Evaluation of your data collection & information retrieval system

submission date: 30.04.2025

Student ID	50251141
Student name	Hamidreza Rahimian

Contents

1 Introduction

1.1 Motivation

Information Retrieval (IR) is a fundamental aspect of modern search engines and data retrieval systems. In this project, we focus on implementing Latent Semantic Indexing in IR system to search within a custom dataset of song lyrics. The dataset consists of **82 text files containing lyrics from some of my favorite rap artists, including Eminem, Dr. Dre, Kanye West, Drake, and Tupac**.

1.2 Goal

Traditional **keyword based** search methods often struggle with these characteristics because they are only searching for exact specific words and not similar ones so it makes it difficult to find songs based on themes or concepts rather than exact word matches.

In this project (Latent Semantic Indexing in the IR system), we will:

1. Implement and **compare** two information retrieval methods (TF-IDF and LSI)
2. Evaluate their **effectiveness** for concept-based searching in rap lyrics.
3. Discuss the findings based on the results obtained.

Through this project, we aim to understand the **advantages and limitations of LSI** search in the context of music lyrics retrieval. Additionally, we will reflect on the effectiveness of it and compare the search with and without LSI .

1.3 Approach

In this project we will use a python script using some libraries to run the search through the songs that we have in our data base. First time we will try to search the queries without and second time with the LSI search. The we compare the search results to see how LSI helped and we will rank the songs based on LSI and normal search to see if there were changes.

2 Methodology

In this project, a Python-based LSI search system was developed to process queries and return documents containing exact matches.

The search process works as follows:

1. **Preprocessing the Lyrics Dataset:**
   - Each song's lyrics are stored in a **separate text file**.
2. **LSI Query Execution:**
   - The system scans all documents and retrieves only those that satisfy the LSI.

Then, I built two different search methods to compare:

TF-IDF (Term Frequency-Inverse Document Frequency): This is a basic method that most search engines use. It looks at how often words appear in songs and ranks them based on that thing. It's good at finding exact matches but struggles with understanding meaning.

LSI (Latent Semantic Indexing): This more advanced method tries to understand the relationships between words. It can tell that "money," "cash," and "wealth" are related concepts, allowing it to find songs that are about the same topic even if they use different words.

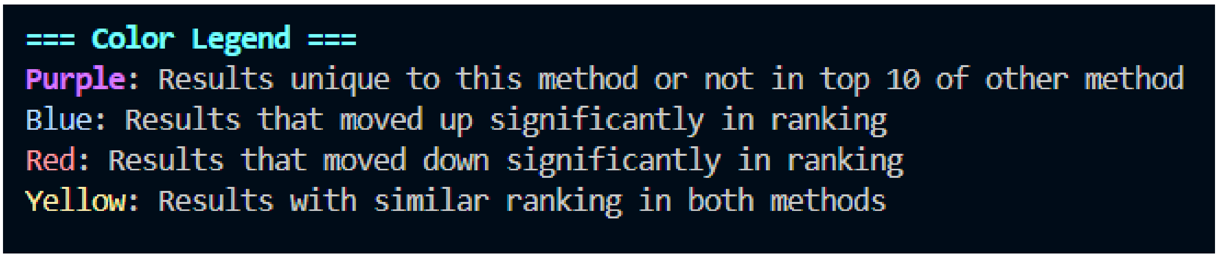
3. **Comparing search result with and without LSI:**
   - After executing a query, we will identify **which documents are truly relevant**. LSI helps to find relevant documents , even if the exact word that we are searching for is not existing in that document.

To make it easy to see the difference between these methods, I added color-coding to the results. Purple shows songs that only appear in one method's results, blue shows songs that moved up in ranking, and red shows songs that moved down.

3 Search Queries

To test the system and show how it works, I ran several different types of queries and analyzed the results blow them. This section covers what I found when searching for different concepts and how the TF-IDF and LSI methods compared.

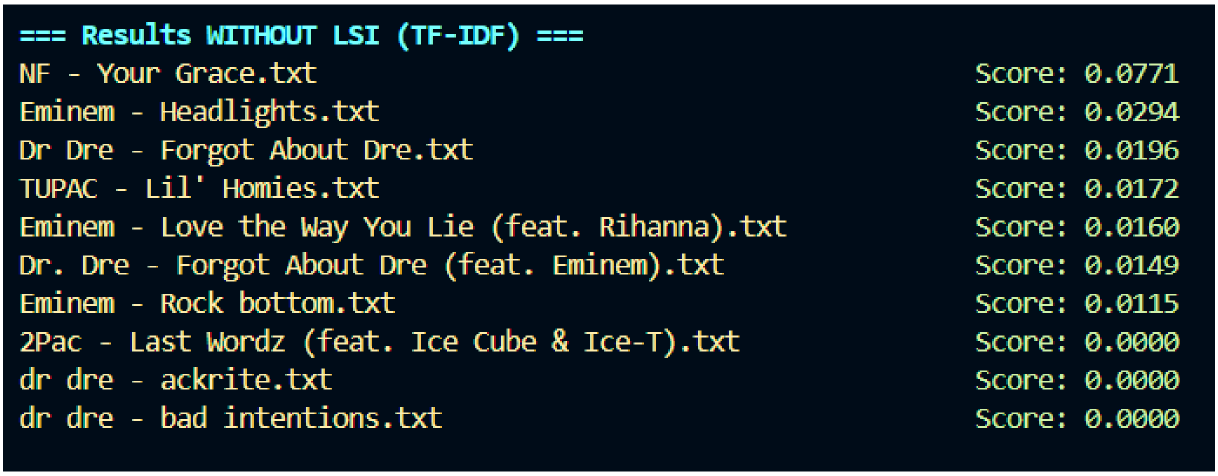
To make the comparison easier I added colors to print output , and here is the instructions:



```
=== Color Legend ===
Purple: Results unique to this method or not in top 10 of other method
Blue: Results that moved up significantly in ranking
Red: Results that moved down significantly in ranking
Yellow: Results with similar ranking in both methods
```

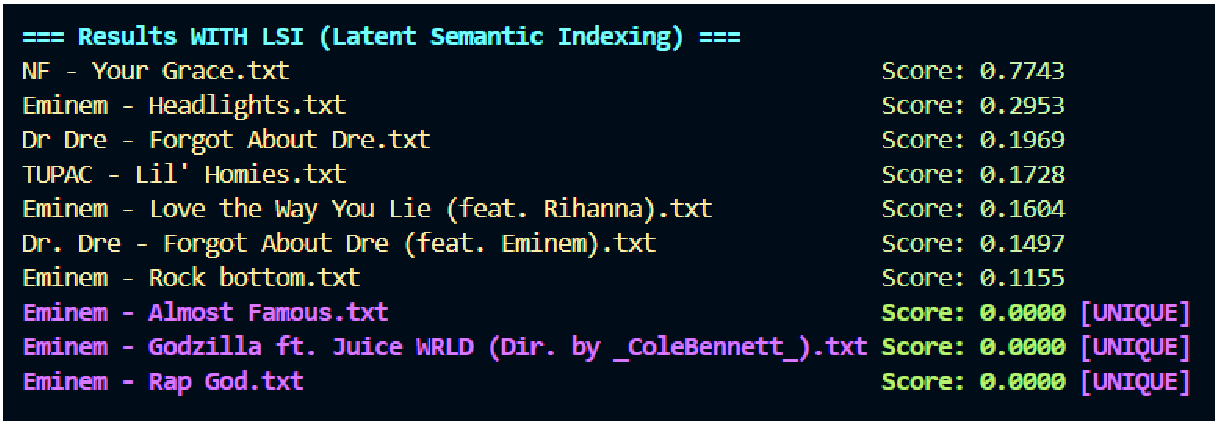
3.1 Query 1: " sadness beauty broken "

TF-IDF Results:



```
=== Results WITHOUT LSI (TF-IDF) ===
NF - Your Grace.txt                                          Score: 0.0771
Eminem - Headlights.txt                                      Score: 0.0294
Dr Dre - Forgot About Dre.txt                                Score: 0.0196
TUPAC - Lil' Homies.txt                                      Score: 0.0172
Eminem - Love the Way You Lie (feat. Rihanna).txt            Score: 0.0160
Dr. Dre - Forgot About Dre (feat. Eminem).txt                Score: 0.0149
Eminem - Rock bottom.txt                                     Score: 0.0115
2Pac - Last Wordz (feat. Ice Cube & Ice-T).txt               Score: 0.0000
dr dre - ackrite.txt                                         Score: 0.0000
dr dre - bad intentions.txt                                  Score: 0.0000
```

LSI Results:



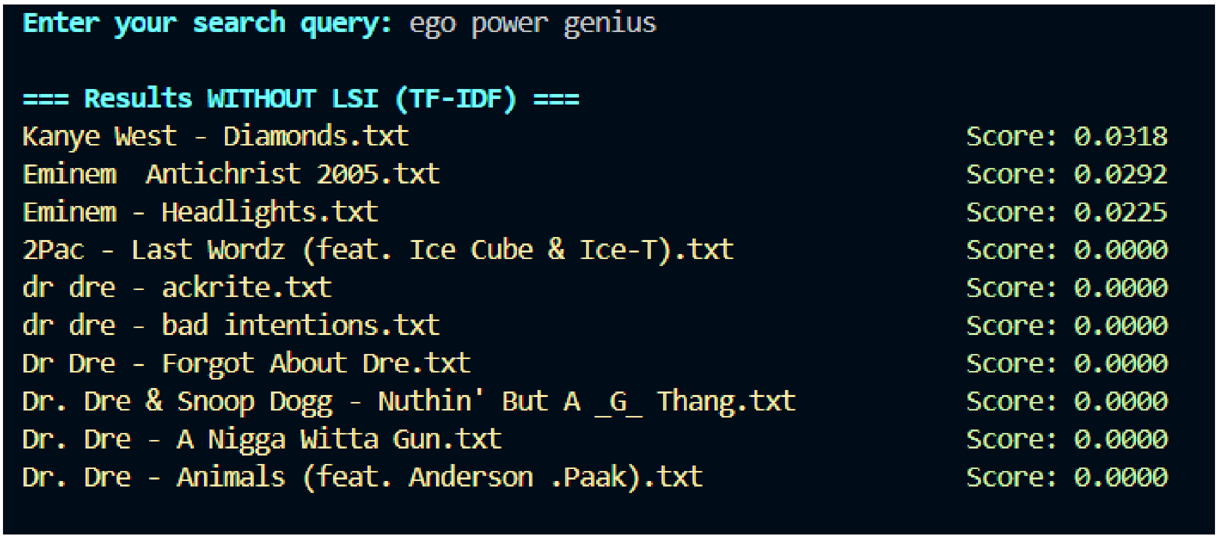
```
=== Results WITH LSI (Latent Semantic Indexing) ===
NF - Your Grace.txt                                       Score: 0.7743
Eminem - Headlights.txt                                   Score: 0.2953
Dr Dre - Forgot About Dre.txt                             Score: 0.1969
TUPAC - Lil' Homies.txt                                   Score: 0.1728
Eminem - Love the Way You Lie (feat. Rihanna).txt         Score: 0.1604
Dr. Dre - Forgot About Dre (feat. Eminem).txt             Score: 0.1497
Eminem - Rock bottom.txt                                  Score: 0.1155
Eminem - Almost Famous.txt                                Score: 0.0000 [UNIQUE]
Eminem - Godzilla ft. Juice WRLD (Dir. by _ColeBennett_).txt Score: 0.0000 [UNIQUE]
Eminem - Rap God.txt                                      Score: 0.0000 [UNIQUE]
```

Observations:

- *It's clear that the result is very different , with TF-IDF search I showed the first 10 relevant song , as it's shown in first section , however after using LSI , the result is going to be different , we even have 3 new song in the search result , more over the score of the songs that were already in top 10 changed now .* those 3 songs were not visible in *TF-IDF search and now with LSI we know that these 3 songs are also relevant .*

3.2 Query 2: " ego power genius "

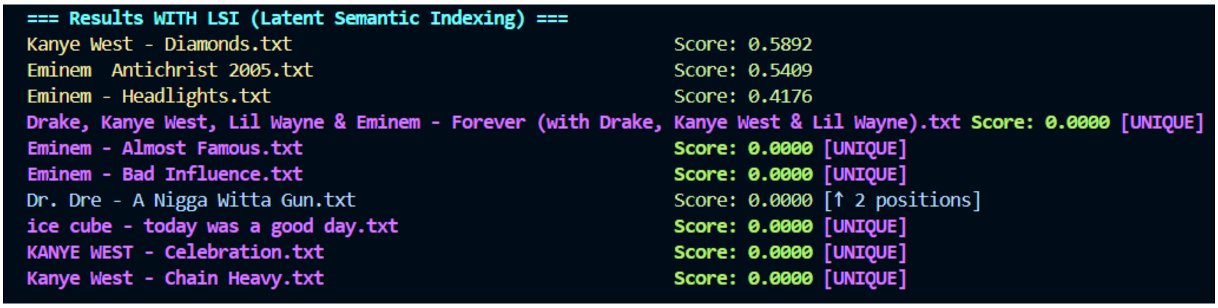
TF-IDF Results:



```
Enter your search query: ego power genius

=== Results WITHOUT LSI (TF-IDF) ===
Kanye West - Diamonds.txt                                    Score: 0.0318
Eminem  Antichrist 2005.txt                                  Score: 0.0292
Eminem - Headlights.txt                                      Score: 0.0225
2Pac - Last Wordz (feat. Ice Cube & Ice-T).txt               Score: 0.0000
dr dre - ackrite.txt                                         Score: 0.0000
dr dre - bad intentions.txt                                  Score: 0.0000
Dr Dre - Forgot About Dre.txt                                Score: 0.0000
Dr. Dre & Snoop Dogg - Nuthin' But A _G_ Thang.txt           Score: 0.0000
Dr. Dre - A Nigga Witta Gun.txt                              Score: 0.0000
Dr. Dre - Animals (feat. Anderson .Paak).txt                 Score: 0.0000
```

LSI Results:



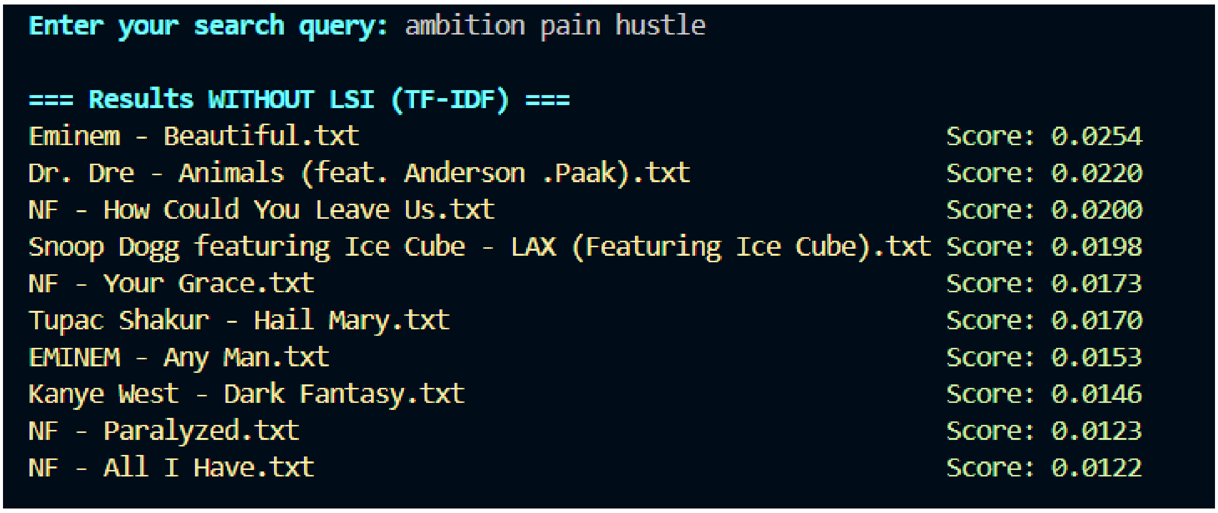
```
=== Results WITH LSI (Latent Semantic Indexing) ===
Kanye West - Diamonds.txt                                Score: 0.5892
Eminem  Antichrist 2005.txt                              Score: 0.5409
Eminem - Headlights.txt                                  Score: 0.4176
Drake, Kanye West, Lil Wayne & Eminem - Forever (with Drake, Kanye West & Lil Wayne).txt Score: 0.0000 [UNIQUE]
Eminem - Almost Famous.txt                               Score: 0.0000 [UNIQUE]
Eminem - Bad Influence.txt                               Score: 0.0000 [UNIQUE]
Dr. Dre - A Nigga Witta Gun.txt                          Score: 0.0000 [↑ 2 positions]
ice cube - today was a good day.txt                      Score: 0.0000 [UNIQUE]
KANYE WEST - Celebration.txt                             Score: 0.0000 [UNIQUE]
Kanye West - Chain Heavy.txt                             Score: 0.0000 [UNIQUE]
```

Observations:

- *In this example the difference between TF-IDF and LSI is even bigger and more visible , after running LSI search we will see that the top 10 search result are clearly different . 6 of the top 10 are new result and one of the songs that has been showed in blue has changed the ranking in from 9th relevant to 6th relevant .*

3.3 Query 3: " ambition pain hustle "

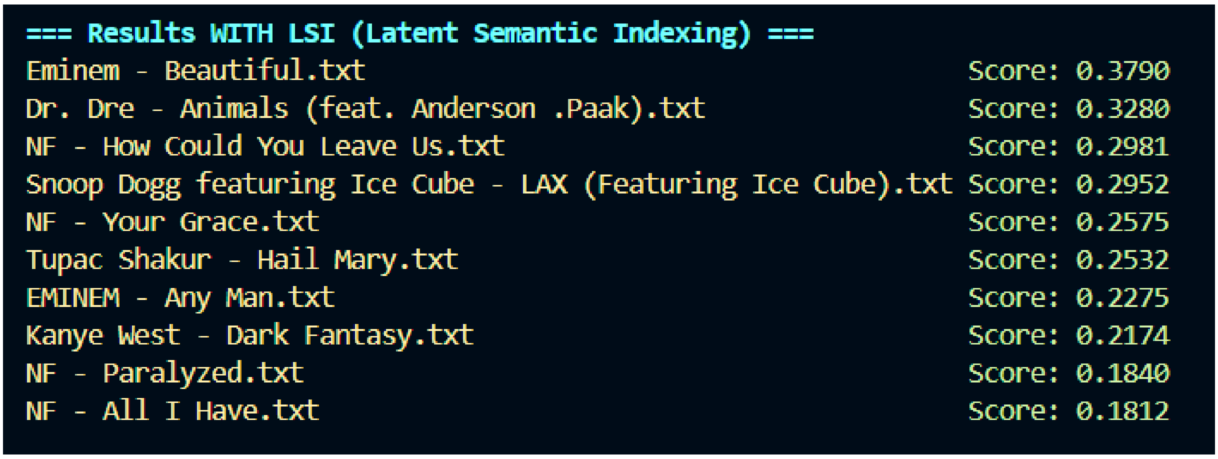
TF-IDF Results:



```
Enter your search query: ambition pain hustle

=== Results WITHOUT LSI (TF-IDF) ===
Eminem - Beautiful.txt                                       Score: 0.0254
Dr. Dre - Animals (feat. Anderson .Paak).txt                 Score: 0.0220
NF - How Could You Leave Us.txt                              Score: 0.0200
Snoop Dogg featuring Ice Cube - LAX (Featuring Ice Cube).txt Score: 0.0198
NF - Your Grace.txt                                          Score: 0.0173
Tupac Shakur - Hail Mary.txt                                 Score: 0.0170
EMINEM - Any Man.txt                                         Score: 0.0153
Kanye West - Dark Fantasy.txt                                Score: 0.0146
NF - Paralyzed.txt                                           Score: 0.0123
NF - All I Have.txt                                          Score: 0.0122
```

LSI Results:



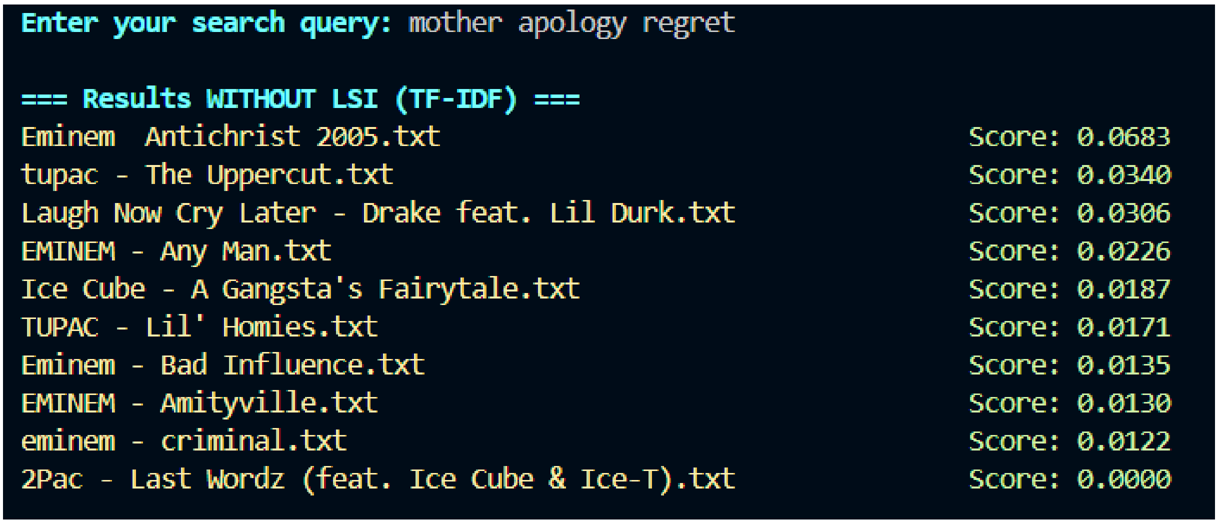
```
=== Results WITH LSI (Latent Semantic Indexing) ===
Eminem - Beautiful.txt                                       Score: 0.3790
Dr. Dre - Animals (feat. Anderson .Paak).txt                 Score: 0.3280
NF - How Could You Leave Us.txt                              Score: 0.2981
Snoop Dogg featuring Ice Cube - LAX (Featuring Ice Cube).txt Score: 0.2952
NF - Your Grace.txt                                          Score: 0.2575
Tupac Shakur - Hail Mary.txt                                 Score: 0.2532
EMINEM - Any Man.txt                                         Score: 0.2275
Kanye West - Dark Fantasy.txt                                Score: 0.2174
NF - Paralyzed.txt                                           Score: 0.1840
NF - All I Have.txt                                          Score: 0.1812
```

Observations:

- *Here we will see no new result that is unique in LSI, of course this case can happen , it simply means that the top 10 songs in TF-IDF were are top 10 considering the LSI . however, the scores will be different in LSI .*

3.4 Query 4: " mother apology regret "

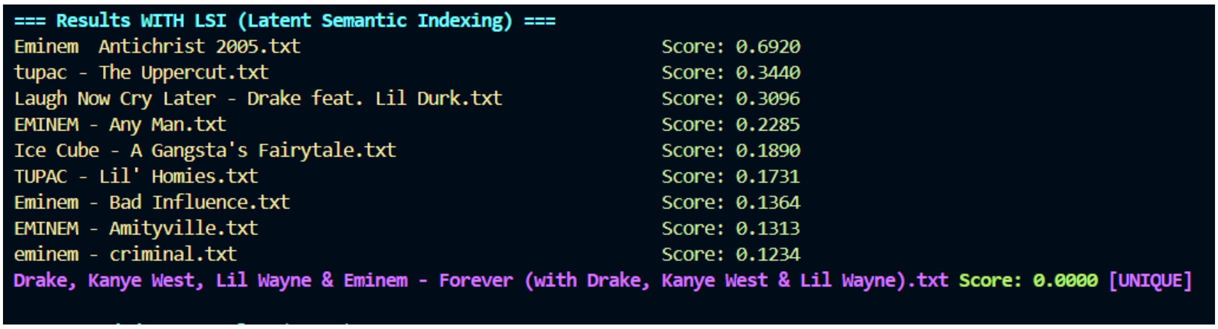
TF-IDF Results:



```
Enter your search query: mother apology regret

=== Results WITHOUT LSI (TF-IDF) ===
Eminem  Antichrist 2005.txt                            Score: 0.0683
tupac - The Uppercut.txt                               Score: 0.0340
Laugh Now Cry Later - Drake feat. Lil Durk.txt         Score: 0.0306
EMINEM - Any Man.txt                                   Score: 0.0226
Ice Cube - A Gangsta's Fairytale.txt                   Score: 0.0187
TUPAC - Lil' Homies.txt                                Score: 0.0171
Eminem - Bad Influence.txt                             Score: 0.0135
EMINEM - Amityville.txt                                Score: 0.0130
eminem - criminal.txt                                  Score: 0.0122
2Pac - Last Wordz (feat. Ice Cube & Ice-T).txt         Score: 0.0000
```

LSI Results:



```
=== Results WITH LSI (Latent Semantic Indexing) ===
Eminem  Antichrist 2005.txt                          Score: 0.6920
tupac - The Uppercut.txt                             Score: 0.3440
Laugh Now Cry Later - Drake feat. Lil Durk.txt       Score: 0.3096
EMINEM - Any Man.txt                                 Score: 0.2285
Ice Cube - A Gangsta's Fairytale.txt                 Score: 0.1890
TUPAC - Lil' Homies.txt                              Score: 0.1731
Eminem - Bad Influence.txt                           Score: 0.1364
EMINEM - Amityville.txt                              Score: 0.1313
eminem - criminal.txt                                Score: 0.1234
Drake, Kanye West, Lil Wayne & Eminem - Forever (with Drake, Kanye West & Lil Wayne).txt Score: 0.0000 [UNIQUE]
```

Observations:

- *Here the difference between them is not that crazy , considering LSI there were only one new song in top 10 search result , but still count. So now its kind of clear to us that the change between TF-IDF and LSI is really based and depends on the queries. It can be really different , exact same or sometimes just some small changes.*

4 Conclusion

After running TF-IDF and LSI through 5 different queries and comparing result we reached some important conclusions:

1. **TF-IDF works best for**:
   - Direct searches where you know the exact words
   - Searches for uncommon or unique terms
   - When you need simple, straightforward results
   - When computing resources are limited
2. **LSI excels at**:
   - Finding songs based on concepts or themes
   - Discovering connections between related terms
   - Emotional or abstract searches
   - When you want depth over exact matching

It’s obvious that in most cases LSI did a better job , because it not only search about a word , but also try to find the same concept . for example, maybe there is no word “money” in a text file how ever some other word like “cash” or “expensive” or “fame” are there , these words are also relevant to money , because they are all in the concept of having money , so LSI can perfume generally better , how ever if our goal is to exactly target a word the TF-IDF is of course a better option for search because there the only important thing is if the exact word was used or not , even if the meaning of that word can be a little different than what we expect. Other things that was clear is that most of the time after using LSI the result will clearly change , however sometimes the search result can be still the same . basically its very depended of what we are searching and our search queries. Basically the biggest different is that after running LSI we will have some new results , some files are now added in the search result , this files didn’t contain the exact word that we were search for word , but they add synonyms of the search queries or other similar and related word , so LSI consider that and decided that this file is also related to the topic we are searching for .

Thanks for reading my report and for your time.