

Assignment 2 Report: Latent Semantic Indexing

Name: Ke Wang NetID: kw427

On the basis of Assignment 1, I implemented this latent semantic indexing program.

1. Mathematical Basis

For LSI, we need to create the term-document matrix, and apply singular value decomposition to create an LSI representation of the matrix. So I use the function from Assignment 1 to read the test files, stop list, analyze them and store them into corresponding data structures. Calculating the weighing $tf.idf$ by function `getTF()` and `getIDF()`.

Using JAMA package of Java, I can apply singular value decomposition and get the matrices of left and right singular vectors and the diagonal matrix of singular values. And based on the resulting matrices, I can reduce the dimension of the matrices to eliminate noise from synonymy and recognizes dependence. According to the selection of k , I will get new singular values, vectors for later similarity calculation.

To compute the similarity, I first analyze the input query and convert it to the query vector to the LSI space, that is to represent a query as a pseudo document by function `getQueryV()` and `getNewQueryV()` and convert document vector in word space to concept space by `getDocV()`. After that, compute the cosine of the two vectors as the similarity measure and sort the results by cosine value in a descending order. The documents appear first are more relevant.

Also, I use $\text{cosine} \times 100$ as the score for each document, thus documents with high scores are most relevant and I only print the portion of the first 5 documents that are most relevant.

2. Test

Just run the Test.java and then input the query like Assignment 1.

Case 1: if I use all the words in the stop list or words don't in the corpus,(i.e "have") the program will not get relevant words in the corpus and it will return "sorry, there is no corresponding result".

Case 2: If query "happy summer" and I will get following result. File21 scores highest in the rank and "happy" and "summer" do exist in file 21. The score is close to 100.

results

File21: Score: 98.43

why amazons will be the first successful android tablet september pm pdt lets just get one thing out of the way amazons mysterious and still officially unannounced tablet is not an ipad killer the ipad can look forward to living a long and prosperous life at least in tech terms but today the first report came down of someone actually seeing and using the amazon tablet which is actually the next kindle according to techcrunchs mg seigler who was not allowed to photograph the device as zdnet's larry dignan points out the new kindle or kindles cnet also reported as

File29: Score: 71.6

want windows phone youre not alone september pm pdt microsoft has dismissed critics who say that windows phone is too little too late and that the company has missed its opportunity to be the kind of operating system powerhouse in smartphones that it has been in pcs au contraire say the redmondians we are still early in the game at least in the us the critics can point to early validation according to npds mobile phone track percent of the handsets sold to consumers in the second quarter of were smartphones android aided by a presence on all four major

File24: Score: 58.34

how desktop virtualization survived the recession september am pdt its fair to say desktop virtualization has had a checkered past as far back as vmworld had presentations on the topic of desktop virtualization also known as virtual desktop infrastructure by vmworld had developed a desktop virtualization track with a

Case 3: If the query terms I inputs didn't show in the same document, it will still return the rank of relevant documents where one of them may exist. For example, "had higher". Actually only "had" appears in file 24, but there are 5 occurrences of this word, so the file ranks first. But the score is not as high as previous case because the two words doesn't show in the same document.

results	
File24: Score: 74.95	how desktop virtualization survived the recession september am pdt its fair to say desktop virtualization has had a checkered past as far back as vmworld had presentations on the topic of desktop virtualization also known as virtual desktop infrastructure by vmworld had developed a desktop virtualization track with a number of deepdive technical sessions in june of idc issued a report on the promise of desktop virtualization touting how desktop virtualization can help rein in the costs of managing and maintaining pc infrastructures in february of crn reported gartners predictions that by between percent and percent of enterprise pcs would
File13: Score: 65.25	warming streams could be the end for salmon september am california chinook salmon face many threats but a new study predicts that climate change alone could finish them off as streams become too warm for spawning warming streams could spell the end of springrun chinook salmon in california by the end of the century according to a study by scientists at uc davis the stockholm environment institute and the national center for atmospheric research there are options for managing water resources to protect the salmon runs although they would impact hydroelectric power generation said lisa thompson director of the center
File12: Score: 63.98	resistance to antibiotics is ancient mcmaster study finds september am scientists were surprised at how fast bacteria developed resistance to the miracle antibiotic drugs when they were developed less than a century ago now scientists at mcmaster university have found that resistance has been around for at least years research

3. Varying k (number of singular values)

I use "happy summer" which shows the result as below. We can found that when k is large, the resulting score is not as much high as when it is small. Actually, the rank of relevant documents are not necessarily the same. By properly reducing the value of k, we can eliminate the noise data. After several tests, I found that when k is around 10, we will get more accurate result. For "graphene","kids","engineering", the documents listed will not rank high when k is large. But when k is around 10, the program will retrieve the documents listed.

"happy summer":

k=40	k=10	k=5
file21: 62.42	file21: 98.43	file26: 97.85
file13: 51.47	file29: 71.6	file25: 97.8
file18: 38.66	file24: 58.34	file21: 97.11

"engineering":

k=40	k=10	k=5
file37: 51.34	file32: 98.12	file32: 98.09
file31: 46.69	file30: 94.99	file33: 95.54
file12: 36.3	file33: 93.64	file31: 93.79

Actually, when k = 40, the document 33 ranks almost the last of the list.