Bronco ID: 015506777
Last Name: Lado
First Name: Martin

CS 4250 - Assignment #1

1a. Querying a database table is easier than querying text documents because the information stored in a text document is unstructured while the database does have structure. Another reason why is that a database system is made and optimized for querying structured data quickly and efficiently compared to a text document.

1b. Information Retrieval researchers have used text to compare multimedia documents by using the metadata associated with the pieces of media. Metadata properties such as titles, tags, creation date, and descriptions are pieces of text that can be compared to decide what information is relevant/important.

This situation is being changed by advancements being made in areas such as machine learning and information gathering. In addition to just using text, researchers can now focus on other forms of media such as pictures, videos, or music and gather useful information. They can also implement machine learning models to find complex patterns and create connections between all forms of media.

2.

a. Web search engine - search and index everything on the web. Ex: Browsing the web using Google Search

b. Vertical search engine - search focusing on a single topic. Ex: Using Yelp to browse information and reviews of businesses and restaurants.

c. Enterprise search engine - searching within a company. Ex: Using Microsoft search to look through your work files, find co-workers, and emails

d. Desktop search engine - search and index local files on a user's computer. Ex: Windows search to find a specific application.

e. Peer-to-peer search engines are different from the other types because they do not have a centralized server. They instead have users share files/resources with one another on a decentralized server.

3.

a. Classification, you must correctly identify the relevant labels for each document/file.
b. Ad-hoc search, finding relevant documents to the users text query.
c. Question answering, given the user's query you must provide a specific answer.
d. Filtering, based on the user's past preferences and watch history, netflix should filter movies that the user will most likely watch next.

4.

a. Both topical and user relevance should be considered during search to be able to provide the most relevant and useful documents to the user. If a user searched for “severe weather events”, you would want to provide information that is relevant to the user’s context based on their location and time frame.

b. User profile: A person interested in music
Query: Good piano music
Document Returned: A list of the most popular piano pieces with links to the music and the artist

c. User profile: A person interested in music, located in Los Angeles
Query: Good piano music
Document Returned: A list of the most popular music festivals or Jazz clubs near Los Angeles with reviews and information about upcoming events that may interest the user based on their preferences

d. User profile: A person interested in music, located in Los Angeles
Query: Good piano music
Document Returned: List of most popular pianists and their music catalog as well as information on their career and if they are holding any events near Los Angeles.

5.

a. Precision: ⅔ = 67%
Recall: ⅔ = 67%

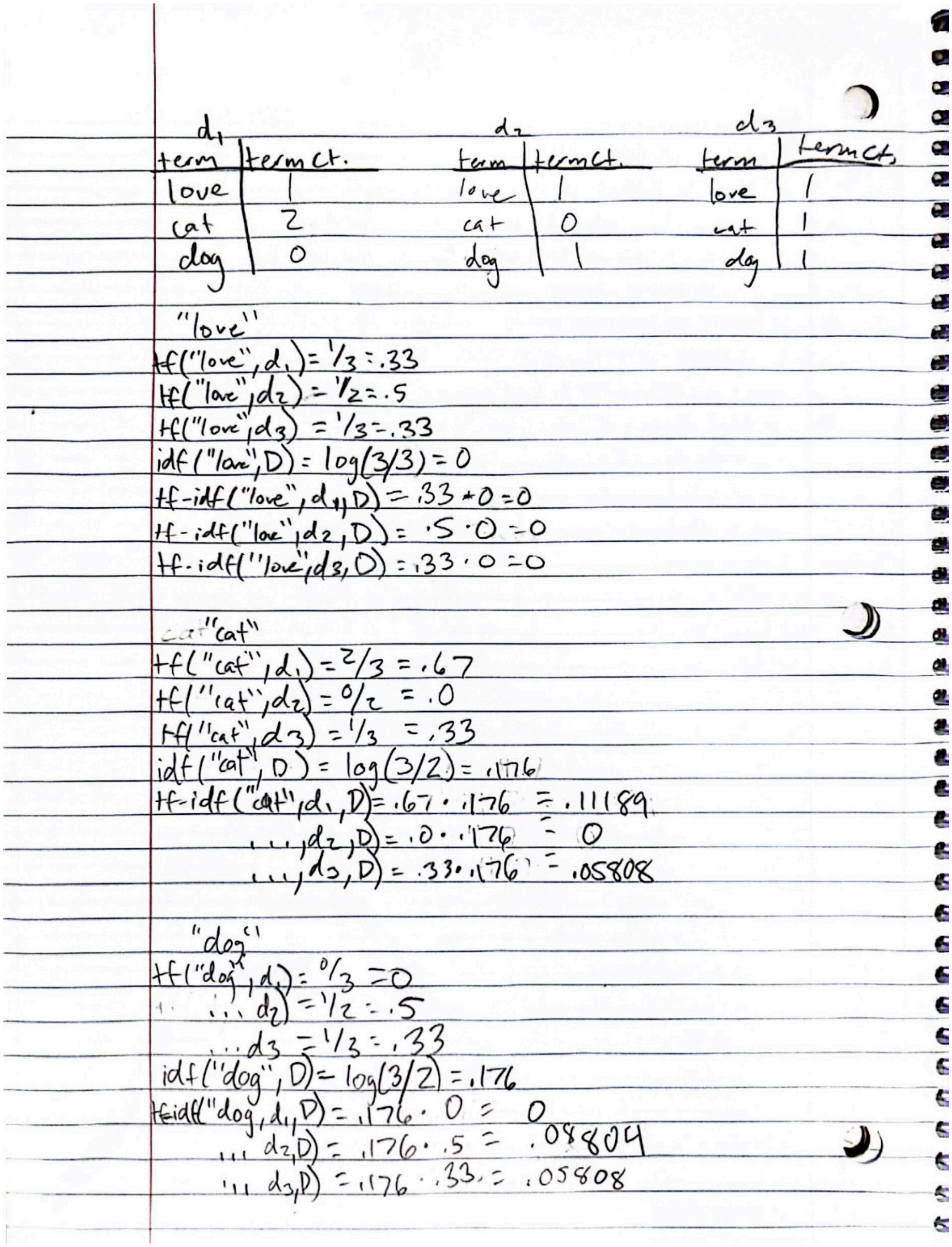
b. Precision: ⅗ = 60%
Recall: 3/3 = 100%

c. Precision: 2/2 = 100%
Recall: ⅔ = 67%

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

6. The first process of a web search engine is the **Text Acquisition**, which is when documents (or **web pages**) are identified and stored for indexing in the **document data store**. Usually, web crawlers are used to save the metadata. Second, **text transformation** transforms the documents into index terms/features so that **index creation** can be done and this information will be stored into the **index**. These two processes create a data structure that enables fast searching. On the other side, **user interaction** takes care of the user queries and then displays the results. During this process **ranking** of documents is done based on the query and indexes to give the most relevant documents **back to the user**. Finally, **evaluation** of the effectiveness and efficiency is taken into account and stored into the **log data** for the future.

7.



d_1

term	term ct.
love	1
cat	2
dog	0

d_2

term	term ct.
love	1
cat	0
dog	1

d_3

term	term ct,
love	1
cat	1
dog	1

"love"

tf("love", d_1) = 1/3 = .33

tf("love", d_2) = 1/2 = .5

tf("love", d_3) = 1/3 = .33

idf("love", D) = log(3/3) = 0

tf-idf("love", d_1, D) = .33 * 0 = 0

tf-idf("love", d_2, D) = .5 · 0 = 0

tf-idf("love", d_3, D) = .33 · 0 = 0

~~cat~~"cat"

tf("cat", d_1) = 2/3 = .67

tf("cat", d_2) = 0/2 = .0

tf("cat", d_3) = 1/3 = .33

idf("cat", D) = log(3/2) = .176

tf-idf("cat", d_1, D) = .67 · .176 = .11189

..., d_2, D) = .0 · .176 = 0

..., d_3, D) = .33 · .176 = .05808

"dog"

tf("dog", d_1) = 0/3 = 0

..., d_2) = 1/2 = .5

...d_3 = 1/3 = .33

idf("dog", D) = log(3/2) = .176

tfidf("dog, d_1, D) = .176 · 0 = 0

..., d_2, D) = .176 · .5 = .08804

..., d_3, D) = .176 · .33 = .05808

Document Term Matrix	"Love"	"Cat"	"Dog"
Doc 1	0	0.11189	0
Doc 2	0	0	0.8804
Doc 3	0	0.05808	0.05808

8. GitHub Repo