# CSE508 Information Retrieval

## Winter 2024

## Assignment-1

Konam Akhil Vamshi

2020513

# Question 1:

**Approach and methodology:**

1. Applied the below-mentioned pre-processing steps on the given dataset to narrow down the size of indexing in the coming parts.
    - Lowercasing text
    - Tokenization
    - Removing stopwords
    - Removing punctuations
    - Removing blank spaces

2. Preferred spaCy over NLTK for pre-processing as spaCy is more robust and contains a broader set of stopwords. This helps to further reduce the vocabulary size, reducing the size of indexes. This may help in faster retrieval of queries.

**Outputs:**

Text 1
Loving these vintage springs on my vintage strat. They have a good tension and great stability. If you are floating your bridge and want the most out of your springs than these are the way to go.
---
Preprocessed_Text 1
loving vintage springs vintage strat good tension great stability floating bridge want springs way
---

Text 2
Works great as a guitar bench mat. Not rugged enough for abuse but if you take care of it, it will take care of you. Makes organization of workspace much easier because screws won't roll around. Color is good too.
---
Preprocessed_Text 2
works great guitar bench mat rugged abuse care care makes organization workspace easier screws wo roll color good
---

Text 3
We use these for everything from our acoustic bass down to our ukuleles. I know there is a smaller model available for ukes, violins, etc.; we haven't yet ordered those, but these will work on smaller instruments if one doesn't extend the feet to their maximum width. They're gentle on the instruments, and the grippy material keeps them secure.

The greatest benefit has been when writing music at the computer and needing to set a guitar down to use the keyboard/mouse - just easier for me than a hanging stand.

We have several and gave one to a friend for Christmas as well. I've used mine on stage, and it folds up small enough to fit right in my gig bag.
---
Preprocessed_Text 3
use acoustic bass ukuleles know smaller model available ukes violins etc ordered work smaller instruments extend feet maximum width gentle instruments grippy material keeps secure greatest benefit writing music computer needing set guitar use keyboard mouse easier hanging stand gave friend christmas stage folds small fit right gig bag
---

Text 4
Great price and good quality.  It didn't quite match the radius of my sound hole but it was close enough.
---
Preprocessed_Text 4
great price good quality match radius sound hole close
---

Text 5
I bought this bass to split time as my primary bass with my Dean Edge. This might be winning me over. The bass boost is outstanding. The active pickups really allow you to adjust to the sound you want. I recommend this for anyone. If you're a beginner  like I was not too long ago, it's an excellent bass to start with. If you're on tour and/or music is making you money, this bass will be beatiful on stage. The color is a bit darker than in the picture. But, all around, this is a great buy.
---
Preprocessed_Text 5
bought bass split time primary bass dean edge winning bass boost outstanding active pickups allow adjust sound want recommend beginner like long ago excellent bass start tour and/or music making money bass beatiful stage color bit darker picture great buy
---

# Question 2:

**Methodology**

- **Preprocessing with spaCy:** Lowercasing text, tokenisation, removal of stopwords, punctuation and spaces using spaCy.
- **Inverted Index Creation:** Python function inv_index that iterates over preprocessed text files in a directory, creating a mapping of unique words to the files they appear in.
- **Index Serialization:** Using pickle to serialise the inverted index to save it and use it later.
- **Boolean Query Processing:** process_query function to support AND, OR, AND NOT, OR NOT operations by manipulating sets of documents based on the presence or absence of query terms in the inverted index.
- **Query and Operation Input:** Taking query and operations input from the user.
- **Preprocessing User Queries:** Preprocessing the above input using spaCy.
- **Result Display:** Displaying the results of retrieved documents as per instructions.

**Sample input and output:**

**Input:**
1
Car bag in a canister
OR, AND NOT

**Output:**
Query 1: car OR bag AND NOT canister
Number of documents retrieved for query 1: 31
Names of the documents retrieved for query 1: file863.txt, file363.txt, file174.txt, file930.txt, file746.txt, file981.txt, file699.txt, file956.txt, file892.txt, file118.txt, file686.txt, file698.txt, file73.txt, file860.txt, file797.txt, file864.txt, file264.txt, file780.txt, file886.txt, file166.txt, file466.txt, file313.txt, file459.txt, file942.txt, file404.txt, file665.txt, file682.txt, file542.txt, file573.txt, file3.txt, file738.txt

# Question 3:

**Methodology:**

- **Positional Index Creation:** Created a positional index mapping each unique word to its positions within documents.
- **Serialisation with Pickle:** Using pickle to serialise the inverted index to save it and use it later.
- **Handling Phrase Queries:** ‘process’ function to process input phrase queries by checking for the exact sequence of words in the positional index.
- **Query and Operation Input:** Taking query and operations input from the user.
- **Preprocessing User Queries:** Preprocessing the above input using spaCy.
- **Result Display:** Displaying the results of retrieved documents as per instructions.

**Sample input and output:**

**Input:**
1
fun play

**Output:**
Number of documents retrieved for query 1 using positional index: 3
Names of documents retrieved for query 1 using positional index: file279.txt, file6.txt, file854.txt