Recommendation Engine for ‘Life Altering Films’

Springboard - Data Science Intensive

By Justin Barton

August 2016

Finding films that will: inspire you, challenge your view of the world, make you think and make you feel – <u>'Life Altering Films'</u>

What are the limitations of current techniques?

- A lot of time spent searching searching the web or using recommendation systems
- Even with the most rigorous of efforts from the users, many sites and recommendation engines are not well targeted to the user.
- Many will not find the films they were meant to watch;
    - The filmmaker cannot spread their message wide enough,
    - The population cannot challenge their view of the world

Potential limitations to current recommendation engines?

- Focussed on finding films you are likely to pay to watch
- Not hand crafted to the specific user; tackling the problem at scale
- Getting enough user data (their ratings) on which to make recommendations
    - The hypothesis - more data typically trumps a better model
    - How many previously rated films are needed from the user?
    - Gamifying the experience to get the most user data

The Datasets

- **MovieLens** – historical user ratings (films previously rated by 1000s of other users)
- **OMDB** – budget, revenue, runtime, job titles (Director, Producer, Editor etc.)
- **IMDB** – Actors (actor name, and # ranking in credits)

Merging and cleaning the datasets

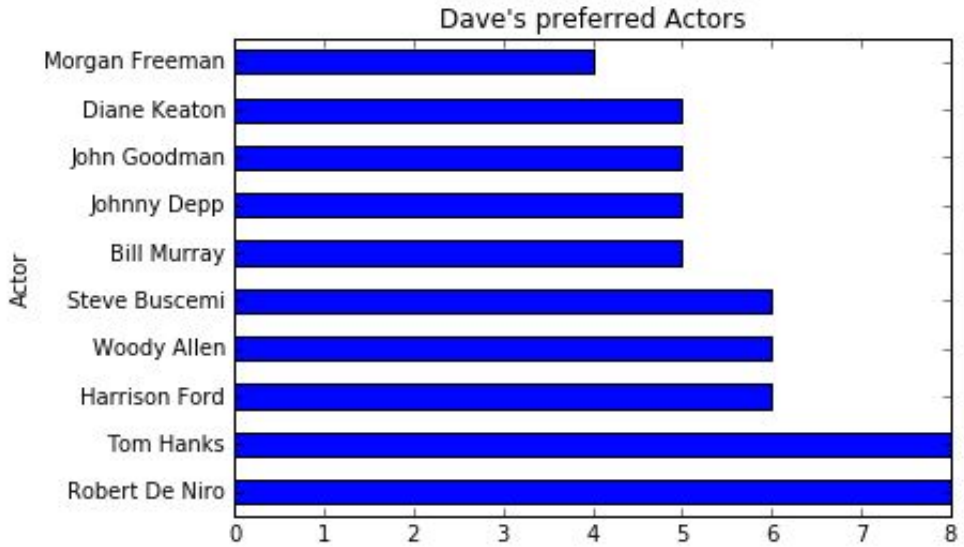
- Datasets not linked - create algorithm for matching on title and year
- Need a robust system as there are various edge cases on matching:
  - 'City of God (Cidade de Deus)' in MovieLens and 'City of God' in OMDB
  - 'Birdman' in MovieLens and OMDB and 'Birdman or (The Unexpected Virtue of Ignorance)' in IMDB
  - 'V for Vendetta' is year 2006 in MovieLens and OMDB, but year 2005 in IMDB
  - ‘Usual Suspects, The (1995)’ in MovieLens is ‘The Usual Suspects’

Generating surveys and collecting data

- Need films users are likely to have seen (e.g. English, at least 10 ratings, after 1965)
- Aggregate previous ratings, take the top 1000 films, randomise for new users
- Gathering new user ratings from Google Sheets
    - Connecting to Google Sheets crucial to make getting data quick
    - Not well documented to connect to pandas
    - Speed and authentication issues to workaround.

Creating profiles for each user

- Profiles are made up of features, which could be in various datasets in various forms
- Using user profiles to get an insight as to what features might work



Linear regression models per new user

- Linear regression on film attributes:
  - budget, revenue, runtime, num_ratings, average rating
  - Altering this on ‘good films’ or all films, and all features or individual features
- Linear regression on old user ratings:
  - Finding old users who had rated the most in common with the new user
  - Find models with 1 - 500 different size of users
  - New users who rate more films, have substantially more films in common with old users

Finding old users with similar ratings to new users

- Find old users who have rated the same films
- For each old rater, reduce the list to films that only both have rated
- Calculate new and old user rating's standard deviation and mean
- Compare the difference in these stats between old and new (ratio difference)
- Remove old users who differ to new user by some tolerance
- Extract all of the films of these old users as recommendations
- In the future, rigour can be added to this approach with classification models

Conclusion

- Each user prefers different people involved in the films they love
- There exists similar users out there to yourself
- Not all features are useful
- Linear models are limited, classification models are needed.
- Hand crafted single person models take time - trade-off with easier scaled methods
- How much user data is needed to find their Life Altering Films?