Wolf of bolly street

Aditya Jain - 193050028
Kartavya Kothari - 193050021
Karthik Prakash - 193050008
Sanjay Kumar - 193050068

November 2019

Contents

1 Introduction

In the day and age of demand for profitability and success in every work and dimension, it becomes imperative to produce high quality and accurate results to achieve these objectives. Also, the availability of better resources coupled with smarter programming logic has propelled artificially intelligent machines that simulate human logic. The prediction of profitable and qualifiable possibilities not by an expert human but by a computer system has led to the birth of prediction systems.

These prediction systems make use of machine learning, which is essentially an expert system that learns patterns from the data from the past, to predict patterns as to how will the future statistics be. Ranging from daycare to entertainment, today almost each sector employs machine learning i.e. prediction systems or recommendation systems to evaluate optimised solutions for relative problem statements. Such systems favour user's requirements, along with obtaining excellent output results and improving efficiency.

Owing of the expanding size of the film industry, due of hundreds of talented but hidden newcomers and the prominent presence of renowned actors, classifying these actors as per quality is a herculean task requiring end-to-end knowledge via social reach and contacts, excellent relations with talent agencies. Moreover, the job of a casting director is a huge challenge. (S)he is required to identify, classify and precisely suggest actors that promise to not only match the script suggested by the producers or directors, but match up to the genre and the work level requirements.

For new directors in the industry, identifying suitable actors for their script is a hindrance even with the presence of casting directors, as a good movie needs a huge investment, and there always exists a chance of a flop movie even if undivided efforts are put up. In certain instances, it can happen that personal bias towards certain prominent actors hamper chances for other talented actors and spoil the success of the film at the box office.

To overcome such discrepancies, a recommendation system can be built up to suggest a star cast, that can solve the above inefficiencies and ease the job of directors simultaneously.

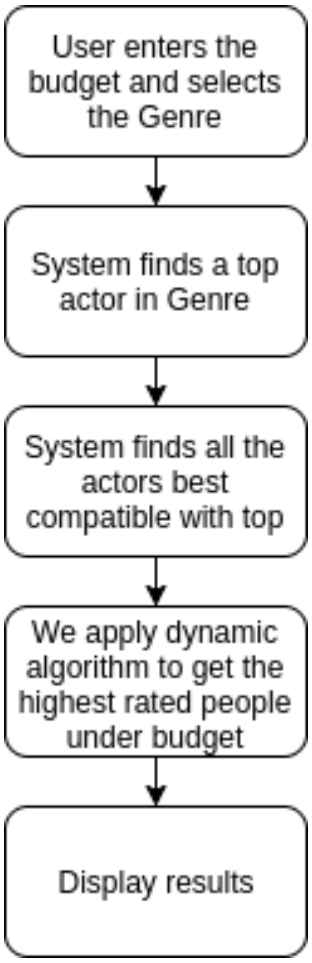

Figure 1: Project flow

2 Goals

- **Recommend a statistically most likely to succeed star cast**

  We take the data of most successful movies and check which stars were cast together. This is based on the assumption that if a star cast has worked together and given a blockbuster, they will most likely deliver again

- **Recommend an inter compatible star cast**

  We use Apriori algorithm [1] which checks through the data and gives the set which appear together many times. This is based on assumption that if a cast has worked many times together, they will be compatible to work with each other the next time too!

- **Minimize the budget and maximize the profits**

  We use the dynamic 0/1 knapsack algorithm to ensure that we get the best rated cast using our budget to the maximum. The assumption here is that the rating of the actor has been very carefully given across multiple different factors and is a good factor to decide the profit we will make while casting such actors

- **Automate the star cast selection procedure**

  We have in place the streamlined training model in place with the preprocessing and everything set up. All the director needs is to update the data, call train and new model is ready to throw predictions. The assumption here is that the data is updated regularly and with correct data.

- **Generate alternative star cast if not happy with current**

  We have a button on the result event UI which runs the recommend part of code again. We get different result due to the rnadom nature of choosing the top actor from the genre based data as a seed to apriori. The assumption here is that we will get a different top actor from random module when we re run the module

- **Make cast generation standard and efficient**

  We have identified the parts of code which we are core services in generating cast. All those functionalities have been presented in a concise UI. We have a set structure of data which can be easily updated. It is very intuitive and lets directors generate cast without hassle and enjoy a statistically optimal cast

3 Literature

Variety VScore [2] developed an actor rating system that rates an actor between 1 to 100 based on his or her performances at the box office and social media presence. Allotting a value to the actor leads to a ranking of best actors, which helps casting directors to select the best actors for their intended purpose. However, the cast selection is to be done manually and there is no budget filter available.

Piedmont Media Research [3] have created an actor valuation system that evaluates the ideal fee that should be charged by an actor in the industry for his/her work. This system helps directors from being overcharged by performers. It employs social media popularity, box office hit count and earlier actor charges to evaluate the current value.

A New York based media website, Vulture [4], explained the factors based on which they created a list of the best actors. Some of them include domestic and overseas box office collections, studio value, critics' score, awards, public likeability and tabloid presence.

In terms of implementation, the adaptation of association rule learning to subgroup discovery using apriori algorithm, researched by Branco Kavsek and Nada Lavrac [1], helped to gain a better understanding of the weighted covering algorithm and properties of the weighted relative accuracy heuristic. The scope of investigation is limited to only to patterns with a certain property of interest. Moreover, producing smaller subsets improves accuracy and reduces computation time, thus stressing on the need of elimination of irrelevant items.

4 Set of approaches

We considered following approaches

1. **Regression** This was the most naive way to think. As we wanted to model the inter compatibility between actor sets, this proved to be way too simple to model the distribution
2. **Neural networks** Designing a neural network approach was on the table but we couldn't map the design to a feasible model based on association mining. We believe this can be leveraged in future to suggest better prediction models
3. **Apriori algorithm** We use the apriori algorithm to generate sets of actors most likely to appear in a movie (Inter compatibility, assuming if a set of actors have appeared in a movie before and are coming quite a lot of times, then they are compatible). The sets of actors we get from the trained model will now be input to dynamic 0/1 knapsack algorithm to get maximum profit set (Highest rated actors) while minimizing weight (What the actors approximately demand)

5 Experiments

5.1 Language and environment

- Jupyter Notebook 6.0.1 (Coding the Application)
- Python 3.7.4 (from conda 4.7.12)
- MLXtend for aprioiri model
- Python Libraries - PyQt5 for UI
- Python Libraries - pandas and numpy for data handling
- (Tested on) Operating System: Ubuntu 16 or later, Windows 8.1 or later

FILES:

1. We have main file entry point StarCastRecommend.py (can be run as)

   python3 StarCastRecommend.py

2. Two UI files (created using PyQt5 designer)
   (a) projectUI.ui: Main page accepting user inputs
   (b) result.ui: Page showcasing the results

3. One data pre-procesing code, GenreDivide.py

   This code deals with the majority of data pre processing and structuring. It creates the TopActors.csv

4. One Apriori trained model generation code

   This code generates the sets which have high association

5. Data files:

   (a) req.csv: Contains actors, their asking cost and normalised rating

   (b) files/newresultsapriori.csv: contains apriori trained output

   (c) files/TopActors.csv: contains genre based actor segregation (*Derived using data pre-processing from original data*)

Link to access data: https://git.cse.iitb.ac.in/kartavya/the-wolf-of-bolly-street

5.2 Experimental results

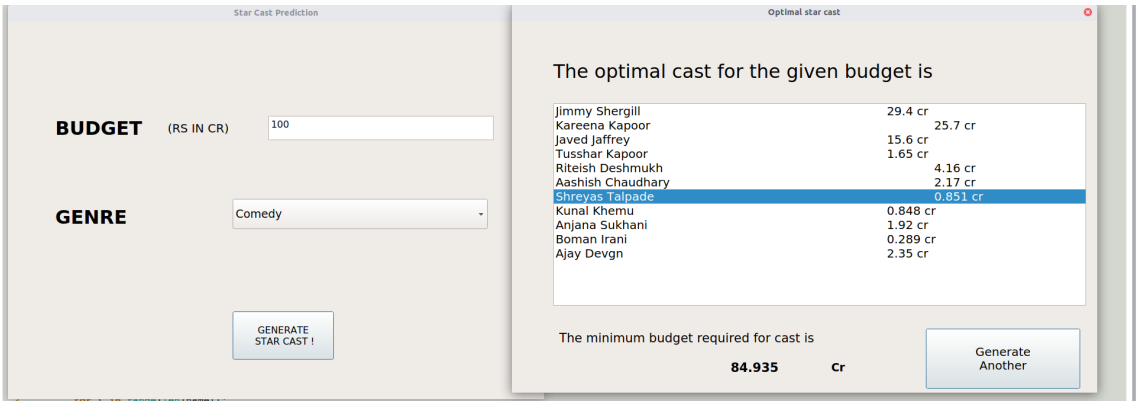

Figure 2: Genre comedy with budget 100cr

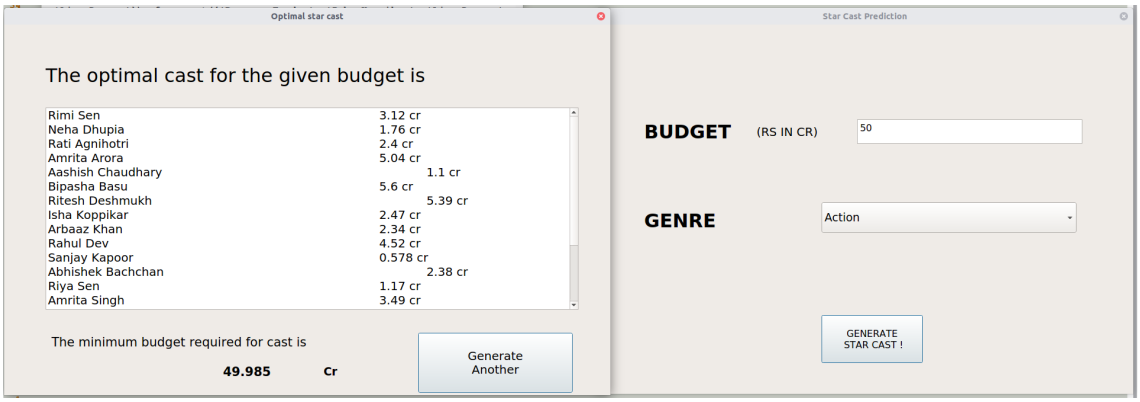

Figure 3: Genre action with budget 50 cr

6 Effort

6.1 Code

1. Algorithm decision and model It took a week for us to decide on the overall approach and algorithm. We identified models that would fit in place well enough for a proof of concept.

2. Data (2 weeks)

   - We got the compiled data from kaggle (https://www.kaggle.com/mitesh58/bollywood-movie-dataset).
   - We identified and hence compiled the data to the format we required. We worked in teams of two and one of the teams worked parallel for the two data processing tasks.

3. Algorithm code modules (2 weeks)

   (a) Train the model to get associations in the from of antecedents and consequent.

   (b) Choose a top actor from given genre

   (c) Give the top actor as a seed to the Apriori algorithm and get the set

   (d) Dynamically maximizing actor rating in given budget range (Dynamic 0/1 knapsack)

   (e) Random module to get alternate cast

4. User interface (Half a week)

   (a) Main user interface page to accept inputs from user

   (b) Call the recommend module from the main screen

   (c) A display screen will show the results

   (d) A "generate another" button calls the random module to give a new cast for the same parameters picking up a new top actor

6.2 Challenges

- We didn't have the actual amount the actors charged. Our first thought was to scrape the data but that turned out to release a whole new plethora of challenges. Hetero-genity of formats, even in structured data that wiki provides (Different currencies). Inconsistency with the data being 5-7 years old. So to counter it we decided to go with a different approach.

- We mapped how much they can charge with their popularity. We mapped the popularity with the google hits based on what data we had available with us.

- This gives way to another good project, that would be related to predicting how much an actor should charge! It would be based on social media scores and such (We see similarity with the techniques of page rank but that's something to explore for future)

References

[1] Branko Kavsek and Nada Lavrac. *Apriori SD – Adapting association rule learning to subgroup discovery.* Applied Artificial Intelligence, 20:543-583, Reading, Massachusetts.

[2] Variety Vscore. https://www.varietyinsight.com/vscore/index.php.

[3] Piedmont Media. Piedmont media research actor valuation http://www.piedmontmedia.com/concept-testing.

[4] Vulture. How vulture ranked its 2013 most valuable stars list http://www.vulture.com/2013/10/most-valuable-stars-2013-methodology.html.