U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

VerseVault
Your Key to Lyrics, Your Door to Music

# (PRI - M3)

André Lima - 202008169
Guilherme Almeida - 202006137
Jorge Sousa - 202006140
José Castro - 202006963

Group 75

# Introduction - **What is it?** (Recap)

- **Lyric-based** Search Engine for **finding songs**
- Aggregates **various metadata** regarding musical content
- Aims to:
  - Allow **text-based querying**
  - Observe a specific song's **composition**

    (both **lyrically** and **structurally**)

# Introduction - **Why do it?** (Recap)

Lyrical and structural musical data is **vast but mostly unexplored**

**Centralized and rich data sources** for our data collection

In its final stage, it will be a **fun and interactive way** to interact and **perceive music production and its history**

Group 75

# Query Structure

Assuming ‘query’ is the user search input, the queries are structured as following:

```
{!parent which="doc_type:track" score=total}
{!edismax qf="title^0.5
content^4 content_phonetic^2" pf="content^10"}
(query~3)
```

This way, we apply a field boosting in the title, content and content_phonetic fields, a phrase boost in the content and a slope of 3 in the query input.

# Information Need 1

**Search scenario:** I want to find a track talking about something inside a particular structural section.

**Information Need:** Find songs that talk about playing the guitar in the chorus.

**User Input:** play guitar

**Query:**

```
{!parent which="doc_type:track" score=total}

{!edismax qf="title^0.5

content^4 content_phonetic^2" pf="content^10"}

((play guitar)~3)
```

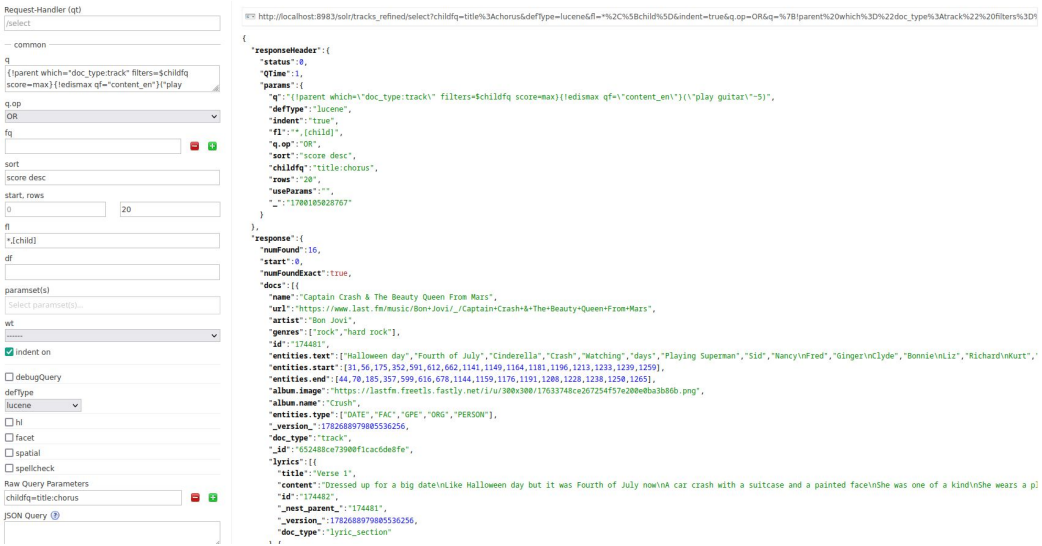


Fig 1: Information Need 1 - Query result in solr (improved query)

Group 75

# Information Need 2

**Search scenario:** I want to find a song talking about a given topic.

**Information Need:** Find songs that talk about slavery in Africa.

**User Input:** slavery africa

**Query:**

```
{!parent which="doc_type:track" score=total}

{!edismax qf="title^0.5

content^4 content_phonetic^2" pf="content^10"}

((slavery apartheid africa)~3)
```

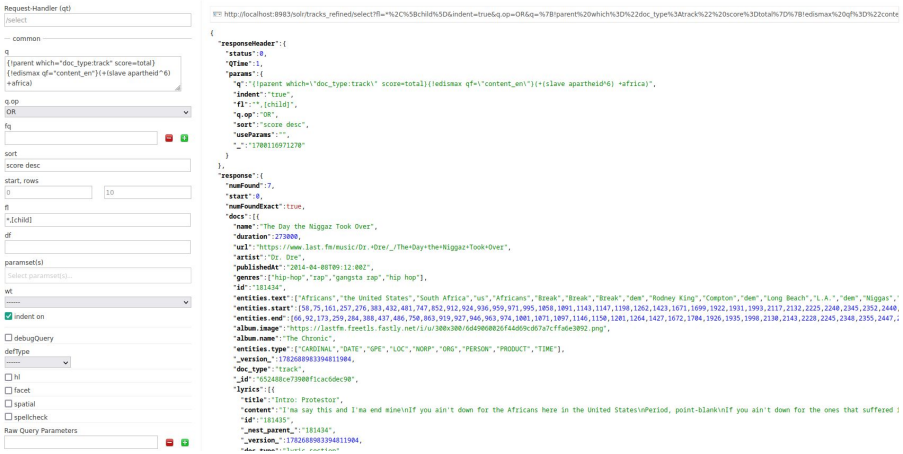


Fig 2: Information Need 2 - Query result in solr (improved query)

Group 75

# Information Need 3

**Search scenario:** I want to find a song of a given genre that talks about a specific topic.

**Information Need:** Find rap songs that talk about immigration.

**User Input:** foreigners immigration

**Query:**

```
{!parent which="doc_type:track" score=total}

{!edismax qf="title^0.5 content^4

content_phonetic^2" pf="content^10"} ((foreigners

immigration)~3)
```

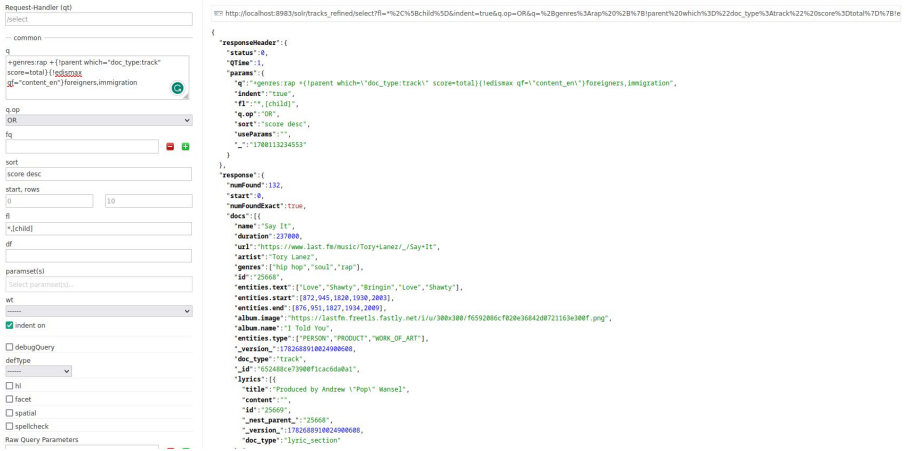


Fig 3: Information Need 3 - Query result in solr (improved query)

# **Semantic Search** vs **Synonym Usage**

lyrical content is **disorganized**, **unstructured**, **unconventional** and very **misleading...**

...how to stop context from being **lost in translation**?

Two possible solutions...
**Semantic Search** and **Synonym Usage**

Group 75

# **Semantic Search** vs **Synonym Usage**

## Semantic Search:

- Sentence-transformers
- *all-MiniLM-L6-v2* Model

→ **content_vector** containing embeddings (song lyrics)

- Query using *DenseVectorSearch* (KNN)>

→ **target_vector** containing embeddings (query)

## Synonym Usage:

- Solr's Synonym filter (query-time only)

Group 75

# Evaluation - Initial Schema

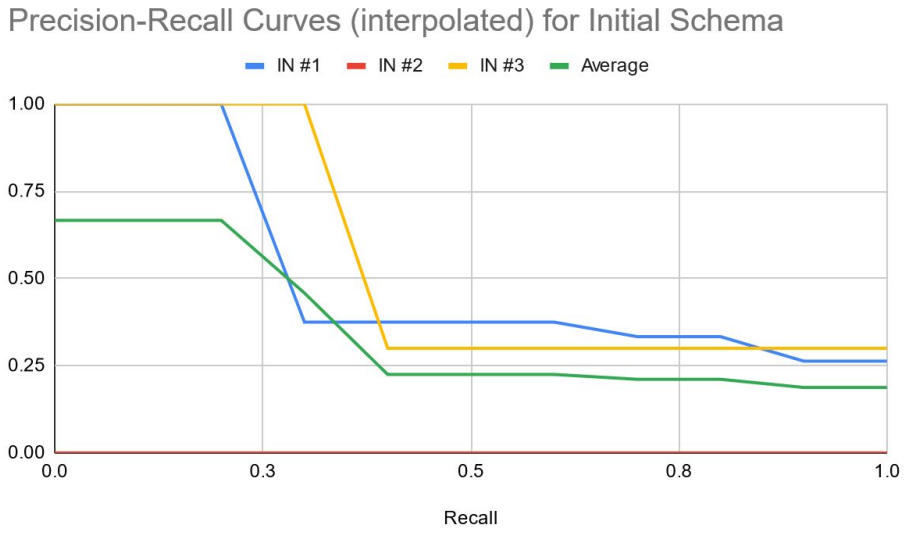


Fig 4: Interpolated P-R Curve for Initial Schema

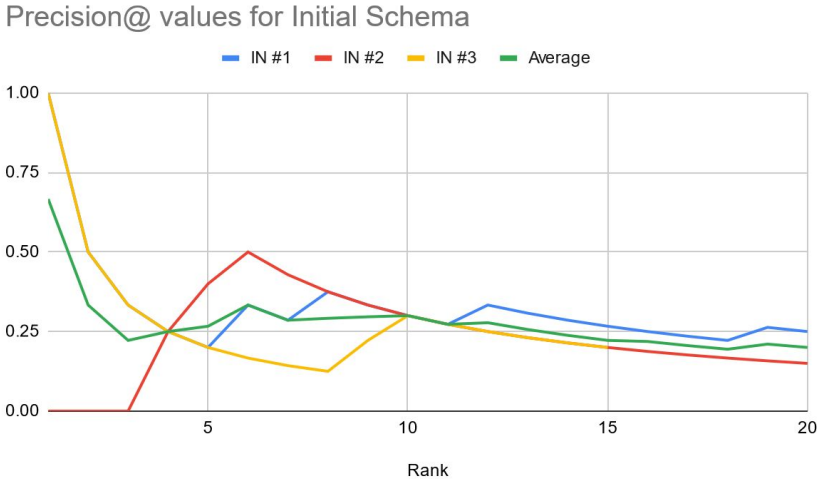


Fig 5: Precision@ values for Initial Schema

Group 75

MAP: 45%

# Evaluation - Refined Schema

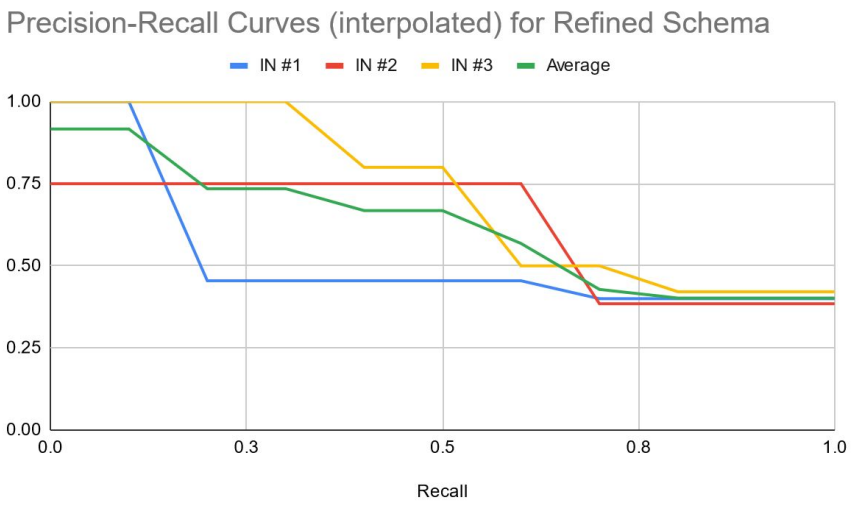


Fig 6: Interpolated P-R Curve for Refined Schema

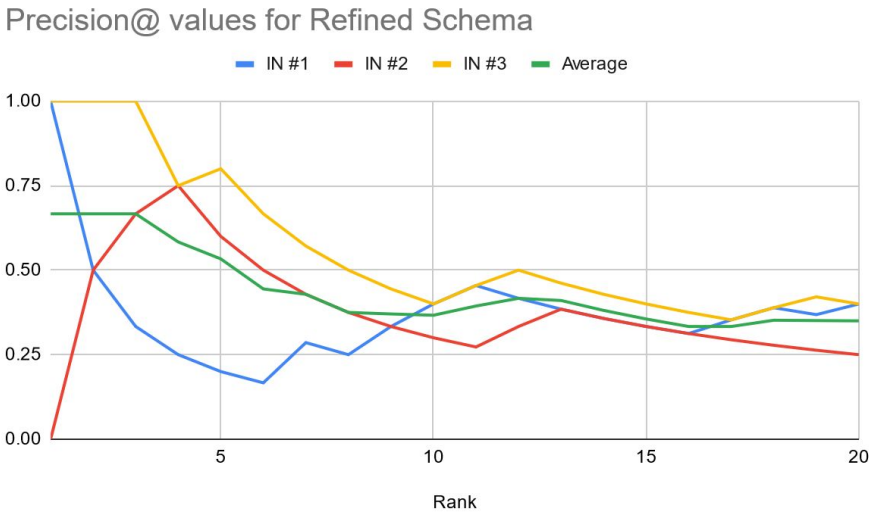


Fig 7: Precision@ values for Refined Schema

Group 75

MAP: 56%

# Evaluation - Semantic Schema

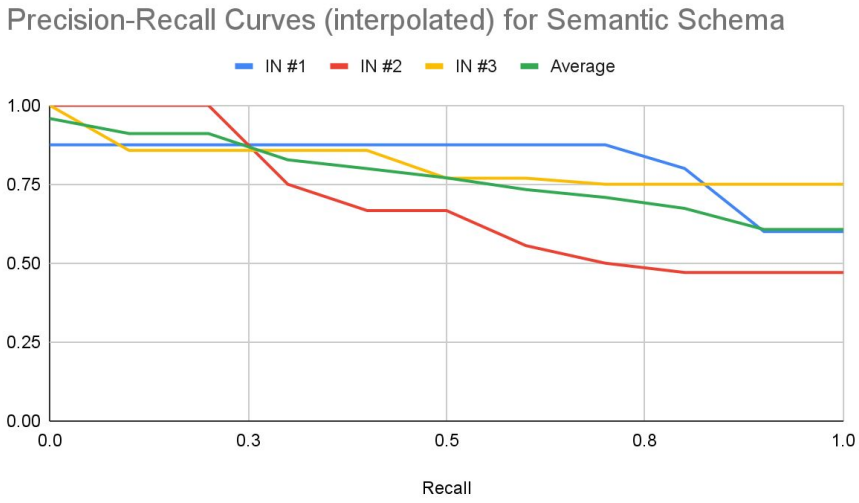


Fig 8: Interpolated P-R Curve for Semantic Schema

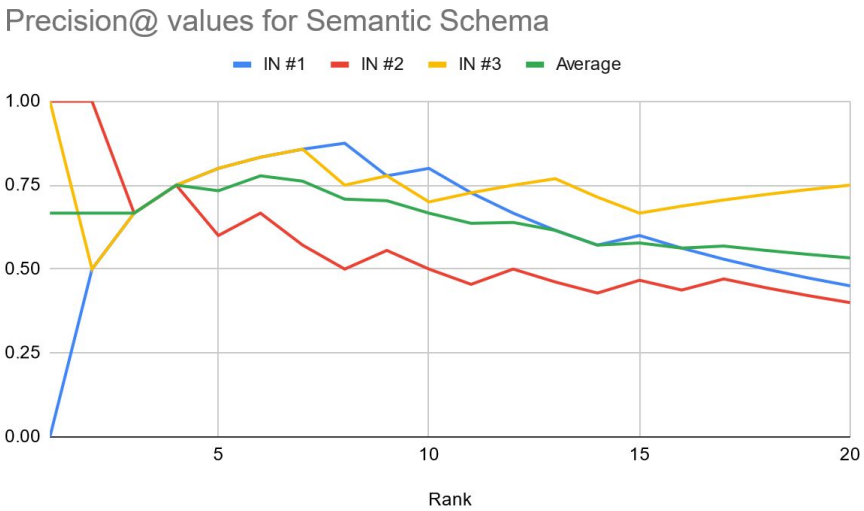


Fig 9: Precision@ values for Semantic Schema

Group 75

MAP: 73%

# Evaluation - Synonym Schema

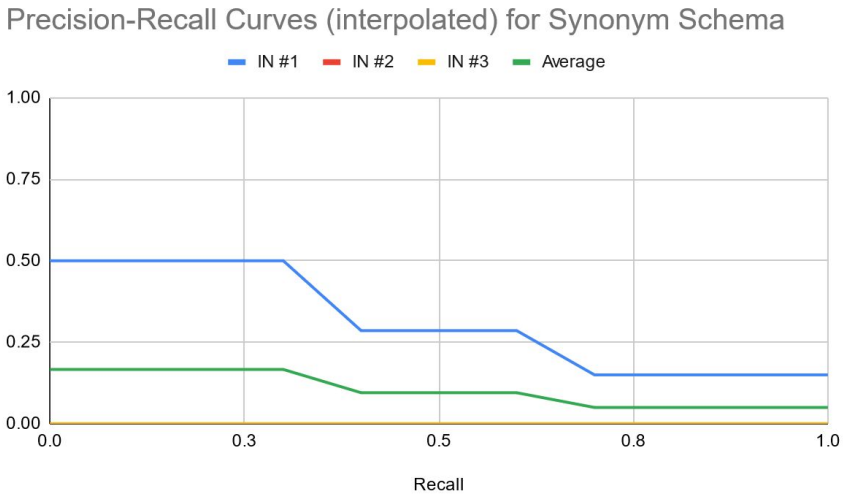


Fig 10: Interpolated P-R Curve for Synonym Schema

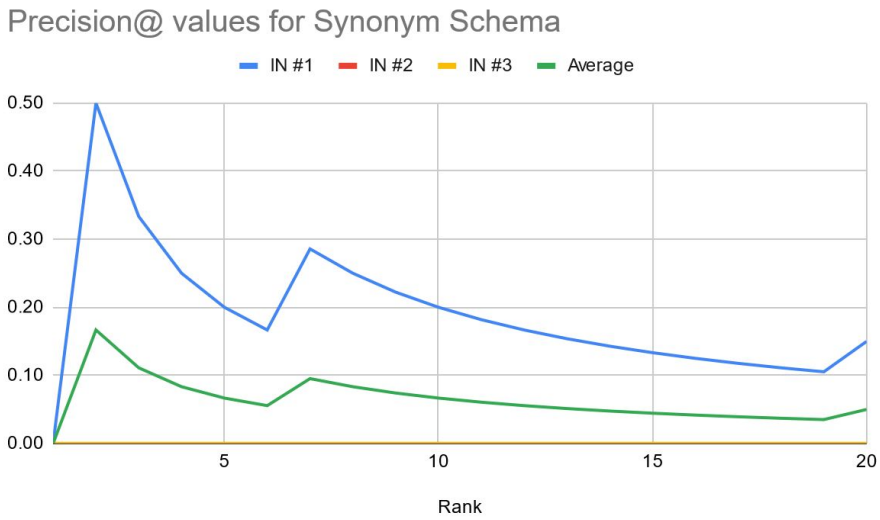


Fig 11: Precision@ values for Synonym Schema

Group 75

MAP: 10.3%

# Evaluation - Comparing Schemas

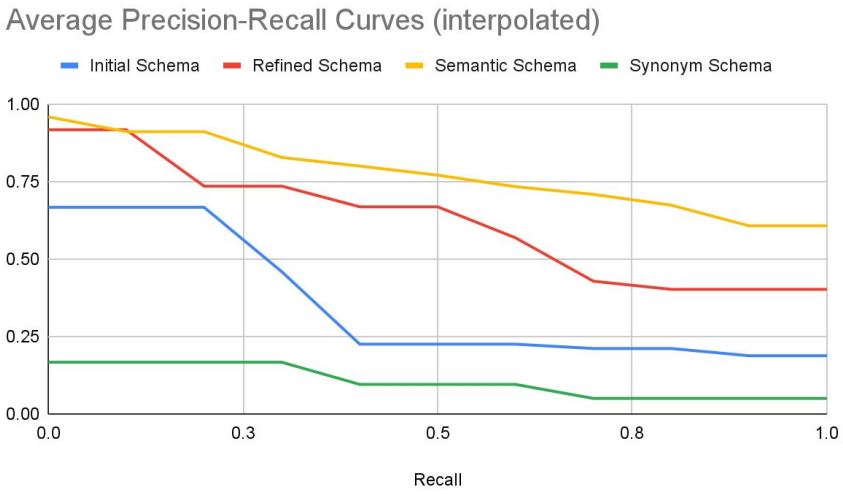


Fig 12: Average Interpolated P-R Curves for all schemas

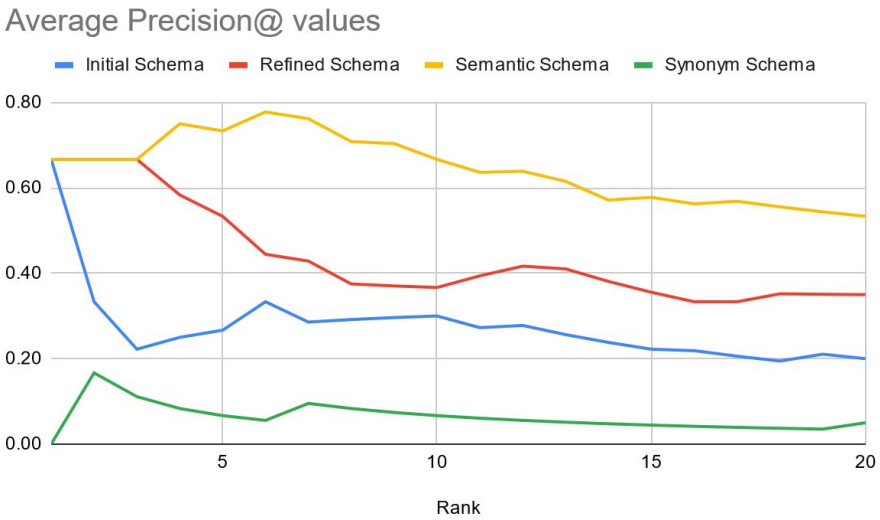


Fig 13: Average Precision@ values for Synonym Schema

Group 75

# Frontend

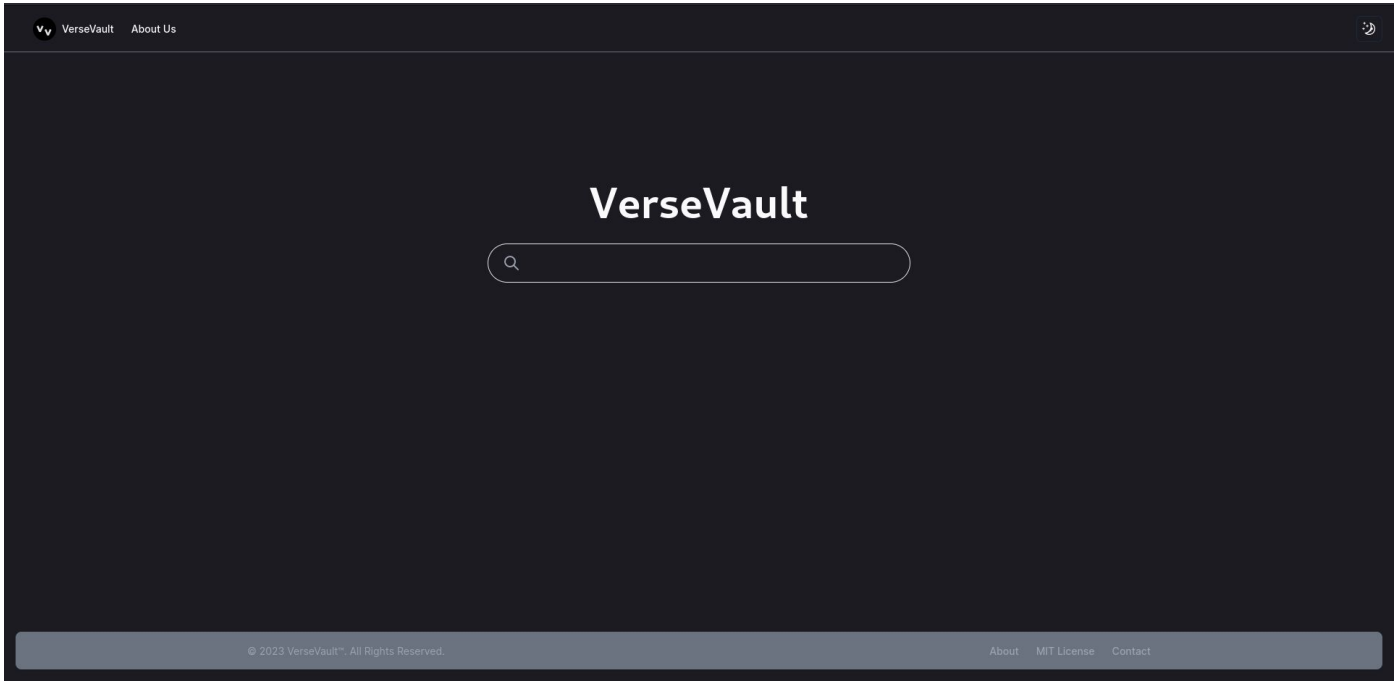


Fig 14: VerseVault's Home Page

# Frontend

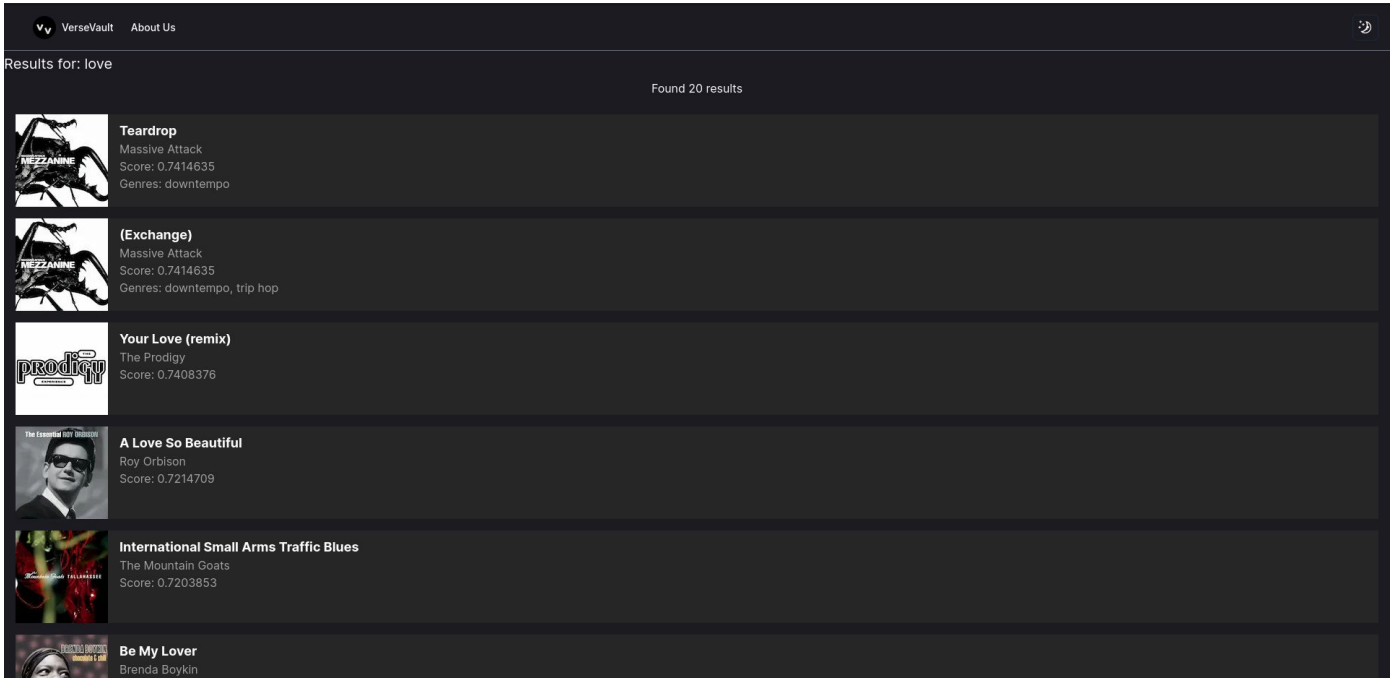


Fig 15: VerseVault's Results Page

# Frontend

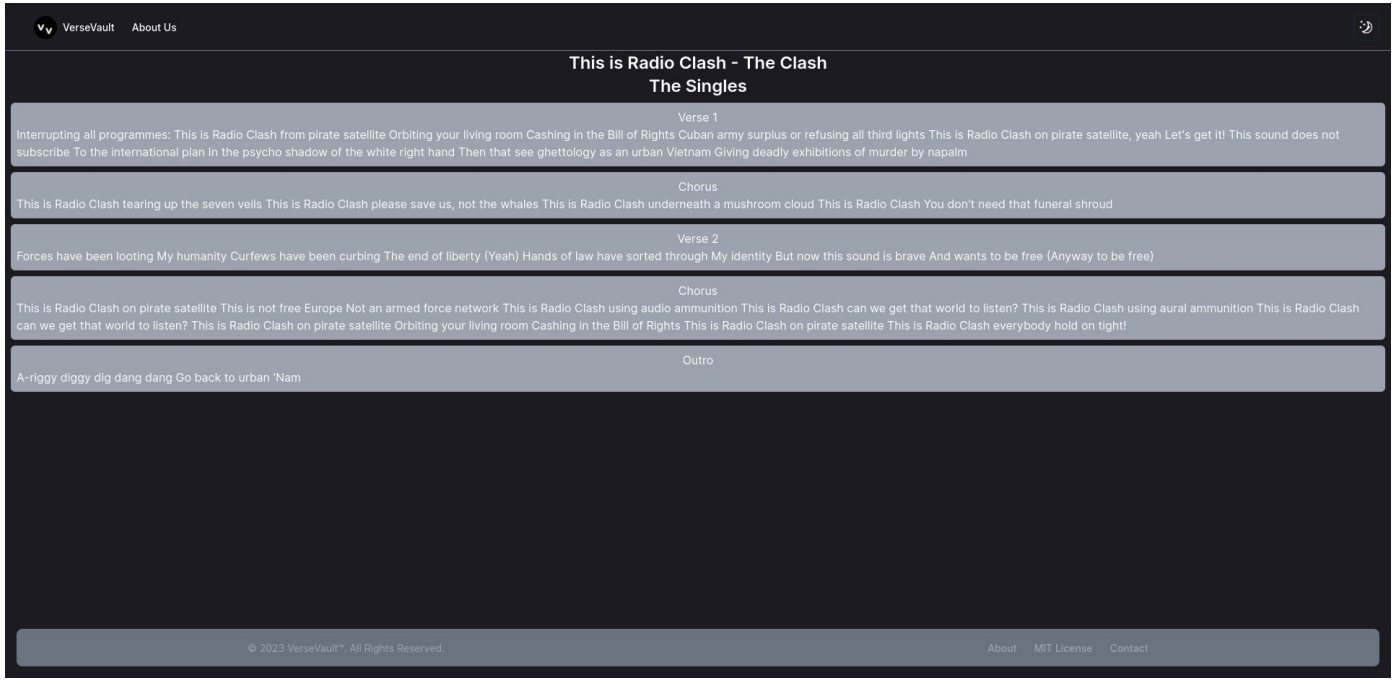


Fig 16: VerseVault's Track Page

# Frontend

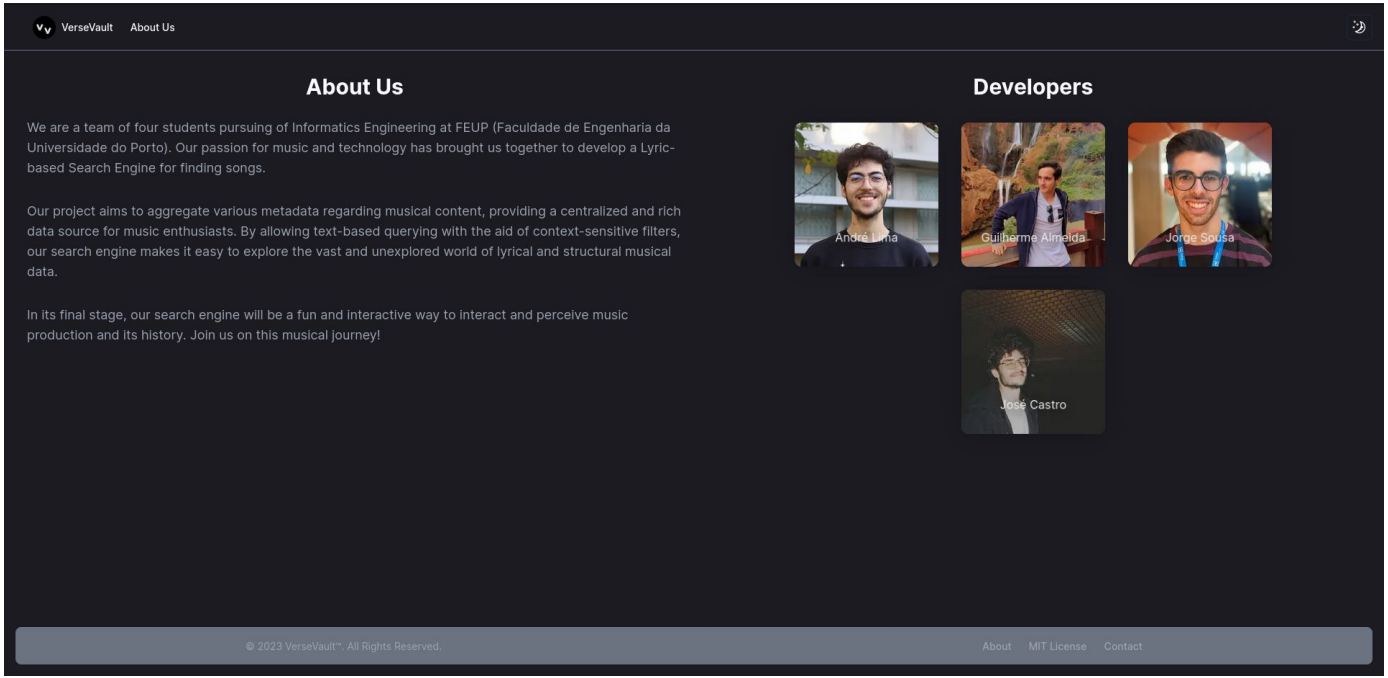


Fig 17: VerseVault's About Page

# **Conclusions** and **Future Work**

- The refined schema provided better results compared to the initial schema, as expected.
- The use of synonyms revealed some limitations due to the broad matching of terms.
- As a signal, we could prioritize tracks that have more lyrical sections.

# Questions

Thank you for your time!

Group 75