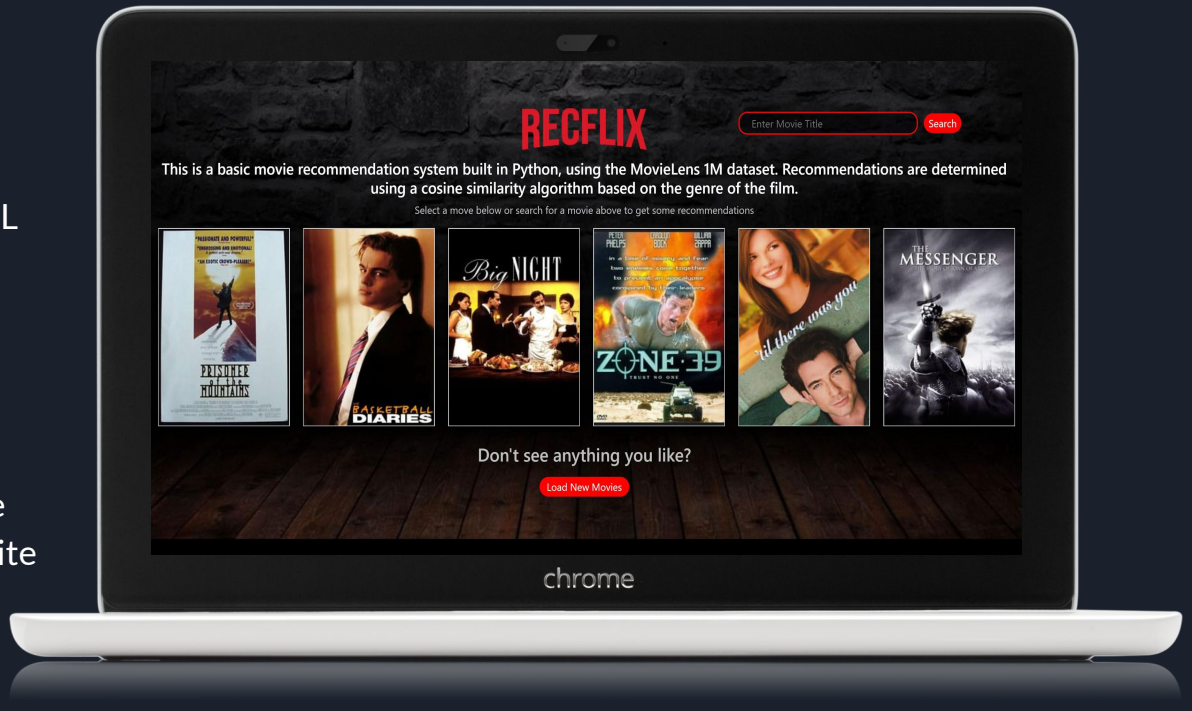
Final Project:

A movie recommendation website created using Python (Pandas, scikit-learn, Flask), jQuery, HTML, & CSS.

By: Greg Leland

Simplified Overview

- Download the MovieLens 1M Dataset
- Import and format the data
- Find a feature of the data to perform ML
- Feed the feature into the algorithm
- Extract results
- Source images for the movie titles
- Create master dataframe
- Streamline the process
- Create a HTML/CSS skeleton/structure
- Create a Flask server to populate website
- Style the website with HTML/CSS
- Create and configure Flask app routes
- Add features and functionality
- Apply finishing touches



Challenges

1. What features should I use?
2. What ML model should I apply?
3. What data do I need to keep, and what can I strip away to lighten the size and required resources to process/store the data?
4. How in the heck do I get Flask to play nicely?
5. What are some useful features I can add?

My Limitations:

The cosine similarity model that I used gets somewhat accurate results a good percentage of the time, but it is definitely limited because it is solely based on the genre of the movie. Because of this, there is a very shallow depth of results, which will often return the same 12 movies if any of the results are searched.

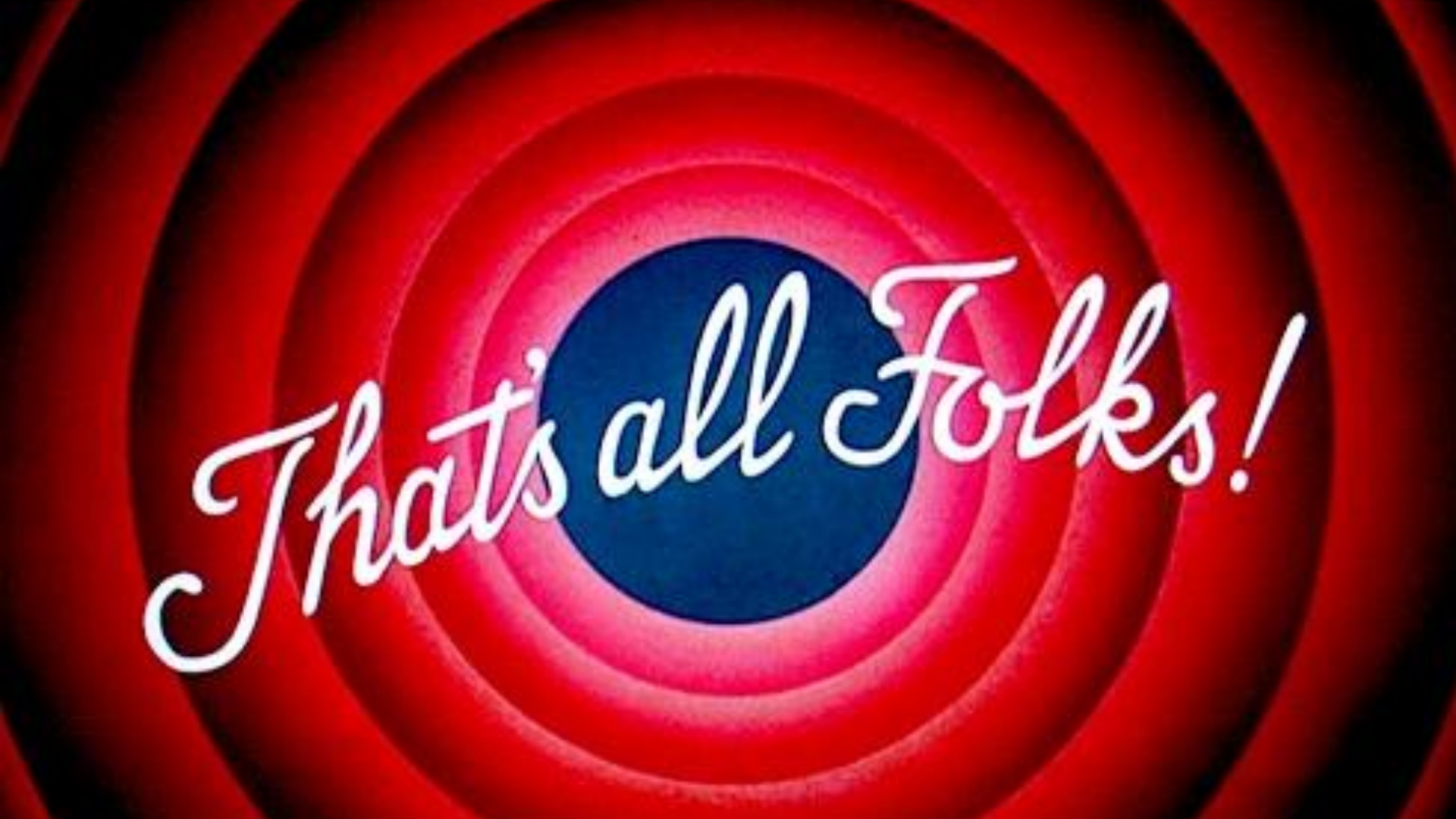
TIME. If I had more time to work on the project, I would have been able to test and train different models to get different results.

Future Plans

Test different models on the dataset and add different search results / routes to the website based on user ratings and other features that seem interesting/useful. Apply deep learning models to the data to see how accurately it can predict user preferences.

Create a SQL database and reconfigure the Flask app to query the database instead of a CSV file.

Find a dataset or a way to scrape movie descriptions and add functionality to the website by and adding hover tooltips on the images, etc.


That's all Folks!