

# A Simple Movie Recommender System

---

Wenjing Yang



# Motivation

---

**Motivation:**

To create a web app which can provide users with movie suggestions according to the user input.

**User Input:**

The name of the movie that the user enjoys.

**Website Output:**

A list of 10 movies recommended to the user.

**Website Link:**

<http://final-deploy-dev.us-west-2.elasticbeanstalk.com/>



# Data Description

---

**Data Source:**

The MovieLens dataset on *Kaggle.com*.

**Data Description:**

Movies: 45,000 movies listed in the Full MovieLens Dataset

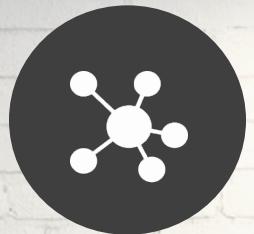
Duration: before 2017

Tables: credits (*cast, crew, id*), keywords (*keyword, id*), movies\_metadata (*genre, title, id*)

**Data Preparation:**

EDA

Merge tables on *id*



# Model Insight

---

**Algorithm Highlight:** content-based recommender system

**Logic Flow:** movie name (user input) → cosine-similarity (matrix) → 10 movie suggestions (website output)

**Success Criterion:** user satisfaction (subjective)

## Model Selection:

- Model 1: recommender system based on movie descriptions
  - transform movie descriptions into matrix to calculate cosine-similarity
- Model 2: recommender system based on movie director, main actors, genres, and keywords
  - extract directors, actors and genres from the JSON-format text
  - keep top2 actors for each movie to make sure model relevancy
  - create a new variable to save all information of movie director, main actors, genres and key words
  - transform the new variable into matrix to calculate cosine-similarity



# Model Discussion

---

## Interesting Findings:

- The content-based recommender system is good for users with consistent tastes.
- To provide users with more relevant movie suggestions based on their input, we can add more movie information to our model for calculating the cosine-similarity matrix.

## Limitations:

- The web app will return the exactly same movie suggestions to all users with the same movie input.
- For users who want to explore new types of movies, this recommender system might fail.

## Improvements:

- The method of collaborative filtering (UBCF & IBCF ) can provide users with more customized movie suggestions.
- Hybrid recommender system can be the next step.



# Thank You!

◆ Contact info:  
[wenjingyang2018@u.northwestern.edu](mailto:wenjingyang2018@u.northwestern.edu)