Movie Success Predictor

KEYWORDS Artificial Intelligence, Machine Learning, Social Networking Services, Success Prediction, Forecasting Models

Prithvi Sharan S

Department of Computer Science and Engineering, M S Ramaiah Institute of Technology, Bangalore, Karnataka, India

ABSTRACT *In this paper, we propose a movie success predictor which will predict the likelihood of a movie to be successful in theatres. We first propose the various parameters to be considered which include the actors' ratings, the director's ratings, the budget, popularity of the trailer to name a few. The parameters will then be considered and using Artificial Intelligence concepts, a success number will be generated. This number generated will also be stored in the database for further learning by the proposed machine. The proposed machine will obtain ratings and other relevant data from different resources, IMDb and Rotten Tomatoes to name a few. Using the genetic algorithm and the, significnt input variables are determined. Furthermore, three machine learning-based nonlinear regression algorithms and their combinations are employed for building forecasting models. We also propose to use Social Networking Services' (SNS) to better the results. With this knowledge, we expect the performance of our proposed machine to be high.*

INTRODUCTION

Recommendation systems are originated from different areas such as approximation theory, cognitive systems, informa- tion retrieval, prediction methods, and management science. In parallel to internet technologies, recommendation systems have been used more especially in e-commerce for movie, music, There are very well known e-commerce examples such as Amazon, MovieFinder, eBay, Reel.com, and Netflix.

As in our daily life, we rely on the idea our friends, peers who share the same likes/dislikes and if they recommend an item, we are likely to enjoy it or vice a versa.

The system aims to predict success of a movie, based solely on what is known about a movie before its release in theatres. The proposed system can be looked at from two diverse angles. One from a movie maker's point of view and the other from a movie watcher's view. The movie maker would be able to make more informed decisions on his movie project using this tool. This can include decisions from how successful a movie will be to what kind of resources required for the project. The movie maker can then alter his parameters as required to meet his target based on the success numbers generated. The movie watcher can know of the success of a movie before the movie is even released. Also the rating given to other movies by that person can be recorded, stored and used in the generation of the success number and wil hencel make it more aaccurate. In short the software would be able to tell you if the movie would be a worthwhile investment or not. The reason for the selection of this project is the very fact that the world around us is influenced by the entertainment industry. The scope of this project can influence the standards of entertainment to a positive degree as those project which have a low rating need

not be started. Also its impact on the market is far and wide as it can cater diversely from daily soaps to big budget films.

METHODOLOGIES This is the other section that you can use.

Data retrieval

Manual data retrieval: Manual data retrieval involves the use of tables to store the data. Data is inserted into the table manually by programmers and then data is retrieved from it manually as well. Hash map is the proposed data structure to be used. A hash map is a two tuple set consisting of a key and a value. Each value is associated with a unique key. So given the key, its corresponding value can be obtained form the hash map using appropriate functions.

Online Db data retrieval: **Page scrapping** Use the standard HTML interface and "scrape" out the info that is required. This requires the use of sockets. Such a program easier to write in another language such as Python, Perl, or even Java i.e. languages that have a regex engine. The biggest problem with this approach is that when the website makes even little changes to their web page, the program code must be changed. **IMDb data files** These are text files and you can process them into any format you may find useful. Problem with this approach is that you would need to download the files every so often to maintain a current Db. **Third party API** These include access to IMDb and Raotten Tomatoes publically available API's for their immense resources on movies, cast, budget, genre among other useful information.

Predction method

Table lookup: **function** TABLE-DRIVEN-AGENT (percept) returns *action*

static: *percepts*, a sequence, initially empty *table*, a table, indexed by percept sequences, initially fully specified

append percept to the end of percepts *action*¡— LOOKUP(*percepts, table*) **return** *action*

The savings in terms of processing time can be significant, since retrieving a value from memory is often faster than undergoing an 'expensive' computation or input/output operation.The tables may be precalculated and stored in static program storage, calculated as part of a program's initialization phase, or even stored in hardware in application-specific platforms.

Support Vector Machine

The second machine learning technique we applied was SVM. With SVM, we could use all 176 of our input vectors and then pare down the space by use of filter feature selection, which uses the forward search paradigm to choose a subset of features with which to make predictions. As with locally weighted linear regression, we predicted whether a movie performed better or worse than the median found across our entire dataset for each output feature. We asked the following questions (based on the medians):

Does a movie have a Rotten Tomatoes critics score above 60 (on a scale to 100; Rotten Tomatoes labels a movie either fresh with a 60+ score or rotten with a score below 60).

Does a movie have a Rotten Tomatoes audience score above 64 (the median value, also on a scale to 10.

Did a movie gross over USD7.49M in the United States (the median)?

Did a movie gross over USD500K in the United States its opening weekend (the median)?

Did a movie have an IMDB score of 6.5 or above (the median, on a scale to 10).

For each output feature, we performed a separate filter feature selection. We grouped bit vector features that were from the same category. For example, our dataset has 22 bit vectors to represent a movies genre categorization (22 total genres). We grouped these 22 features together when performing filter feature selection.

DATA CLEANING AND IMPROVEMENT

After some preliminary runs of our machine learning algorithms, we had decent results, but felt we could do better if we cleaned up the data. Each IMDB rating comes with both a numeric rating and the number of votes cast. Concerned that less well-known movies with fewer votes would be unreliably rated, we plotted the number of votes vs. the rating, and what we found was interesting. Movies with more than 100 votes trend towards a higher rating with more votes. However, with fewer than 100 votes, there is little structure to the data. Based on this, we removed all movies with fewer than 100 votes from our dataset.

DYNAMIC ONLINE RETRIEVAL

Data needs to be retieved dynamiclly from online databases like IMDb and RottenTomatoes as the user votes change on an hourly basis. Such a retrieval mechanism will make the heuristic value of the actors and actresses more accurate. This can be established by introducing a hardware interface between the API of the database and our proposed system.

VISUALISATION

We can use visualisation tools like Gephi in order to better understand the strength of associations between two actors, and this would directly influence their on screen chemistry.

FUTURE ENHANCEMENTS

Better accuracy through the use of a more sophisticated algorithm and agent architecture. A better algrorithm can be developed through the formulation of an optimising function with the use of the use of regression curves. Better agent architecture can be developed by using SVMs.

Include algorithms that consider regional specifications, time of release and some trivial parameters. Also the languages in which the movie is going to be released will play a role. If a movie is dubbed into multiple languages is likely to reach to a wider audience.

Better database querying can enhance the overall performance of the system.

Final application must be available on mobile platforms for ease of access. PUSH notifications can be implented for convenience

CONCLUSION

There are many techniques that we can use to recommend the success of a movie ,some of them being ta- ble lookup,SVM,decision trees nd to an extent even neural netwroks. These methods can be used appropriately to increase the number of parameters and thus increase the efficiency of the overall result.

Smarter retrival startegies that yield quicker results can be incorporated. Visualization tools to increase the user efficiency and thus increase the user base is of utmost importance. In terms of market value we feel that it has a wide scope as it is targeted towards one of the largest industries being the entertainment industry.

REFERENCES

1. A Movie Rating Prediction Algorithm with Collaborative Filtering(paper)
2. A Predictor for Movie Success Jeffrey Ericson and Jesse Grodman CS229, Stanford University
3. www.wikipedia.com www.imdb.com www.rottentomatoes.com
4. Predicting movie success with machine learning and visual analytics- Philip Omentisch