Information Retrieval
CS F469
First Semester, 2019-20

Assignment 1

Music Lyrics Search Engine

Design Document

Team #4

Ekanshi Agrawal	**Sandesh Thakar**	**Vrutik Halani**	**Vivek Soni**
2017A7PS0233H	2017A7PS0181H	2017A7PS1732H	2017A7PS0173H

The Idea

We often find ourselves in a situation where we listen to a song in say, a public place like a mall, or on the television, and it remains stuck in our minds, and yet we do not know the title or the artist behind it.
The system we present in this assignment is meant to help its users find the title, artist and the entire lyrics, by just entering a few words from the lyrics from the song.

The Model

The system is presented as a tkinter GUI application made in Python.

Storage Structure:

We have acquired a dataset of words and lyrics, which are processed and stored for convenient retrieval. The dataset of the lyrics is read and indexed. Each distinct word (term) in the dataset is indexed against IDs in the dictionary. The dictionary is made accessible by both IDs as well as words.
Songs are collected from the dataset and returned for further processing. Details of each song are saved in a dictionary according to the different zones (Artist name and title) and processed and stored in a data structure (an array of key-value pairs, where the key is the zone) that enables access to the songs and their corresponding details using the songIDs.

User Queries:

The user enters a list of words into the system, that he/she knows the song contains or wants the result to contain. Queries are fetched from the tkinter text field and processed by replacing characters, making spelling corrections and replacing the words by their IDs. Processing involves replacement of shortened forms (‘ve for have, ‘ll for will, etc.) with their full forms, and splitting

of the resultant query into words. Stemming is done using the Porter Stemmer. Words in the query that are not available in the dictionary are checked for spelling errors/closest words.

Tolerant Retrieval:

Edit distances between misspelled query words and dictionary terms are used to act as spelling correction measures. The problem of calculating edit distances between two words is treated as a dynamic programming problem, to find the lowest number of operations (replace, remove, or insert) required to change the query term into a word that exists in the dictionary.
Since our corpus is not too large, we run the computation of edit distance for each query term that does not occur in the dictionary, with each term in the dictionary and note the term with the minimum edit distance as the autocorrected spelling to be used for further processes.

Ranking:

Tf-idf scores of each word in the lyrics of each song are calculated by iterating through them and stored by songID. Cosine Similarity scores are used to rank songs for a query. Cosine score of every song is calculated with respect to the query, and songs are sorted according to the calculated scores in descending order. The top 15 songs are returned as results to the query, in a text area in the tkinter GUI.

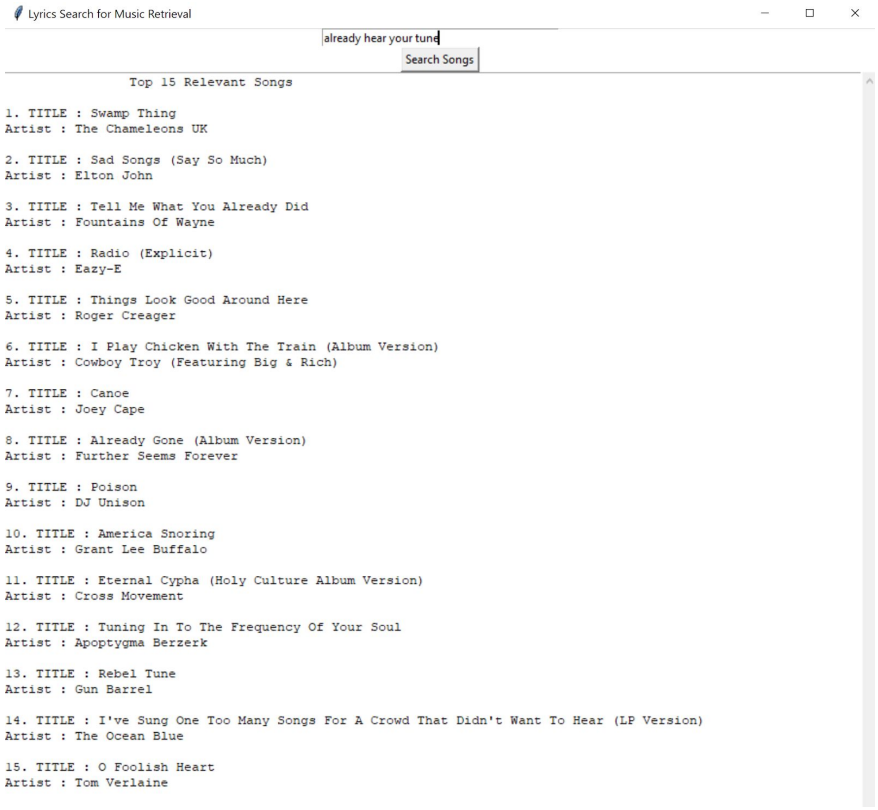
Interdependencies Between Processes:

The processing of user entered queries is dependent on stemming of the query terms, which is to be followed by the autocorrection of the terms (edit-distance calculation). Edit distance should not be calculated before stemming/lemmatization as it could reduce recall or precision (as un-stemmed words may not be present in the dictionary and have a short edit distance with a word unrelated to it, causing inaccuracy).
The calculation of Cosine Similarity Scores has a precondition of having tf-idf scores of every term stored. The Cosine Similarity scores between the terms in the dictionary and the query

terms should be calculated only after the query has been processed completely (stemmed, lemmatized and autocorrected). Not doing this will result in faulty cosine scores and inaccurate results.

Outputs :

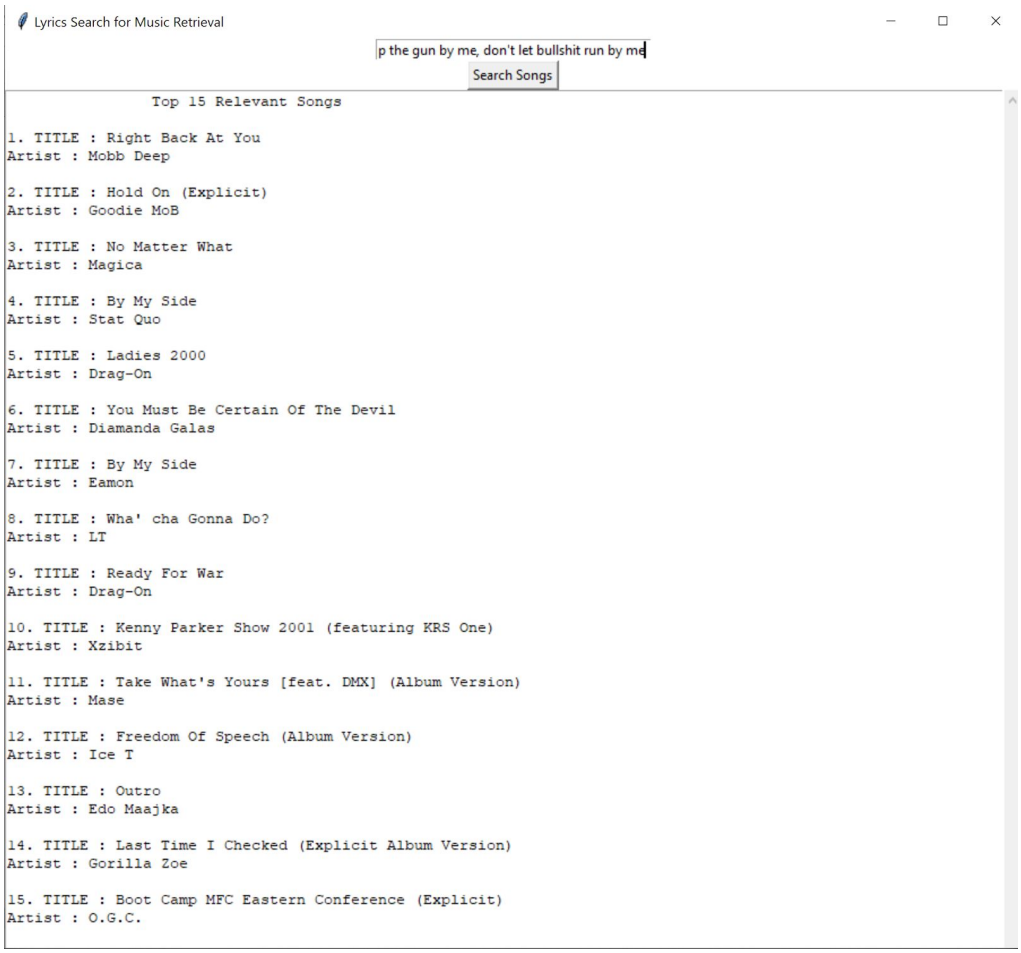
1. Query : “Already hear your tune “ . → Swamp Things



Start Time : 2019-10-12 01:44:22.574852

End Time : 2019-10-12 01:44:24.724134

2. Query : “Keep the gun by me, don't let bullshit run by me” → Ladies 2000



Start Time : 2019-10-12 01:51:32.294693

End Time : 2019-10-12 01:51:36.330117