Analyzing “MovieLens” Dataset with Association Rules

Bartłomiej Pukacki 151942
Mateusz Tabaszewski 151945

Table of Contents

Introduction

This assignment is meant to show both the process as well as results of analyzing the "MovieLens" dataset, in particular, the focus was put on the "ratings" and "movies" datasets available in the CSV format. "movies" dataset contains information regarding the title of the movie, production year, as well as genres to which the film belongs. Each movie is uniquely recognized by its id, in a dataset named "movieId". When it comes to the "ratings" dataset, the available information pertains to the rating given by a user, with each user having their own unique "userId", id of the reviewed movie, as well as the timestamp showing a number of seconds since 1st of January 1970 as of the time of submitting the review.

The goal of the assignment is to find and showcase interesting and unique Association Rules, hopefully with some real-life applications. This report as well as the submitted notebook file are meant to show the process and explain how the rules were obtained, an exemplary application was also provided to show how the inferred rule set could be used and how a similar approach could be used for other, similar datasets to create a recommendation system for movie-streaming sites.

Attempted Methods

In one of the experiments rules aiming to find associations between year-based ratings of unique users were searched. Every combination of year and simplified rating value (very bad/bad/mediocre/good/very good) was created and for each distinct user occurrences of year-rating combinations were binarized (1902|very bad:0/1, 1902|bad:0/1, etc.). The resulting rules formed associations of the following arrangement: (1994|good, 1993|good) -> (1995|good).

	userId	1902|very bad	1902|bad	1902|mediocre	1902|good	1902|very good	1903|very bad	1903|bad	1903|mediocre	1903|good	...	2017|very bad	2017|bad	2017|mediocre
605	606	False	False	False	False	False	False	False	False	False	...	False	False	False
606	607	False	False	False	False	False	False	False	False	False	...	False	False	False
607	608	False	False	False	False	False	False	False	False	False	...	False	False	False
608	609	False	False	False	False	False	False	False	False	False	...	False	False	False
609	610	False	False	False	False	False	False	False	False	False	...	False	False	True

Fig 1. Example of data frame with previously stated structure.

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction	zhangs_metric
0	(1990|mediocre)	(1994|good)	0.401639	0.734426	0.365574	0.910204	1.239340	0.070599	2.957526	0.322747
1	(1991|good)	(1994|good)	0.488525	0.734426	0.444262	0.909396	1.238240	0.085477	2.931148	0.376171
2	(1992|mediocre)	(1994|good)	0.375410	0.734426	0.339344	0.903930	1.230798	0.063633	2.764382	0.300227
3	(1993|mediocre, 1989|mediocre)	(1994|good)	0.321311	0.734426	0.304918	0.948980	1.292137	0.068938	5.205246	0.333126
4	(1993|mediocre, 1989|mediocre)	(1995|mediocre)	0.321311	0.654098	0.301639	0.938776	1.435221	0.091470	5.649727	0.446807

Fig 2. Example of data frame with induced rules.

This approach did not produce very informative results and the experiment was generalized to include year intervals (e.g. (1995,2000]) and a narrower rating scale (bad/mediocre/good). Despite the changes, finding informative rules was difficult. Many of the rules seemed to not exclude the existence of other rules that would imply a different conclusion. This might be due to the fact that counting at least a single occurrence of a rating in a year interval is not representative of the likelihood of different ratings appearing together as the information about the frequency of ratings is lost due to the chosen grouping.

	userId	(1920,1930]|bad	(1920,1930]|mediocre	(1920,1930]|good	(1930,1940]|bad	(1930,1940]|mediocre	(1930,1940]|good	(1940,1950]|bad	(1940,1950]|medic
605	606	False	True	True	False	True	True	False	T
606	607	False	False	False	True	True	True	False	F
607	608	False	False	False	True	True	False	True	T
608	609	False	False	False	False	False	False	False	F
609	610	False	False	False	False	True	False	False	T

Fig 3. Example of data frame with previously specified structure.

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction	zhangs_metric
0	((1970,1980]|mediocre)	((1990,2000]|mediocre)	0.639344	0.980328	0.636066	0.994872	1.014836	0.009299	3.836066	0.040534
1	((1970,1980]|mediocre)	((1990,2000]|good)	0.639344	0.934426	0.606557	0.948718	1.015295	0.009137	1.278689	0.041769
2	((1980,1990]|good)	((1980,1990]|mediocre)	0.662295	0.852459	0.604918	0.913366	1.071449	0.040339	1.703044	0.197464
3	((1990,2000]|bad)	((1980,1990]|mediocre)	0.798361	0.852459	0.709836	0.889117	1.043003	0.029266	1.330601	0.204472
4	((1980,1990]|mediocre)	((1990,2000]|bad)	0.852459	0.798361	0.709836	0.832692	1.043003	0.029266	1.205201	0.279446
5	((1990,2000]|mediocre)	((1980,1990]|mediocre)	0.980328	0.852459	0.845902	0.862876	1.012220	0.010212	1.075970	0.613695
6	((1980,1990]|mediocre)	((1990,2000]|mediocre)	0.852459	0.980328	0.845902	0.992308	1.012220	0.010212	2.557377	0.081826

Fig 4. Example of data frame with induced rules.

Yet another attempt was meant to capture relationships between individual film reviews and the films genres, years of production as well as the day of the review, and what time of the day the review was submitted, although this allowed for some level of interpretability, it ultimately did not provide information which could have immediate real-world use.

	rating_low	rating_medium	rating_high	year_old	year_new	Action	Adventure	Animation	Children	Comedy	...	reviewed_Monday	reviewed_Tuesday	rev
0	0	0	1	1	0	0	1	1	1	1	...	0	0	
1	0	0	1	1	0	0	1	1	1	1	...	0	0	
2	0	0	1	1	0	0	1	1	1	1	...	1	0	
3	1	0	0	1	0	0	1	1	1	1	...	0	0	
4	0	0	1	1	0	0	1	1	1	1	...	0	1	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	

Fig 5. Example of data frame with the specified structure.

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
0	(rating_low)	(year_old)	0.189149	0.610546	0.109802	0.580506	0.950799	-0.005682	0.928391
1	(year_new)	(rating_low)	0.389276	0.189149	0.079317	0.203755	1.077222	0.005686	1.018344
2	(rating_low)	(year_new)	0.189149	0.389276	0.079317	0.419336	1.077222	0.005686	1.051769
3	(Action)	(rating_low)	0.303810	0.189149	0.062031	0.204178	1.079459	0.004566	1.018886
4	(rating_low)	(Action)	0.189149	0.303810	0.062031	0.327951	1.079459	0.004566	1.035921

Fig 6. Example of data frame containing induced rules.

Final Method

Overview

The main idea behind the final implementation of the Association Rule finding algorithm for the “MovieLens” dataset, was to combine several possible attributes in order to create a flexible and robust set of Association Rules which could be used in a recommendation system for a film-streaming website. This approach required combining information from both “movies” and “ratings” datasets in order to obtain a system capable of taking into account multiple attributes.

First, the system uses a 3-bin binarizing method to classify all movies reviewed by a user as either “bad”(rating less than 3), “medium”(rating equal to 3), or “good”(rating greater than 3). That way the resulting data frame has rows corresponding to every user and 3 columns corresponding to each movie, i.e. “movie_id bad”, “movie_id medium”, and “movie_id good”. Of course, in cases where the user has not submitted a review for a movie, the value for all 3 columns will be 0, otherwise, the value will be equal to 1 for the column corresponding to the appropriate category, for example, if a user gave the film number 23 a rating of 3.5, then the 3 columns corresponding to that film would have values like: “23 bad” : 0, “23 medium” : 0, “23 good”: 1. The main idea behind this system is to find films that may attract opposite preferences, as there may be films that are liked by those who disliked some other films, etc. Furthermore, columns corresponding to appropriate genres have been added as a means of identifying fans of certain genres and recommending either specific films or other genres based on the found rules. Each genre has a column with a value of 0 if the

average rating of films for that genre for the particular user is less than or equal to 4, and 1 if it is greater than 4. Lastly, the day of the week when the review was submitted was inferred from the "timestamp" column of the "ratings" dataset. Every day of the week also has its own column with a value of 1 for the day when the particular reviewer submitted the most reviews and 0 for all the remaining days. Perchance, the users who submit the most reviews on a particular day of the week may end up being fans of a particular genre, for example, people who mostly review films on Fridays may be fans of action or horror flicks.

The below visualization is meant to showcase how the matrix created according to the above description looks like before running the Apriori Algorithm for finding association rules:

	old	new	Action	Adventure	Animation	Children	Comedy	Crime	Documentary	Drama	...	Everything or Nothing: The Untold Story of 007 (2012) medium	Codependent Lesbian Space Alien Seeks Same (2011) high	2 Days in the Valley (1996) high	2 Days in the Valley (1996) low	2 Days in the Valley (1996) medium	Last Stand, The (2013) high	m
0	True	True	True	True	True	True	True	True	False	True	...	False	False	False	False	False	False	
1	False	False	False	True	False	False	False	False	True	False	...	False	False	False	False	False	False	
2	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	False	
3	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	False	
4	False	False	False	False	True	True	False	False	False	False	...	False	False	False	False	False	False	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	

Fig 7. Data frame containing the final agreed-upon visualization.

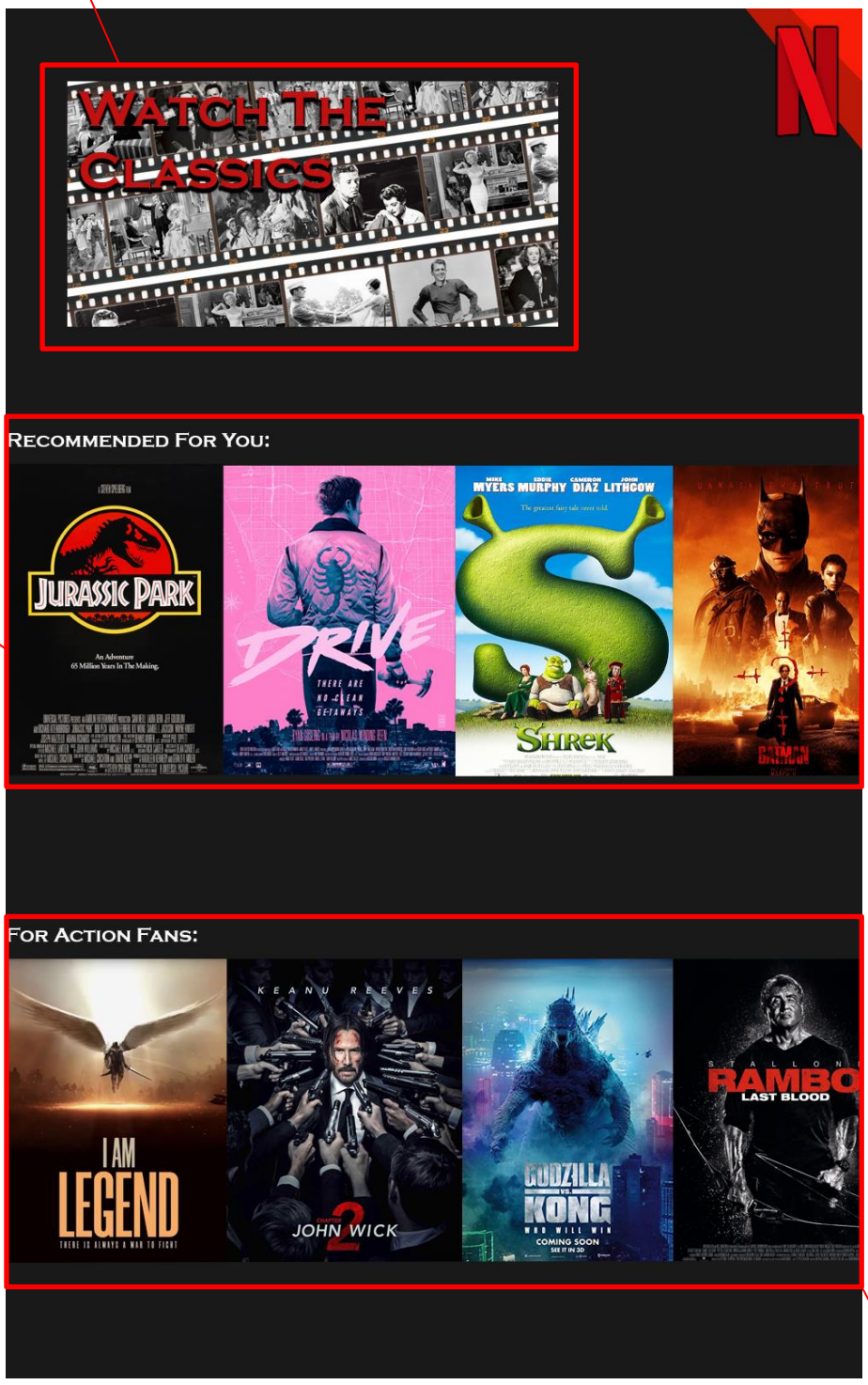
Practical Application

The aforementioned approach could result in a set of rules which could be used in a more complicated recommendations system for film-streaming websites. Of course, for a truly accurate and complex system, the inferred rules would only serve as a basis for a more complex architecture, however by assigning weights to certain rules, ordering recommended films by confidence or lift values, and perhaps adding different algorithms into the system, a robust and accurate recommendation system could be built.

The below visualization is meant to represent the possible use of the previously described method to create association rules in the context of a film recommendation system from the user's point of view:

Genres/types of films recommended on broader relations like in this case consequent related to old films. This could also change based on rules related to days of the week.

Recommendations based on more specific rules with both consequents and antecedents being individual movie ratings.



Recommendations based on association rules induced from genres.

Examples of Induced Rules

First, it is important to note that a lot of the present rules can be considered useful in terms of their application for a recommendation system but not necessarily interesting when trying to find unusual patterns within the data. However some interesting rules have been uncovered:

- Minimum support: 0.1 minimum confidence: 0.6
  Western->Forrest Gump (High)

- Minimum support: 0.1 minimum confidence: 0.9
  Romance, drama, adventure->old

- Minimum support: 0.1 minimum confidence: 0.1
  Silence of the Lambs->Children (Although confidence is relatively low it is still an unexpected rule)

- Minimum support: 0.11 minimum confidence: 0.75
  Drama, old, thriller-> Romance

- Minimum support: 0.1 minimum confidence: 76
  Romance, Shawshank Redemption (High)-> Crime

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction	zhangs_metric
0	(new)	(old)	0.209493	0.260229	0.124386	0.593750	2.281643	0.069870	1.820974	0.710581
1	(old)	(Action)	0.260229	0.194763	0.139116	0.534591	2.744834	0.088433	1.730172	0.859292
2	(Action)	(old)	0.194763	0.260229	0.139116	0.714286	2.744834	0.088433	2.589198	0.789431
3	(old)	(Adventure)	0.260229	0.235679	0.158756	0.610063	2.588531	0.097426	1.960113	0.829555
4	(Adventure)	(old)	0.235679	0.260229	0.158756	0.673611	2.588531	0.097426	2.266532	0.802910
...	...	...	...	...	...	...	...	...	...	...
12701	(Star Wars: Episode IV - A New Hope (1977) hig...	(Lord of the Rings: The Return of the King, Th...	0.162029	0.114566	0.104746	0.646465	5.642713	0.086183	2.504513	0.981873
12702	(Lord of the Rings: The Fellowship of the Ring...	(Lord of the Rings: The Return of the King, Th...	0.153846	0.119476	0.104746	0.680851	5.698630	0.086365	2.758974	0.974432
12703	(Matrix, The (1999) high, Lord of the Rings: T...	(Star Wars: Episode IV - A New Hope (1977) hig...	0.186579	0.126023	0.104746	0.561404	4.454773	0.081233	1.992668	0.953408
12704	(Star Wars: Episode IV - A New Hope (1977) hig...	(Lord of the Rings: The Return of the King, Th...	0.148936	0.117840	0.104746	0.703297	5.968254	0.087196	2.973207	0.978125
12705	(Star Wars: Episode V - The Empire Strikes Bac...	(Star Wars: Episode IV - A New Hope (1977) hig...	0.140753	0.121113	0.104746	0.744186	6.144563	0.087699	3.435649	0.974405

Fig 8. Data Frame showing the output of Apriori algorithm used for inducing association rules for the final, agreed-upon structure of the “MovieLens” data.

Conclusions

In conclusion, a rich, diverse, and interesting set of rules can be created based on the provided datasets. The final attempt at the task of inferring association rules was based on the idea of creating a set of rules which could be used in real-world applications for creating a recommendation system for a film-streaming website.

However, one of the noticeable limitations of the algorithm could be related to the fact that the columns containing ratings for the individual films will end up being very sparse which could lead to rules based on genres(which often overlap) to dominate the space of discovered rules. However, this problem could be addressed in the next stage of development of the more complex film-recommending architecture.

Additionally, movies with high rating seemed to dominate a lot of “movie specific” rules, meaning that perhaps when using the algorithm to find more varied, at least when it comes to ranking, rules high ranking films would need to be considered separately.

The assignment also showed the difficulty of choosing interesting rules especially when trying to find appropriate values of support and confidence thresholds to narrow the search space to association rules that can be considered both interesting and useful when it comes to real-world applicability.

Lastly, the aforementioned real-world use and easy interpretability of induced rule make the proposed method practical and relatively easy to apply for similar datasets. As such, the proposed algorithm is believed to be a preferred way of obtaining association rules for the “MovieLens” dataset.