CalPolyPomona

CS 4250 – Assignment #1

Maximum Points: 100 pts.

Bronco ID: 01 437 22 80
Last Name: Garcia
First Name: Jeremiah

Note 1: Your submission header must have the format as shown in the above-enclosed rounded rectangle.
Note 2: Homework is to be done individually. You may discuss the homework problems with your fellow students, but you are NOT allowed to copy – either in part or in whole – anyone else's answers.
Note 3: Your deliverable should be a .pdf file submitted through Gradescope until the deadline. Do not forget to assign a page to each of your answers when making a submission. In addition, source code (.py files) should be added to an online repository (e.g., github) to be downloaded and executed later.
Note 4: All submitted materials must be legible. Figures/diagrams must have good quality.
Note 5: Please use and check the Canvas discussion for further instructions, questions, answers, and hints. The bold words/sentences provide information for a complete or accurate answer.

1. [6 points]. Despite the current advances in the field, the primary focus of Information Retrieval is still on text and text documents. Based on this information, answer the questions below:

   a. [4 points]. Why is querying a database table easier compared to querying text documents? For full marks, **list** and **explain** at least **two factors** to elaborate your answer.
   b. [2 points]. Explain how **text** has been **used** by Information Retrieval researchers to compare multimedia documents and how this **scenario** is currently being **changed**.

a. Two reasons querying a database table is easier than querying a text document are, indexing and ranking. Indexing is the process of listing all of the terms in the documents you have and their frequency. Indexing makes it easier for your IR system to look for the terms in your query because it does not have to look through all of the documents each time a query is generated. Ranking is comparing the query to the index and deciding what documents are going to be the most relevant. We can do this Ranking because we have the index of terms to search.

b. Text has been an easy way for us to communicate with a computer and so it has been something we are able to process. By taking multimedia elements and describing them with text, such as transcriptions of audio recordings, we are able to make that media searchable. This is changing because modern search systems are learning to interpret other types of media, mainly images.

2. [10 points. 2 points each]. A search engine is the practical application of Information Retrieval techniques to large-scale text collections. **Explain** the scope of the different search engine applications and give **one practical example**.

   a. Web search engine.
   b. Vertical search engine.
   c. Enterprise search engine.
   d. Desktop search engine.
   e. Finally, **explain** how peer-to-peer search engines differ from the other previous types.

a.
Scope: Searches the entire web for any document that could be relevant to the users query. The only limit is the reach of the web.
Example: Google

b.
Scope: Similar to a web search, a vertical search covers the entire internet but it only indexes documents that are relevant to the vertical it was designed for.
Example: Google Scholar, only searches scholarly articles

c.
Scope: This is a search that is limited to an Enterprises information. For example searching for a company SOP.
Example: My company uses a program to store all of our reference material for our employees.

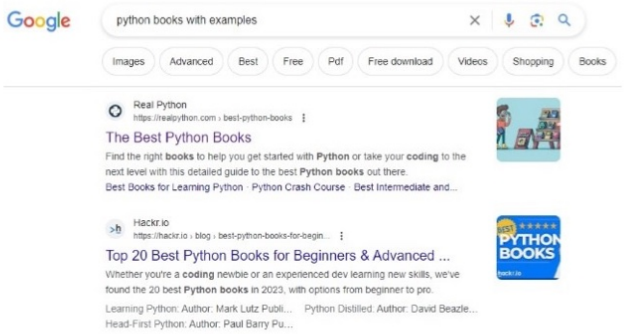
d.
Scope: This search is limited to the documents in your computers storage system.
Example: Your PC's file explorer

e.
Scope: This is limited to any computers on the network.
Example: A group of computers shared in an office that all can see each others files.
How it Differs: These are not limited by document type and they likely use a system that a lay person would have a hard time interacting with.

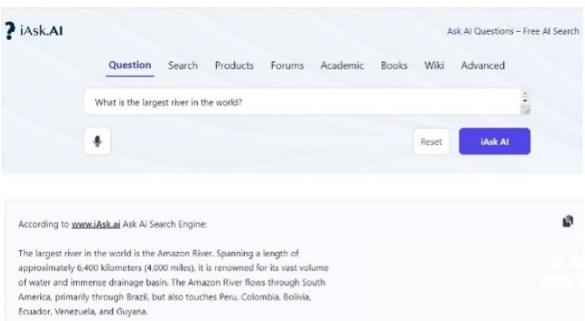
3. [8 points – 2 points each]. **Identify** and **explain** the following tasks that involve Information Retrieval.

a.

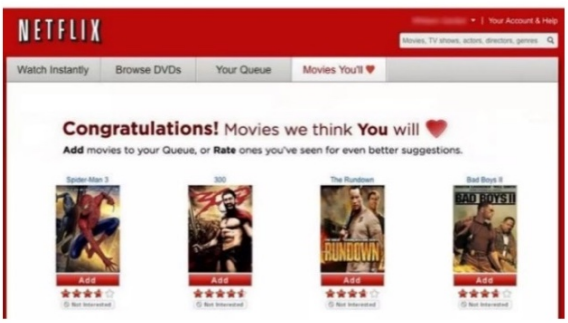
b.



c.



d.



a. Classification: Creating labels or in this case tags, for a document.

b. Ad hoc searching: Finding any relevant documents for a random search.

c. Question Answering: Creating a specific answer to a question query rather than simply providing relevant information.

d. Filtering: Deciding what document or movie in this example, is relevant to a user.

4. [8 points. 2 points each]. A retrieval model is a formal representation of the process of matching a query and a document, forming the basis of ranking algorithms that sort documents according to their relevance. Considering that **relevance** is one of the big issues for Information Retrieval research, answer the questions below.

   a. Explain why **topical relevance** and **user relevance** should be considered during search.
   b. Considering **only topical** but **not user relevance**, give a hypothetical example of a good search engine output based on a query. Provide the **user profile**, **query,** and **document returned**.
   c. Considering **only user** but **not topical relevance**, give a hypothetical example of a good search engine output based on a query. Provide the **user profile**, **query,** and **document returned**.
   d. Considering both **topical** and **user relevance**, give a hypothetical example of a good search engine output based on a query. Provide the **user profile**, **query,** and **document returned**.

a. Topical Relevance and User Relevance should be considered because they help confirm that the documents you are suggesting are not only relevant to the query that the user gave but what that user might mean more specifically than any other user with the same query.

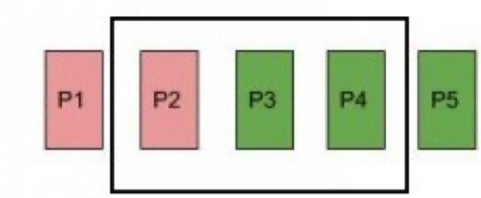
b.
User Profile: Dog Walker
Query: Python Course
Document Returned: Intro Course to Python Programming
Analysis: This is a good return because it is not dependent on the user profile at all. It simply considers the topic the user is querying.

c.
User Profile: Dog Walker
Query: Python Course
Document Returned: Pythons and other Snakes
Analysis: This is a good return because it is only dependent on the user profile. It does not consider the topic the user might be querying.

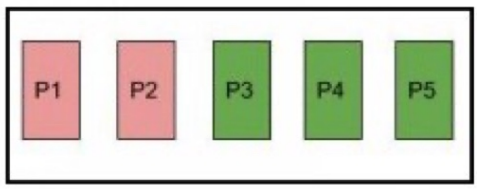
d.
User Profile: Dog Walker
Query: Python Course
Document Returned: Pythons and other Snakes
Analysis: This is a good return because it takes into account the user profile even though the topic might suggest the user is looking for programming both are considered and the SE landed on snakes.

5. [8 points. 2 points each]. Another core issue for information retrieval is evaluation. Two measures that have been extensively used for comparing search engines are precision and recall. Given the scenarios below, calculate the **precision** and **recall** of the corresponding search engines. Hint: green and red colors show the relevant and irrelevant documents respectively for a given query, and the black rectangles show the retrieved documents. Show your **math** (fraction and final value) for full marks.

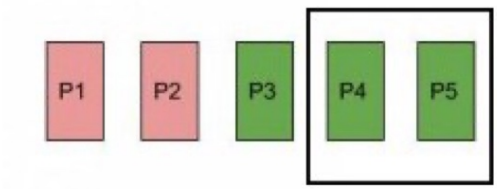
a.



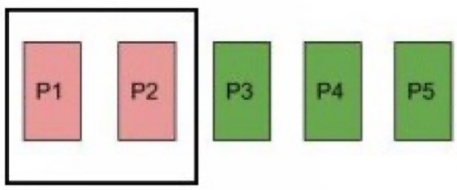
b.



c.



d.



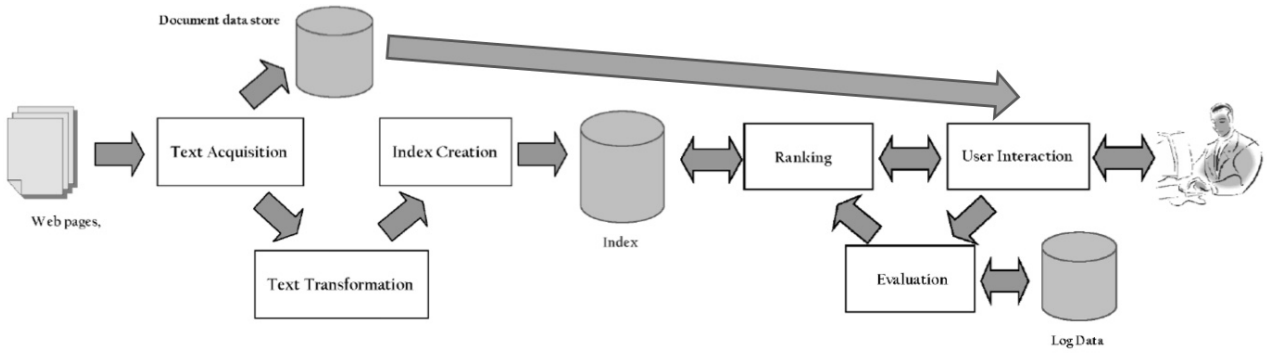
a.
Precision: 2/3 = 66%
Recall: 2/3 = 66%

b.
Precision: 2/5 = 40%
Recall: 3/3 = 100%

c.
Precision: 2/2 = 100%
Recall: 2/3 = 66%

d.
Precision: 0/2 = 0%
Recall: 0/3 = 0%

6. [20 points]. Assume that you work for company A which wants to implement a competitive Web search engine. Use the components in the image below to explain in one paragraph how those systems operate while executing the **indexing** and **query** processes to your supervisor. For full marks, **highlight** all components in your text. Hint: you can use word variations for each component.



The Web Search consists of two main processes, indexing and querying. While **indexing** the web pages or **documents** are **scanned to get their text**. Then the text is stored in the **Document Data Store** for use during querying. The text is also **transformed** to be organized into the **final index**. Then during **querying** the **user defines the query** and that query is used to **rank documents** based on the text in the **index**. That rank defines what data from the **Document Data Store** is shown to the user as the **result of their query**. While that all happened the system was **logging the users query** and **what they did** with the ranked information we gave them to **evaluate** the entire system and improve future performance.

7. [20 points]. Index term weights reflect the relative importance of words in documents and are used to compute scores for ranking. One of the most common types used in retrieval models is known as tf-idf. Derive the tf-idf document-term matrix according to the data below. Requirements: 1) you must conduct **stopword removal** and **stemming** before indexing the terms, 2) place the terms in the matrix following the sequence of their occurrences in the documents from d_1 to d_3, 3) show your **math** for full marks.

d_1 = "I love cats and cats".

d_2= "She loves her dog".

d_3= "They love their dogs and cat".

Stopwords: pronouns, conjunctions.

Documents after Stopword Removal and Stemming

d1 = "love cat cat"
d2 = "love dog"
d3 = "love dog cat"

Index
- love: 3
- cat: 3
- dog: 2

Love

$tf("love", d1) = \frac{1}{3} = 0.33$

$tf("love", d2) = \frac{1}{2} = 0.5$

$tf("love", d3) = \frac{1}{3} = 0.33$

$idf("love", D) = \log(\frac{3}{3}) = 0$

$tf\text{-}idf("love", d1, D) = 0.33 \times 0 = 0$

$tf\text{-}idf("love", d2, D) = 0.5 \times 0 = 0$

$tf\text{-}idf("love", d3, D) = 0.33 \times 0 = 0$

Cat

$tf("cat", d1) = \frac{2}{3} = 0.66$

$tf("cat", d2) = \frac{0}{2} = 0$

$tf("cat", d3) = \frac{1}{3} = 0.33$

$idf("cat", D) = \log(\frac{3}{2}) = 0.18$

$tf\text{-}idf("cat", d1, D) = 0.66 \times 0.18 = 0.1188$

$tf\text{-}idf("cat", d2, D) = 0 \times 0.18 = 0$

$tf\text{-}idf("cat", d3, D) = 0.33 \times 0.18 = 0.0594$

Dog

$tf("dog", d1) = \frac{0}{3} = 0$

$tf("dog", d2) = \frac{1}{2} = 0.5$

$tf("dog", d3) = \frac{1}{3} = 0.33$

$idf("dog", D) = \log\left(\frac{3}{2}\right) = 0.18$

$tf\text{-}idf("dog", d1, D) = 0 \quad \times 0.18 = 0$

$tf\text{-}idf("dog", d2, D) = 0.5 \quad \times 0.18 = 0.09$

$tf\text{-}idf("dog", d3, D) = 0.33 \quad \times 0.18 = 0.0594$

Document-Term Matrix	"love"	"cat"	"dog"
Document 1	0	0.1188	0
Document 2	0	0	0.09
Document 3	0	0.0594	0.0594

8. [20 points]. Complete the Python program (indexing.py) that will read the file collection.csv and output the tf-idf document-term matrix. Add the link to an online repository as the answer to this question.

https://github.com/Jman316b/CS4250.01-Assignment-1.git