

Report: Search a Movie

s0239369-Ioannis Tzortzakos

- **The problem**

This [project](#) is a search engine that searches a database with movies, integrated into a flask web application. You can select to search by Title, by Description, or by both. You can write the search term in the text input area and filter the results by genre. The results are shown in the web page.

- **Overview**

The front-end is built with HTML, CSS and Javascript. The back-end is implemented by Python library [flask](#). The IR part is implemented with the Python library [Whoosh](#). The dataset is the [kaggle dataset](#) being preprocessed by some custom Java scripts. Swoosh library is utilizes the Okapi BM25F ranking function.

- **Implementation**

The front-end is implemented with vanilla HTML, CSS, and Javascript. I use Ajax to communicate between the front and back. The back-end is implemented with flask. Depending on the action of the user, it calls accordingly the correct Python script.

`app.py`

This file runs other scripts and sends back and forth data to the front-end. It handles errors and it prints debugging messages when an action is performed.

`indexing_script.py` and `destroy_index.py`

The indexes can be created and destroyed by pressing the button on the home screen. I took the stop words from this [git](#).

`search_script.py`

Depending on the arguments, this script will retrieve and return the information the user wants. It utilizes a MultiParser in order to search in the Titles, Overviews,

and Keywords for a more accurate information retrieval. It retrieves 2000 results, which is ~5% of the initial dataset. After applying filters (genres) the script returns the top 15 results. The reason behind the initial 5% of retrieval is purely for performance issues. The number could be adjusted accordingly to the size of the dataset.

README.txt

Contains information about the correct dependencies installations and the running of the system.

- **Evaluation**

The results are not too hard to invalute taking into consideration that the dataset contains correct data. Swoosh Library returns information that is relative to the query. When the user searches the titles, the library returns movies with the query in the title. If genre filters are applied, the movies have these genres (shown explicitly in the results table).

- **Results of Queries (discussion)**

Example 1)

Search query “toy” in “titles” with genre filter:

Search a Movie!



- ☐ Documentary ☐ Mystery ☐ Foreign ☐ Thriller ☐ History
☐ Music ☐ TV Movie ☐ Fantasy ☐ Western ☐ Family
☐ Animation ☒ Comedy ☐ Fiction ☐ Horror ☐ Crime
☐ Drama ☐ Adventure ☐ Action ☐ Romance ☐ War

Results

Title	Tags	Watch
The Toy	Comedy,Family	<input type="button" value="Watch"/>
The Toy	Comedy	<input type="button" value="Watch"/>
Toy Story	Animation,Comedy,Family	<input type="button" value="Watch"/>
Toy Story 2	Animation,Comedy,Family	<input type="button" value="Watch"/>
Toy Story 3	Animation,Family,Comedy	<input type="button" value="Watch"/>
Toy Story of Terror!	Animation,Comedy,Family	<input type="button" value="Watch"/>

All the titles contain the word “toy” and every movie has the “Comedy” Genre Tag.

Example 2)

Search query “toy” in “Descriptions” without a genre filter:

Search a Movie!

 ▼

- ☐ Documentary ☐ Mystery ☐ Foreign ☐ Thriller ☐ History
☐ Music ☐ TV Movie ☐ Fantasy ☐ Western ☐ Family
☐ Animation ☐ Comedy ☐ Fiction ☐ Horror ☐ Crime
☐ Drama ☐ Adventure ☐ Action ☐ Romance ☐ War

Results

Title	Tags	Watch
Dolls	Fantasy,Horror	<input type="button" value="Watch"/>
Small Soldiers	Comedy,Adventure,Fantasy,Science Fiction,Action	<input type="button" value="Watch"/>
Toy Story That Time Forgot	Animation,Family	<input type="button" value="Watch"/>
Alma	Animation,Fantasy,Mystery	<input type="button" value="Watch"/>
Silent Night, Deadly Night 5: The Toy Maker	Horror,Science Fiction	<input type="button" value="Watch"/>
Toys	Fantasy,Comedy,Science Fiction	<input type="button" value="Watch"/>
Home Alone 3	Comedy,Family	<input type="button" value="Watch"/>
Mr. Magorium's Wonder Emporium	Comedy	<input type="button" value="Watch"/>
Partysaurus Rex	Animation,Comedy,Family,Fantasy	<input type="button" value="Watch"/>
Christmas Evil	Drama,Horror	<input type="button" value="Watch"/>
Toy Story 2	Animation,Comedy,Family	<input type="button" value="Watch"/>
Toy Reanimator	Science Fiction,Fantasy	<input type="button" value="Watch"/>
There's Always Tomorrow	Comedy,Drama	<input type="button" value="Watch"/>
Child's Play 3	Thriller,Horror,Action	<input type="button" value="Watch"/>
The Christmas Toy	Animation,Family	<input type="button" value="Watch"/>

The movies retrieved contain the word toy in their description. Not necessarily in their title.

Example 3)

In this example, we will search the same query as in example 2. This time I searched both titles and Descriptions.

Search a Movie!



- ☐ Documentary ☐ Mystery ☐ Foreign ☐ Thriller ☐ History
☐ Music ☐ TV Movie ☐ Fantasy ☐ Western ☐ Family
☐ Animation ☐ Comedy ☐ Fiction ☐ Horror ☐ Crime
☐ Drama ☐ Adventure ☐ Action ☐ Romance ☐ War

Results

Title	Tags	Watch
Toy Story That Time Forgot	Animation,Family	Watch
Toy Story 2	Animation,Comedy,Family	Watch
Toy Reanimator	Science Fiction,Fantasy	Watch
Silent Night, Deadly Night 5: The Toy Maker	Horror,Science Fiction	Watch
The Christmas Toy	Animation,Family	Watch
Dolls	Fantasy,Horror	Watch
Small Soldiers	Comedy,Adventure,Fantasy,Science Fiction,Action	Watch
The Toy	Comedy,Family	Watch
Toy Story of Terror!	Animation,Comedy,Family	Watch
Toy Story	Animation,Comedy,Family	Watch
The Toy	Comedy	Watch
Tin Toy	Animation	Watch
Alma	Animation,Fantasy,Mystery	Watch
Toy Story 3	Animation,Family,Comedy	Watch
Toys	Fantasy,Comedy,Science Fiction	Watch

Every movie here has “toy” in the title and in their description. The ranking is depending on both the titles and the descriptions.

Here is a [demo video](#). Due to some technical issues the quality is not great, but considering the README files and the pictures above, it shouldn't be a problem figuring out how it works.

- Limitations

I'm aware that the results are not every time in the order a user is used to from other engines. This can be fixed by utilizing machine learning. When the “watch” button is pressed, could train the search engine to show first what most of the users are looking for, and extensionally, adjust to the specific taste of the user.

The code is not in a “src” folder. There is a README file that explains how to run the application.

The system searches for the information using the Titles, Overviews, and Keywords. It could implemented to use more criteria but the complexity of the program goes up exponentially.

Sometimes, when the program starts it bugs and the homepage is frozen. With a simple refresh of the home page in your browser, it's ready to go.