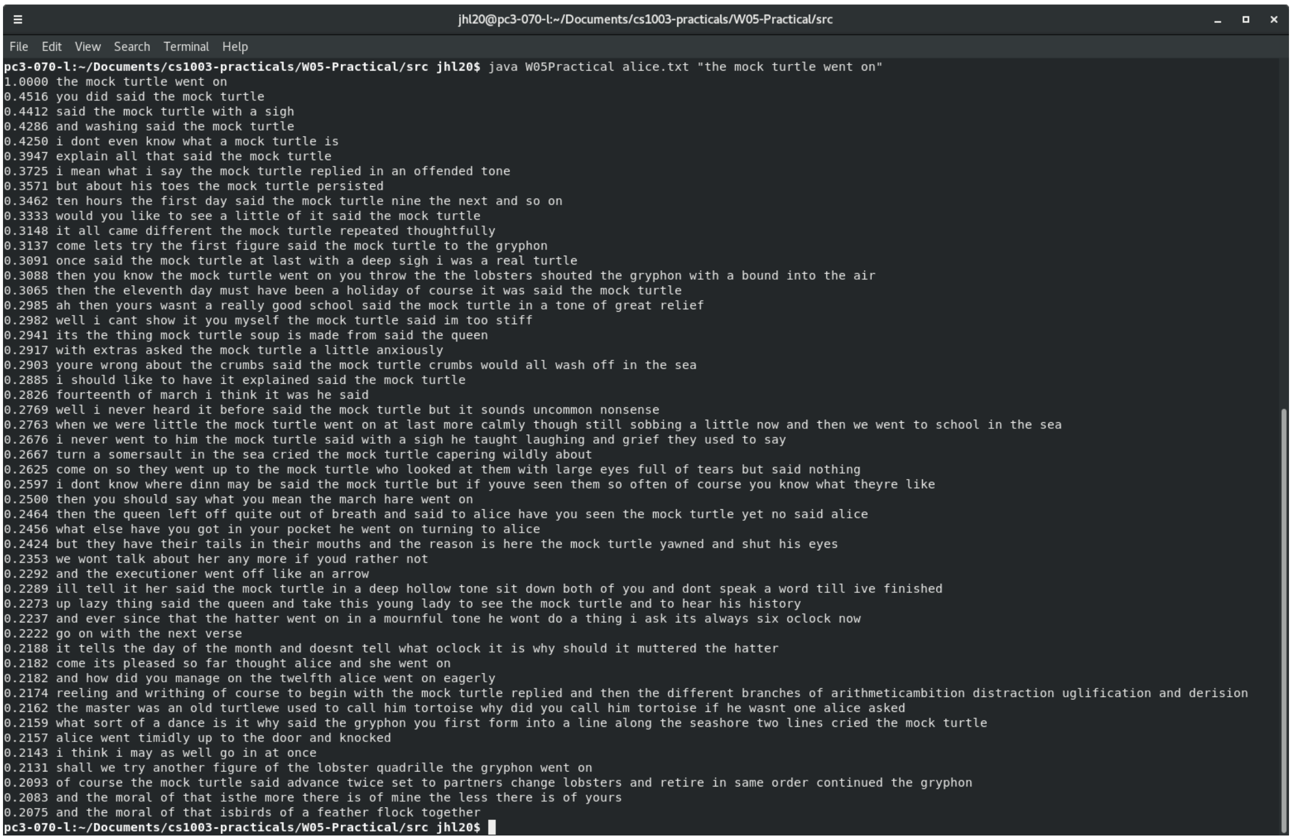
The first test to ensure that the program works according to the specification is the autochecker which is provided in the specification. The program passed all 13 of the autochecker tests.



jhl20@pc3-070-l:~/Documents/cs1003-practicals/W05-Practical/src
File
Edit
View
Search
Terminal
Help

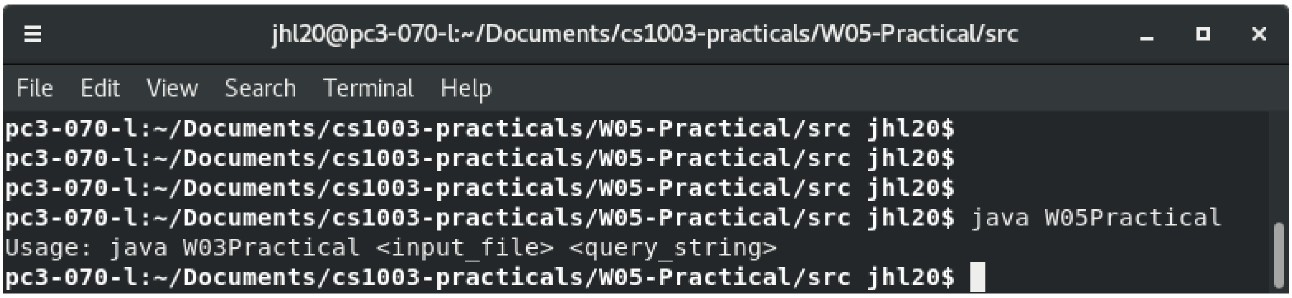
```
pc3-070-l:~/Documents/cs1003-practicals/W05-Practical/src jhl20$
pc3-070-l:~/Documents/cs1003-practicals/W05-Practical/src jhl20$ stacscheck /cs/studres/CS1003/Practicals/W05/Tests
Testing CS1003 Week 5 Practical
- Looking for submission in a directory called 'src': Already in it!
* BUILD TEST - public/build : pass
* TEST - public/alice/01/test : pass
* TEST - public/alice/02/test : pass
* TEST - public/alice/03/test : pass
* TEST - public/alice/04/test : pass
* TEST - public/alice/05/test : pass
* TEST - public/great/01/test : pass
* TEST - public/great/02/test : pass
* TEST - public/great/03/test : pass
* TEST - public/great/04/test : pass
* TEST - public/great/05/test : pass
* TEST - public/top10/good-morning/test : pass
* TEST - public/top10/success/test : pass
13 out of 13 tests passed
pc3-070-l:~/Documents/cs1003-practicals/W05-Practical/src jhl20$ 
```

A second test is carried out to check the program manually.


jhl20@pc3-070-l:~/Documents/cs1003-practicals/W05-Practical/src
File Edit View Search Terminal Help

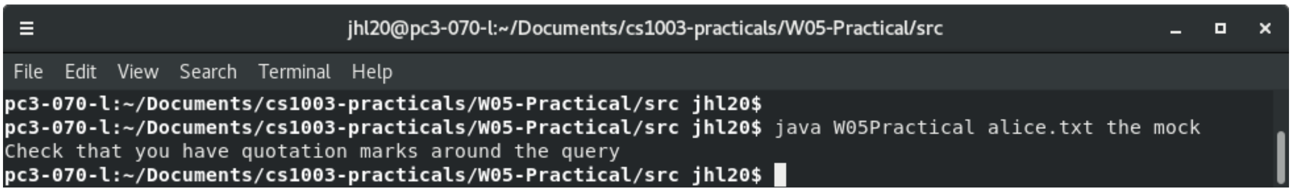
```
pc3-070-l:~/Documents/cs1003-practicals/W05-Practical/src jhl20$ java W05Practical alice.txt "the mock turtle went on"
1.0000 the mock turtle went on
0.4516 you did said the mock turtle
0.4412 said the mock turtle with a sigh
0.4286 and washing said the mock turtle
0.4250 i dont even know what a mock turtle is
0.3947 explain all that said the mock turtle
0.3725 i mean what i say the mock turtle replied in an offended tone
0.3571 but about his toes the mock turtle persisted
0.3462 ten hours the first day said the mock turtle nine the next and so on
0.3333 would you like to see a little of it said the mock turtle
0.3148 it all came different the mock turtle repeated thoughtfully
0.3137 come lets try the first figure said the mock turtle to the gryphon
0.3091 once said the mock turtle at last with a deep sigh i was a real turtle
0.3088 then you know the mock turtle went on you throw the the lobsters shouted the gryphon with a bound into the air
0.3065 then the eleventh day must have been a holiday of course it was said the mock turtle
0.2985 ah then yours wasnt a really good school said the mock turtle in a tone of great relief
0.2982 well i cant show it you myself the mock turtle said im too stiff
0.2941 its the thing mock turtle soup is made from said the queen
0.2917 with extras asked the mock turtle a little anxiously
0.2903 youre wrong about the crumbs said the mock turtle crumbs would all wash off in the sea
0.2885 i should like to have it explained said the mock turtle
0.2826 fourteenth of march i think it was he said
0.2769 well i never heard it before said the mock turtle but it sounds uncommon nonsense
0.2763 when we were little the mock turtle went on at last more calmly though still sobbing a little now and then we went to school in the sea
0.2676 i never went to him the mock turtle said with a sigh he taught laughing and grief they used to say
0.2667 turn a somersault in the sea cried the mock turtle capering wildly about
0.2625 come on so they went up to the mock turtle who looked at them with large eyes full of tears but said nothing
0.2597 i dont know where dinn may be said the mock turtle but if youve seen them so often of course you know what theyre like
0.2500 then you should say what you mean the march hare went on
0.2464 then the queen left off quite out of breath and said to alice have you seen the mock turtle yet no said alice
0.2456 what else have you got in your pocket he went on turning to alice
0.2424 but they have their tails in their mouths and the reason is here the mock turtle yawned and shut his eyes
0.2353 we wont talk about her any more if youd rather not
0.2292 and the executioner went off like an arrow
0.2289 ill tell it her said the mock turtle in a deep hollow tone sit down both of you and dont speak a word till ive finished
0.2273 up lazy thing said the queen and take this young lady to see the mock turtle and to hear his history
0.2237 and ever since that the hatter went on in a mournful tone he wont do a thing i ask its always six oclock now
0.2222 go on with the next verse
0.2188 it tells the day of the month and doesnt tell what oclock it is why should it muttered the hatter
0.2182 come its pleased so far thought alice and she went on
0.2182 and how did you manage on the twelfth alice went on eagerly
0.2174 reeling and writhing of course to begin with the mock turtle replied and then the different branches of arithmeticambition distraction uglification and derision
0.2162 the master was an old turtlewe used to call him tortoise why did you call him tortoise if he wasnt one alice asked
0.2159 what sort of a dance is it why said the gryphon you first form into a line along the seashore two lines cried the mock turtle
0.2157 alice went timidly up to the door and knocked
0.2143 i think i may as well go in at once
0.2131 shall we try another figure of the lobster quadrille the gryphon went on
0.2093 of course the mock turtle said advance twice set to partners change lobsters and retire in same order continued the gryphon
0.2083 and the moral of that isthe more there is of mine the less there is of yours
0.2075 and the moral of that isbirds of a feather flock together
pc3-070-l:~/Documents/cs1003-practicals/W05-Practical/src jhl20$
```

The third test is to check when no arguments are provided.


jhl20@pc3-070-l:~/Documents/cs1003-practicals/W05-Practical/src
File Edit View Search Terminal Help

```
pc3-070-l:~/Documents/cs1003-practicals/W05-Practical/src jhl20$
pc3-070-l:~/Documents/cs1003-practicals/W05-Practical/src jhl20$
pc3-070-l:~/Documents/cs1003-practicals/W05-Practical/src jhl20$
pc3-070-l:~/Documents/cs1003-practicals/W05-Practical/src jhl20$ java W05Practical
Usage: java W03Practical <input_file> <query_string>
pc3-070-l:~/Documents/cs1003-practicals/W05-Practical/src jhl20$
```

The final test is to check when the query is provided without the quotations.


jhl20@pc3-070-l:~/Documents/cs1003-practicals/W05-Practical/src
File Edit View Search Terminal Help

```
pc3-070-l:~/Documents/cs1003-practicals/W05-Practical/src jhl20$
pc3-070-l:~/Documents/cs1003-practicals/W05-Practical/src jhl20$ java W05Practical alice.txt the mock
Check that you have quotation marks around the query
pc3-070-l:~/Documents/cs1003-practicals/W05-Practical/src jhl20$
```

• Evaluation:
The program was required to split a text file into sentences and calculate the Jaccard Index in relations to a query string provided by a user and calculate the top 50 sentences with the highest Jaccard Indexes. The program should also sanitise the sentences in both the text file and the query string provided. Errors should be taken into consideration as done in the previous practical by using try catch blocks. The program should also run all the autochecker's tests without flaws. The program was able to successfully carry out the requirements set by the practical therefore is successful in completing the task provided.

- Conclusion:

In conclusion, the program was successful in carrying out the data manipulation in regards to the regulations placed by the practical. Difficulties were encountered when trying to run the main method but it was found out that the problem was the query string was not encased in quotations. Other than that, there were multiple ArrayOutOfBoundsExceptions that were encountered that were not fixed until the last few days in which the try catch blocks were added into the main method and the for loop to check for the argument was implemented. The Jaccard Index proved a problem as the output did not match the expected but was corrected in which the problem lied in the methods used for the HashSets. None of the extensions were carried out due to the lack of time and knowledge of the usage of outside packages. Given more time, the first extensions would have been attempted as it seems to be the simplest to implement when considering all other extensions.