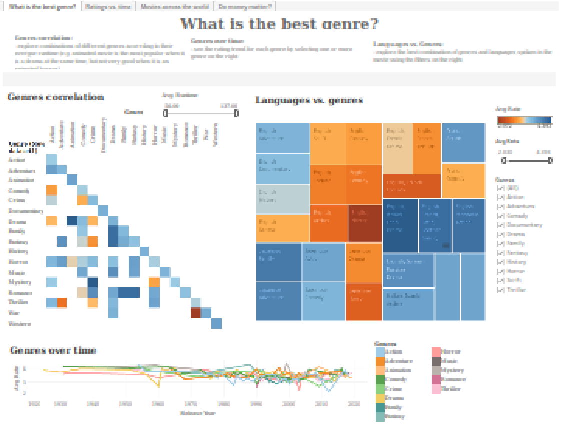
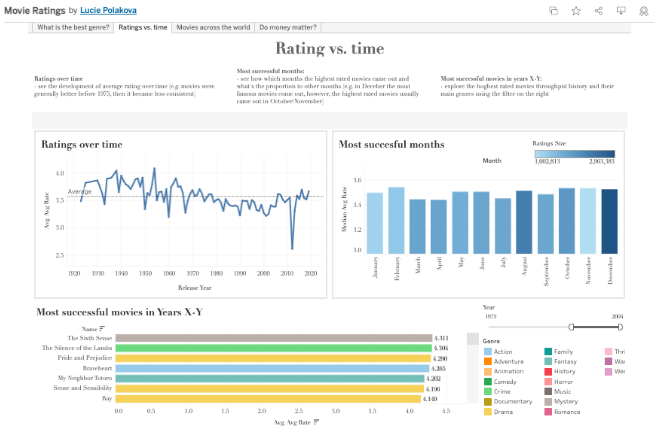
What can we learn from Movie Ratings?

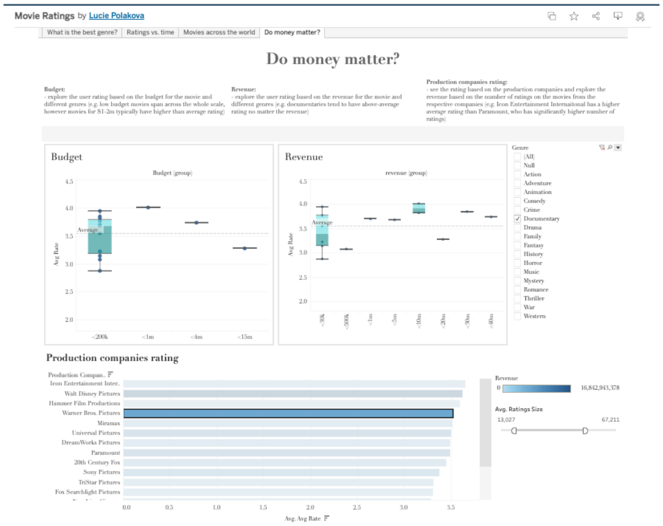
Niloofar Zarif, Deepansha Chhabra. Lucie Polakova,



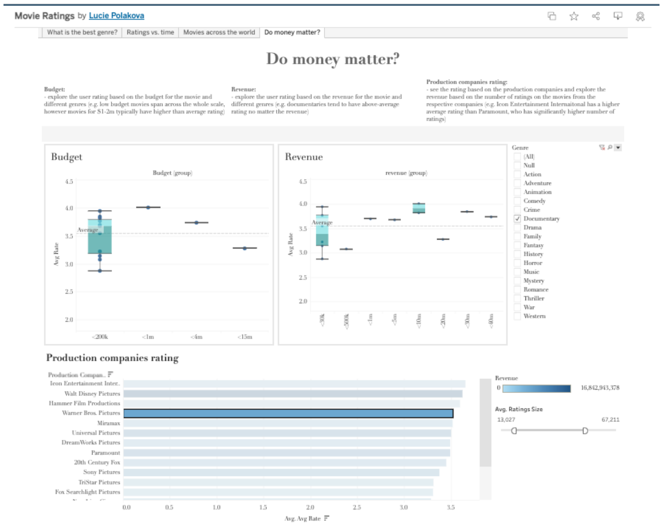
a) Genre exploration dashboard



b) Time dashboard



d) Financial dashboard



c) Geographical dashboard

Abstract—Recommender systems have been developed over the years for different purposes such as movie recommendations. Recommending movies to users is a billion-dollar industry where even small increases in recommendation accuracy and performance efficiency can lead to substantial profit and general business benefits. Some may believe that using machine learning and neural networks is the ultimate solution to building good recommender systems, but in practice we have seen that this is untrue. Recommender experts have recently realized having a good body of knowledge about the data and user behavior is essential to building better solutions. In this project, we are going to explore publicly available movie datasets and visualize the knowledge we extract. We hope the results of this work will provide insight that can be used for designing accurate movie recommender systems.

Introduction

In the past decade, large companies such as Netflix, Amazon, Apple, and Disney have invested a significant amount of resources and efforts into their streaming services [3]. In 2020 the number of subscriptions to online movie streaming services surpassed 1.1 billion and consumers spent around 80$ billion on streaming content, far surpassing the 12$ billion earned by movie theatres [1]. These numbers and records show how important it is to design movie recommendation systems that serve the users as effectively as possible. Designing such systems without knowing user preferences would be a difficult task. However, while different users have different tastes and preferences, there will be patterns in the data that can point towards the most common preference criteria, as well as introduce the temporal or regional user preferences.

The first-ever successful recommendation systems used a number of methods such as collaborative filtering. When an engineer designs such methods, they have to set rules based on which filtering will be done. This means the engineer has to make some assumptions about the data and nature of the problem; however, it is paramount that these assumptions are realistic and backed up by facts Making recommender systems that use content based filtering are similar to the way that a designer makes assumptions about data and the patterns that exist within it.

With the advent of machine learning, scientists started using ML for making recommender models. Deep learning recommender models did not require the designer to make assumptions about patterns in the data or even need to know much about the structure of the data. The neural networks designed for making recommendations could be general-purpose and the engineers could use any dataset they wanted during the training phase; however, this would yield a model which was trained for a specific problem and only work well for that exact one. At this stage, designing flexible deep learning models that can fit any problem once trained with the right dataset seemed like the ultimate solution to recommender systems.

The larger the dataset used with a deep learning model at the training phase, the more details about the problem it can include and the more patterns it can catch. This intrigued engineers to use even larger datasets during the training phase. The outcome was two things: 1. Training the neural networks became time-consuming and resource-demanding. 2. The trained models became larger.

When training became time and resource demanding, many could not easily afford their models anymore. Resource cost is usually not a problem for large companies that have lots of computing and memory capabilities, but individuals and smaller corporations do not have such a luxury. Even large companies which can afford to train huge models, do not necessarily like to as it can seem wasteful.

When trained models become larger, they may no longer be able to fit on edge devices. Doing the inference on a cloud or remote resource may not always be worth it either. Meanwhile, companies like Facebook that do the inference on their own data centers report large amounts of workloads caused by recommender inferences[19]. As a result, Facebook has to spend to decide which posts should be on your Facebook feed. This is a spend they would like to minimize.

A new era in designing recommender models is beginning. The new recommender models suggest using filtering methods to narrow down the possible choices and input the narrowed set t into deep learning models. This helps with making deep learning models smaller and returning them to a manageable size in terms of memory and computing.

A question arises here: How should we do the filtering now? So we are back again at using problem-specific knowledge for our recommendation system design. This means that we need to dive into the dataset again and look for patterns and rules to extract. In this project, we will p respond to this question. We will use visualization as a powerful tool to extract knowledge from the dataset and discover the patterns that may be lying there.

We hope the results of this project can help recommendation system designers in two ways: 1. For the ones who are still interested in offloading the whole problem to neural networks, we can help them to decide what attributes and pieces of data are worth gathering and including in the dataset. This will help them have smaller but more efficient datasets and therefore, make training and running inference cycles on the trained neural network cheaper. 2. For the ones who are going to keep up with the very new trend, we can help them with designing the filtering phases that come before the deep learning model. We can also help them make decisions about what order the attributes should be filtered. The rules used for filtering on earlier stages will yield coarse-grained filtering that ideally rule out the options that are far from the desired candidates. On the other hand, rules that are applied to candidates in later stages are going to use fine-grained filtering to omit the candidates that are not close enough to our ideals. This means the order in which attributes are used for filtering matters.

For this project, we will use Netflix's data to extract knowledge and discover patterns that can be found within the dataset and further used by the target audience. Assuming the rating users give to each movie is the dependent variable, we are going to see how the other independent variables including genre, duration, and country of origin affect the rating. We will seek to identify meaningful patterns in the large body of data we have and will use visualization techniques to present them in a clear dashboard.

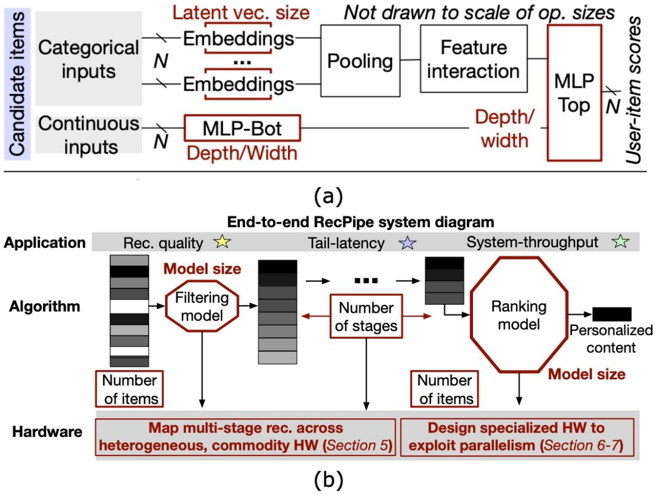
The topic was chosen based on our common interest in recommendation algorithms and the motivation to enhance our understanding of how movie data can be analyzed.

1 Related Work

Before deep learning models took over, researchers were keen to know more about the characteristics of their dataset when making recommendations. This provided them with the ability to design and build recommenders that could give better recommendations based on the nature of the problem and identify the patterns that could be seen within the dataset. Back then we still did not have enough computing power to use deep learning for movie suggestions and instead methods like Collaborative filtering were being used [7][8]. At the same time, Content-Based filtering was a popular method for building recommenders too [4][9][10]. When such methods were being used for movie recommendation, knowing the data and the patterns that existed was important. Therefore, thorough analysis of the data and visualizing it was crucial [11].

As of the mid-2010s we had enough computing capabilities to have deep learning and neural networks in action. Deep learning models for image classification and object detection were being developed rapidly. However, making practical deep learning recommender models was a bit more complicated. But, by the late 2010s, production-level recommender models that were using deep learning came into existence [12][13]. Nowadays, large deep learning recommender models are being used along with older methods, such as collaborative filtering, to produce the best results with high-performance to meet the application level latency agreements [14][15].

For a while people assumed that with enough computing resources, we could give them a large body of data during training and use the model efficiently for making recommendations later by doing inference on the trained model. However, this approach proved to fail. The datasets grew exponentially in size and complexity and it was very important to produce results with high accuracy. Even a 0.01% increase in error can affect the user experience [15]. As a result, the trained models became excessively large, so much so that they could no longer fit in memory. Those models were not practical for online use-cases anymore which is why we see smaller neural networks working along with filtering methods adventing these days [14][15]. Now that filtering is relevant again, analyzing data has once again become important. So once again knowing more about data becomes important when designing these new models. The knowledge extracted from data can be used in the filtering stages of modern recommender models that use pipelined architecture.



(a)

(b)

Figure 1: (a)A common general architecture of the recommender system using only one large model. (b)An example of a pipelined recommender system architecture developed by Meta[15].

Exploring and visualizing movie datasets is not limited to recommender models though. Business analysis or making investment decisions can be another motivation. Computer scientists and visualization enthusiasts may also do it out of curiosity. Some have already done exploratory visualizations on movie datasets that are publicly available [16]. Some of these exploratory visualization projects have used the publicly available Netflix dataset [17] and some use other datasets such as TMDB [18].

With business purposes in mind for visualization, there are projects and papers focusing on profitability and how it relates to different film properties [6]. There are a few solutions where data have been analyzed with the motive of making a contribution towards building a global brand and making a sustainable long-term plan in terms of the production of movies. The relationships between different parameters of the dataset have been shown by scatter plots, bar plots, kernel density estimate plots, histograms box plots, linear model plots, heatmaps, etc., using programming tools like Python Libraries and simple yet efficient tools like Tableau [5]. Based on a few currently available analytical solutions, it has been observed that a particular movie genre may give the highest return per investment but is rated low and hence, in turn, does not bring in high revenue. These analyses have proven to be effective tools assisting the movie production teams to get an insight into the audience's interests. Thus, our idea of identifying the correlations between user rating and other movie properties aims to fill in the gaps in the available research and provide an alternative tool when the aim is not solely financial profit.

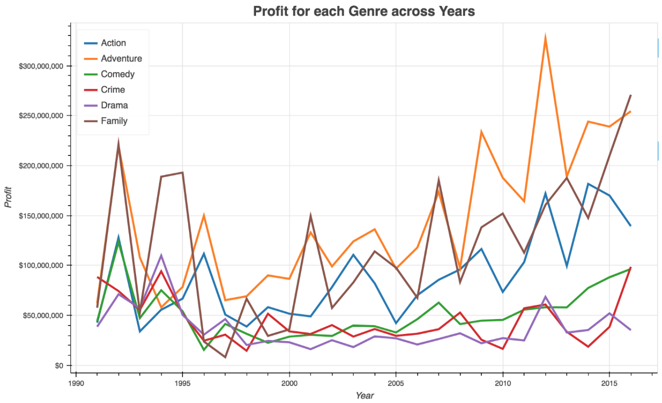


Figure 2: An example of exploratory visualization done on TMDB dataset which does not consider user ratings[6].

2 Data and Task Abstraction

2.1 Data

Data for this project are sourced from publicly available Kaggle datasets and the final dataset was obtained by linking multiple related datasets to gain a wide variety of properties for each attribute. The movie properties were sourced from The Movie Database [20] consisting of 26 thousand movies from the years 1902-2019 and 17 properties, and the Netflix Dataset Latest 2021 [] consisting of 9166 movies and 26 properties. The average rating was obtained from Netflix Movie Rating Dataset containing 26 million items (ratings) on 17.8 thousand movies [21]. While it contained a significant number of ratings by different users, there was no demographic information about the users other than their ID so this did not need to be further filtered and simplified for the purpose of this project.

During the data cleaning and preparation phase, properties were filtered to 16 (properties such as tagline, overview, and poster image were deemed redundant for the purpose of this project). Additionally, after removing movies with incomplete attributes and inconsistent data across the two datasets, we were left with 1,100 movies. Please see the Appendix 1.1 for a detailed table on the data used.

Table 1: What-Why-How-Analysis-Table

What: Data	• Tabular, multidimensional
What: Derived	• Genres (splitting list of genres into multiple strings) • Budget/revenue (form sequential data into categorical - 10-15 groups) • Languages spoken (aggregating all languages into 6 groups: x languages spoken)
Why: Tasks	• Explore movie data set • Interact with idioms
How: Encode	•Heat map, Tree map, Box plots, Bar charts, Line charts
How: Facet	•Multiform, overview-detail
How: Reduce	•Filtering on the applications, filtering on the features
How: Embed	•Instructions on each tab with details (e.g. idiom use
How: Manipulate	•Select, zoom, pan, sort
Scale	•1100 items

2.2 Task

We have deduced two scenarios. While we do not assume that our intended audience has specialized knowledge in movie recommendation models, some level of familiarity with different visualizations is highly recommended to be able to utilize the dashboard's complexity. We expect our audience to participate in tasks related to exploratory data analysis. A machine learning engineer developing a recommendation system might use our dashboard to see how different properties of a movie affects the rating it receives from the users this can include anything from a genre getting higher average rating from the users to movies produced in a country not receiving good ratings from the audience. An owner of an independent movie theater might use our dashboard to identify the most promising movies in terms of the audience ratings as well as to filter through movies based on particular preferences - e.g. movies from a specific era, genre, production company. At the high level, the visualization allows the user to explore the dataset and clearly see the relationship between different properties and user rating. More specifically, the dashboard could help the target users iachieve the following tasks:

- Visualize the overall movie and user rating data in 2D and visually inspect any patterns.
- Evaluate similarities among movies based on selected properties and filters of interest. This could help the ML engineer to determine the main focus of his engineering efforts.
- Interact with the different visualizations. The independent movie theater owner could benefit from tailoring his search of movies based on specific requirements as well as exploring the movies by combining multiple filters.
- Explore movie genres and their importance in user ratings. This could help the ML engineer to explore the rating nuances beyond the statistical patterns. The independent movie theater owner would better understand his customers and preferences.
- Visualize patterns in user rating across time. This would help the users better understand the current user rating and the changing preferences of movie audiences. The ML engineer could also potentially use this to predict future movie ratings.
- Show regional and linguistic differences in user ratings. This could help the users to reflect on the different results and ratings in different environments and encourage implementing such consideration in their project/movie screening.
- Visualize the importance of financial aspects of movies and its impact on user rating. This could help the movie theater owner to focus on smaller, yet more liked, works.

3 Solution

This section introduces the proposed solution that reflects the aforementioned tasks. The final solution of this project consists of an interactive dashboard with an overview of the different movie properties and multiple static and interactive visualizations, each addressing a different property and how it relates to user rating.

Task 1: Visualize the overall movie and user rating data in 2D

From our initial data exploration, we identified multiple themes in the data that would assist in organizing the data for a clear presentation to the user. In order to visualize the variety of movie properties, the solution was split into four tabs based on the data and task themes: *What is the best* *genre?, Ratings vs. time, Movies across the world*, and *Does money matter?*

Task 2: Evaluate similarities among movies based on selected properties and filters of interest.

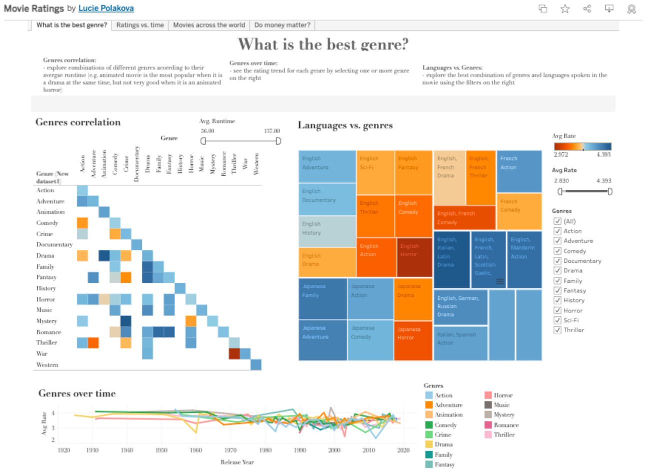
Each tab consists of a short detail for idioms used and their use case providing additional instruction on the potential use of each visualization. Majority of idioms offer additional detail on individual data points (movies) that can be further compared against each other by selecting them in the full view and muting other points for visibility.

Task 3: Interact with the different visualizations.

Interactivity of the proposed dashboard is crucial for delivering value to its target users. Our solution allows users to adjust most of the visualizations based on their preferences. The views adjust based on the filters for genres, user ratings, time period, country of availability, and average rating size.

Task 4: Explore movie genres and their importance in user ratings.

In order to explore the movie genres, a dashboard consisting of a heatmap, a treemap, and a line graph is presented on the first tab (Figure 3). The heatmap provides a compact summary of a quantitative value of user rating with 2D matrix alignment by combination of genres and highlights combinations of genres that yield the highest user rating. The treemap serves to further explore the genres in relation to the languages spoken in the movies. Finally, the multiple line graph shows the long-term trends of individual genres that can be also compared against each other. The filters on each idiom allow the user to interact with and independently explore the data. The consistent use of color as an additional channel of distinguishing different levels of user ratings aids the user to quickly grasp the visualization and the main message.



What is the best genre?

Genres correlation:
- explore combinations of different genres acording to their avergae runtime (e.g. animated movie is the most popular when it is a drama at the same time, but not very good when it is an animated horror)

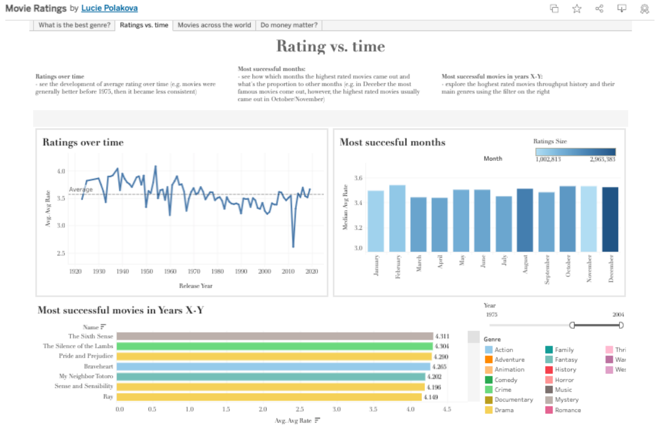
Genres over time:
- see the rating trend for each genre by selecting one or more genre on the right

Languages vs. Genres:
- explore the best combination of genres and languages spoken in the movie using the filters on the right

Figure 3: Genre dashboard

Task 5: Visualize patterns in user rating across time

Second tab presents a dashboard of a line graph and two bar graphs (Figure 4). While the first two idioms are mostly just informative, the second bar graph is again aiming at user interaction and allows to filter and identify the best movies in particular years.



Rating vs. time

Ratings over time
- see the development of average rating over time (e.g. movies were generally better before 1975, then it became less consistent)

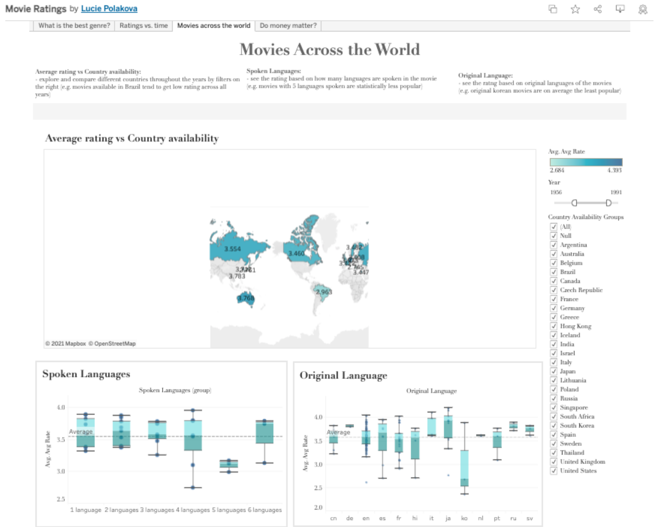
Most successful months:
- see how which months the highest rated movies came out and what's the proportion to other months (e.g. in Deceber the most famous movies come out, however, the highest rated movies usually came out in October/November)

Most successful movies in years X-Y:
- explore the hoghest rated movies throughput history and their main genres using the filter on the right

Figure 4: Time dashboard

Task 6: Show regional and linguistic differences in user ratings.

Tab labeled "Movies across the world" focus on ratings based on geographical location and languages (Figure 5). A map idiom was used to present geographic data of movie availability. Then two box plots provide an overview of the number of languages spoken in the movie as well as the original language affect the ratings.



Movies Across the World

Average rating vs Country availability:
- explore and compare different countries throughout the years by filters on the right (e.g. movies available in Brazil tend to get low rating across all years)

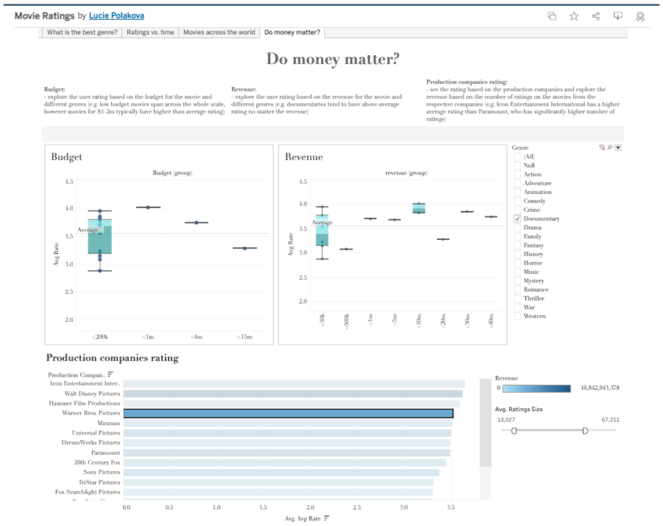
Spoken Languages:
- see the rating based on how many languages are spoken in the movie (e.g. movies with 5 languages spoken are statistically less popular)

Original Language:
- see the ratng based on original languages of the movies (e.g. original korean movies are on average the least popular)

Figure 5: Geographical dashboard

Task 7: Visualize the importance of financial aspects of movies and its impact on user rating.

Final dashboard consists of two boxplots and an interactive bar graph (Figure 6). The boxplots were chosen as the most effective way of presenting multiple statistical information. Interactive filtering allows users to explore budgets and revenues for different genres and see which movies performed above or below average. Since production companies have a large influence on the financial aspects of movies, the final bar graph allows users to explore the different production companies based on the average ratings size and hence pointing towards how big the movies from each respective production company were.



Budget:
- explore the user rating based on the budget for the movie and different genres (e.g. low budget movies span across the whole scale, however movies for $1-2m typically have higher than average rating)

Revenue:
- explore the user rating based on the revenue for the movie and different genres (e.g. documentaries tend to have above-average rating no matter the revenue)

Production companies rating:
- see the rating based on the production companies and explore the revenue based on the number of ratings on the movies from the respective companies (e.g. Icon Entertainment Internaitonal has a higher average rating than Paramount, who has significantly higher number of ratings)

Figure 6: Financial dashboard

4 Implementation

The visualization was implemented using Python libraries and Tableau. Datasets in a CSV format were loaded into a pandas data frame in Python and were cleaned and linked together based on the movie names. An average rating for each movie was derived from the movie rating dataset. The combined datasets were then further processed in Tableau Prep to ensure they were ready for analysis and visualization. All visualizations were designed using Tableau Desktop and the final dashboards were published on the Tableau Public server, accessible as a website.

Design and colors were inspired by other visualizations on the public Tableau server aiming for interactivity and multivariable use. The biggest hurdle was to limit the number of potential idioms used as there could be additional visualizations providing valuable information to the user. Some additional challenges included changing the data types to achieve easier analysis, such as turning a list of genres describing each movie into multiple strings, and the need to create various calculated fields in Tableau to arrive at additional custom properties.

4.1 Breakdown of Work

The following table shows the breakdown of work among the three team members. Please detailed milestones in Appendix 2.

Table 2. Breakdown of work

Work description	Lucie	Niloofar	Deepansha
Design	33%	33%	33%
Data preparation and cleansing	20%	80%	0%
Idioms design	80%	0%	20%
Filters and interactions	90%	0%	10%
Dashboard	100%	0%	0%
Writing (slides+report)	33%	33%	33%

5 Results

As mentioned, there are two main use cases for this visualization: ML engineer and an independent movie theater owner. This section demonstrates how the proposed solution reflects the needs of these two user groups and how it answers the tasks mentioned in section 2.2. Overall, most of these were accomplished, while some could certainly be explored further to improve the overall user experience and reduce cognitive load.

5.1 Use Case 1: ML engineer

As discussed in the related work section, the new trend in the design of recommender systems uses machine learning along with a filtering method. This means the large set of possible candidates needs to be narrowed down using a filtering front-end. The design of the frontend and what tools should be used for doing the task is yet a place of discussion among recommender system experts. But one thing they can agree on is that designing the frontend filtering part, requires context knowledge. Having context knowledge is problem-based so there can not be a general formula for all recommender systems.

An ML engineer who is designing a recommender system can ease the design of the frontend by analyzing the data and discovering patterns. They can feed their data to our dashboard and the dashboard will analyze it and generate the visualizations. The knowledge achieved by the ML engineer through the dashboard can be used for designing filters. For example, the ML engineer can learn which properties of movies have the most effect on ratings users give and which properties have negligible effect (Figure 7). This way the engineer can decide when designing the filters, in which order to use the movie properties for filtering. Because we know the properties that are being used in earlier stages of filtering will be omitting a larger subset of possible candidates. On the other hand, the later our candidates are filtered based on a property, the less candidates will be omitted and therefore the less influence that property may have on the final rated results.

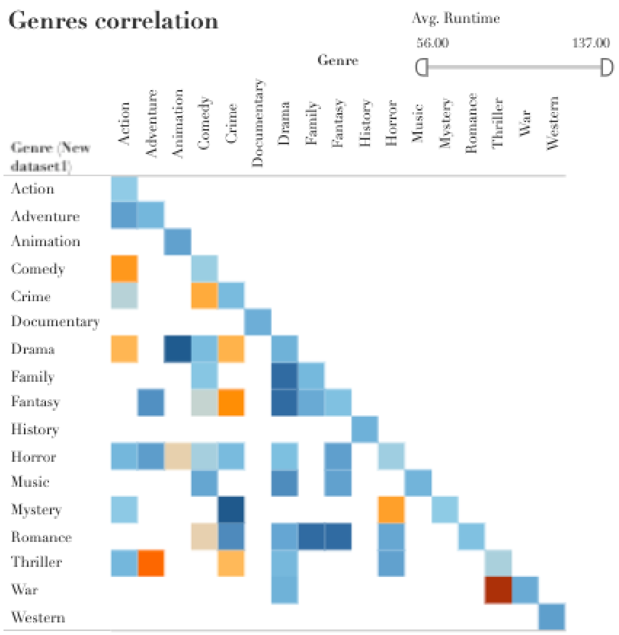


Figure 7: Genre heatmap

5.2 Use Case 2: Independent movie theater owner

Independent movie theaters do not follow the mainstream trends and typically look for more niche movies to screen. While preparing for the upcoming season, an owner of such a movie theater aims to identify movies for specifically themed weeks that would be well accepted by the customers. First, he decided on the genres that should be included in these weeks. By exploring the heat map and treemap, he is able to identify multiple genres and genre-language combinations based on the movie runtime, that would likely lead to a satisfied audience. Furthermore, for the 70s themed week, he explores the Most successful movies in Year's X-Y interactive bar graph by selecting the respective time period on the sliding filter (Figure 8). By exploring the Movies across the world, he might identify the movies to screen based on his geographical location or cultural background of his typical customers. Finally, as he aims to screen mostly less known movies, the final tab allows him to explore movies based on revenue, budget and the production company.

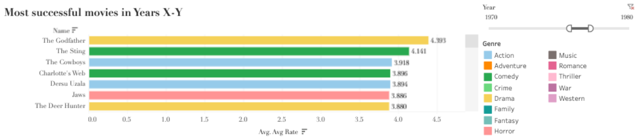


Figure 8: Most successful movies in Year's X-Y

6 Discussion and Future Work

This project demonstrates the visualization of movie ratings with respect to other attributes like movie genres, budget for the movies, revenue generated, original languages, and the other languages a movie was available in. It also visualizes the correlation among different genres based on their appearance together as genres of a movie. Our exploratory dashboard displays all these visualizations and makes it easy for the user to gain insight and analyze the effects of various attributes on movie ratings and make better decisions in movie recommendation system design and movie production.

Tableau is a powerful tool yet there are limitations as well. Generating visualization idioms that are more complicated and can present a high density of information in a brief manner, is not always possible using Tableau and more flexible tools like Python and D3 can be used there. This can increase the complexity of gathering all generated visualizations in a dashboard that is why we leave it to future work.

Our project is not the first one taking an exploratory approach for analyzing movie-related data but as far as we looked, this is the first one that considers user ratings as its center of attention. Our approach was to see the user rating as a goal variable that needs to be maximized hence we discover ways to affect and increase it. This approach goes along well with how the machine learning models used in recommender systems are designed. While training a the neural network, they also see the user rating as the goal that needs to be maximized. This is what separates our project from the similar ones and makes it more suitable to be used by ML engineers.

We did not achieve the expected interactivity in our visualizations. We could include more options for the user to choose from for different visualizations as well as including a wider variety of idioms especially the ones which focus on the interaction of more than one property. But accomplishing this goal only requires some more time on the project and is not far out of reach.

Recommendation systems usually focus on providing personalized recommendations. But in order to be able to do that, you have to have access to user data. As an independent project, we did not have access to any user data because companies do not make those publicly available for privacy reasons. Yet the door is open to parties who have access to user data and have larger datasets for user-movie interactions to come up with more complicated visualizations and consider factors that we had to ignore due to lack of data.

8 Conclusion

It has been observed that the datasets used for training the deep learning models are often quite large, which results in larger models causing the entire training process to become time consuming and resource demanding. This is a problem for smaller companies that do not have enough computing power as eventually they cannot afford such models. Furthermore, larger trained models cannot fit on edge devices which is not worth either. To solve this problem, new recommender systems make use of filtering techniques. For filtering, they need to understand the patterns and extract meaningful rules from the data. We have come up with this solution which will help analyze the data and extract meaningful information from it with the help of visualizations. For those, who still want to train neural networks, we can help them in deciding what pieces of information or attributes to include in the dataset. For those, who want to follow the new trend for trained models, we can help them in deciding on the attributes for filtering. Therefore, our dataset consisted of various attributes like movie name, release year, average rating, rating size, budget, revenue generated, runtime, genres, production companies, production countries, and languages spoken. We visualized the trends of average ratings with all other attributes which helped us analyze how the average ratings are affected by other attributes. We extracted some useful results such as the genre that received the highest ratings in the past years , the effect of revenue generated on the movie ratings etc. We could explore larger datasets and create more complex visualizations and understand the trends in a more detailed manner as part of our future work.

Acknowledgments

We would like to express our gratitude to the University of British Columbia for providing us the opportunity for learning an important and versatile course - Information visualization. The knowledge we have gained through this course can be utilized in various areas. We want to express our sincere gratitude towards Prof. Dr. Tamara Munzner for insightful comments, patience, helpful information, and practical advice throughout the course.

References

[1] Mendelson, Scott. “Streaming Subscriptions Top One Billion: A New Normal, Or A Temporary Disruption?” Forbes, 2021. Forbes, https://www.forbes.com/sites/scottmendelson/2021/03/18/streaming-tops-one-billion-subscribers-a-new-normal-or-a-temporary-disruption/?sh=37e626c8465a. Accessed 18 10 2021.

[2] Kats, Rimma. “Netflix statistics: How many subscribers does Netflix have?” Insider Insights, 2021, https://www.insiderintelligence.com/insights/netflix-subscribers/. Accessed 18 10 2021.

[3] Stodart, Leah, and Ashley Keegan. “Best streaming sites for movies.” Mashable, 2021, https://mashable.com/roundup/best-movie-streaming-sites. Accessed 18 10 2021.

[4] Sunilkumar, Chaurasia Neha. "A review of movie recommendation system: Limitations, Survey and Challenges." ELCVIA Electronic Letters on Computer Vision and Image Analysis 19, no. 3, 2020: 18-37.

[5] Lee, Jeremy. “Exploratory Data Analysis With Movies - Towards Data Science”. Medium, 2020, September 8 https://towardsdatascience.com/exploratory-data-analysis-with-movies-3f32a4c3f2f3. Accessed 20 10 2021

[6] Panchal, Kishan and Swalin, Alvira. “MSDS 622 Final Project”. GitHub, 2021, August 19. https://github.com/k7p/dataviz_project. Accessed 20 10 2021

[7] Subramaniyaswamy, V., R. Logesh, M. Chandrashekhar, Anirudh Challa, and V. Vijayakumar. "A personalised movie recommendation system based on collaborative filtering." International Journal of High Performance Computing and Networking 10, no. 1-2 (2017): 54-63.

[8] Wu, Ching-Seh Mike, Deepti Garg, and Unnathi Bhandary. "Movie recommendation system using collaborative filtering." In 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), pp. 11-15. IEEE, 2018.

[9] Pazzani, Michael J., and Daniel Billsus. "Content-based recommendation systems." In The adaptive web, pp. 325-341. Springer, Berlin, Heidelberg, 2007.

[10] Soares, Márcio, and Paula Viana. "Tuning metadata for better movie content-based recommendation systems." Multimedia Tools and Applications 74, no. 17 (2015): 7015-7036.

[11] Ahmed, Adel & Batagelj, Vladimir & Fu, Xiaoyan & Hong, Seok-Hee & Merrick, Damian & Mrvar, Andrej. (2007). Visualization and analysis of the internet movie database. Asia-Pacific Symposium on Visualization. 17-24. 10.1109/APVIS.2007.329304.

[12] Naumov, Maxim, Dheevatsa Mudigere, Hao-Jun Michael Shi, Jianyu Huang, Narayanan Sundaraman, Jongsoo Park, Xiaodong Wang et al. "Deep learning recommendation model for personalization and recommendation systems." arXiv preprint arXiv:1906.00091 (2019).

[13] Cheng, Heng-Tze, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson et al. "Wide & deep learning for recommender systems." In Proceedings of the 1st workshop on deep learning for recommender systems, pp. 7-10. 2016.

[14] Moreira, Gabriel de Souza P., Sara Rabhi, Ronay Ak, Md Yasin Kabir, and Even Oldridge. "Transformers with multi-modal features and post-fusion context for e-commerce session-based recommendation." arXiv preprint arXiv:2107.05124 (2021).

[15] Gupta, Udit, Samuel Hsia, Mark Wilkening, Javin Pombra, Hsien-Hsin S. Lee, Gu-Yeon Wei, Carole-Jean Wu, and David Brooks. "RecPipe: Co-designing Models and Hardware to Jointly Optimize Recommendation Quality and Performance." arXiv preprint arXiv:2105.08820 (2021).

[16] Singhal, Shashank. “Data Visualization -- Netflix Data set”. Medium, https://medium.com/analytics-vidhya/data-visualization-netflix-data-set-d4fa2da97253. Accessed 15 11 2021

[17] Rastogi, Kashish. “Performing EDA on Netflix Dataset with Plotly”. Analytics Vidhya, https://www.analyticsvidhya.com/blog/2021/09/performing-eda-of-netflix-dataset-with-plotly/. Accessed 15 11 2021

[18] Panchal, Kishan. “Exploring Movie Data with Interactive Visualizations” . Towards Data Science, https://towardsdatascience.com/exploring-movie-data-with-interactive-visualizations-c22e8ce5f663 . Accessed 15 11 2021

[19] U. Gupta et al., "The Architectural Implications of Facebook's DNN-Based Personalized Recommendation," 2020 IEEE International Symposium on High Performance Computer Architecture (HPCA), 2020, pp. 488-501, doi: 10.1109/HPCA47549.2020.00047.

[20] Cochran, David, “The Movie Database, 26K+ Movies from TheMovieDB.org, 1902-2019”, Kaggle, 2021, https://www.kaggle.com/davidcochran/the-movie-database-19022019. Accessed 18 10 2021.

[21] Javia, Rishit, “Netflix Movie Rating Dataset, Dataset from Netflix’s “Netflix Prize” competition”, Kaggle, 2021, https://www.kaggle.com/rishitjavia/netflix-movie-rating-dataset?select=Netflix_Dataset_Movie.csv. Accessed 18 10 2021.

Appendix 1 - Data abstraction

Categorical	**Ordered**
Original language: 9	Budget: 0-400,000,000
Genre: 19	Revenue: 0-1,900,00,000
View rating: 9	Year: 1930-2020
Release month: 12	Rating: 1-5
Languages spoken: 26	Runtime: 70-210
Original languages: 10	Number of votes: 1-5,639

Appendix 2 - Milestone overview

Milestone	Components	Estimated hours	Actual hours	Estimated date of completion	Actual date of completion
Preliminary Work	Prepare project pitches, exploration and discussion on data source	4 (All)	5	Oct 13	Oct 12
	Meetings to discuss project direction	5 (All)	7		
Project Proposal	Our first step is choosing the best dataset we can find and clean it such that it will be ready for our further use	5 (All)	5	Oct 21	Oct 21
	Develop and edit project proposal	15 (All)	16		
Preliminary data exploration, data preparation and vis finalisation	Cleaning data and linking relevant datasets to create a final dataset for analysis.	15 (Niloo)	20	Nov 10	Nov 13
	Aggregation of rating records based on movie name	15 (Niloo)	12		
	Further exploration of vis options, finalizing the visualizations used, sketching/designing low-fidelity prototypes	15 (Lucie and Deepansha)	13		
	Group discussion of prototypes and design finalization	2 (All)	2		
Update - Create first version of vis software using finalized idioms	Knowledge extraction, preparing data in Tableau Prep, preliminary statistical analysis, seeking meaningful patterns and useful knowledge	4 (Lucie)	6	Nov 16	Nov 16
	Finalize related works section for the final paper	8 (Niloo)	10		
	Begin data analysis and correlation overview in Tableau.	10 (Lucie and Deepansha)	10		
	Write an update report.	10 (All)	10		
Visualization Design	Design 6 idioms on movie properties	15 (Deepansha)	15	Nov 24	Nov 28
	Design 10 idioms on movie properties	20 (Lucie)	30		
Finalize Viz Dashboard	Final refinements and bug fixes	20 (All)	18	Dec 3	Dec 12
	Developing other features of the vis and final dashboard (multiple views together with chosen idioms, interactivity to support selection/filtering)	20 (All)	20 (Lucie)		
Final Presentation	Slides	10 (Deepansha)	10 (All)	Dec 15	Dec 15
	Rehearsal, demo	3 (All)	2		
Final Paper	Documenting and reporting the results, finalization of the paper, formatting and citation check	20 (All)	25	Dec 17	Dec 17