Data Visualization
COM-480 Project Milestone 3

ActorsGalaxy - Process Book

Authors:
Hugo Casademont
Cezary Januszek
Marc Odermatt

June 3, 2022

École Polytechnique Fédérale de Lausanne

1 Project idea

1.1 Dataset

The dataset chosen for our project was the Movies Dataset from Kaggle: `https://www.kaggle.com/datasets/rounakbanik/the-movies-dataset`. In this dataset we could find metadata, taken from IMDB, TMDB and GroupLens websites, for 45'000 movies released until 2018. For the need of our project we focused mainly on the data provided in the credits and movies-metadata csv files.

In the data pre-processing step we also restricted ourselves to keep only actors that played in more than 20 movies - that resulted in considering still almost 4'000 actors.

1.2 Project goal

The goal of our project was to create a website for people to discover their favorite movie actors throughout interactive representations. There already exist many possibilities to discover new movies through other movies, like the recommender systems implemented by streaming platforms (e.g. Netflix). However we wanted to focus on the actor side of this movie universe and allow people to discover connections between actors they never thought about and movies they played in together. That is why we created *ActorsGalaxy* - a unique website to learn more about your favorite actors and find the best movies they played in.

2 Website description

On our website the user can discover everything he didn't know yet about his favorite actors. He can start discovering our actors universe in two ways: either by exploring mainly through the connection graph starting from an actor and trying to reach the one he looks for or if he's less patient just type the name of the actor in the search bar in the right top corner of the website.
Our website is all about visualizing the actors data in an exciting way and discovering either new facts about actors or new movies you never heard about. To provide such an experience we be split the website into 3 main visualization parts:

2.1 Actors connection graph

The connection graph is the core of our project. It represents the connections between a chosen actor and its closest 20 co-actors. We measured the closeness between the actors by the number of movies they played in together. In our graph each actor is represented by a node and every connection with an edge between two actor-nodes. Each connection has also a weight associated to it depending on the number of movies the actors share. The weights are visualized by the thickness of the edge and you can discover the title of common movies corresponding to each connection, by hovering the mouse over an edge.

You didn't know that Tom Cruise and Alec Baldwin played together in *Mission Impossible: Rogue Nation*? Well, now you know it!

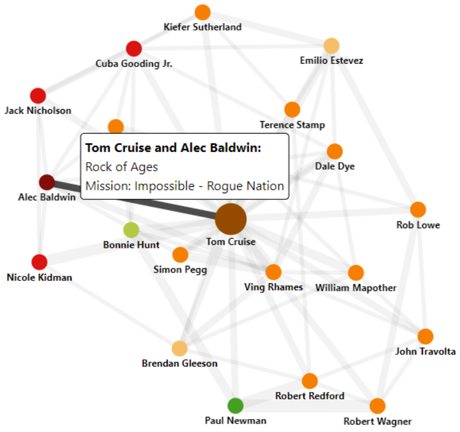


Figure 1: Tom Cruise connection graph

2.2 Actor's movie and genre treemap

Besides discovering just the connections between actors it would be interesting to also learn a bit more about the actor himself. That's why the other side of the website is the actor's profile containing the actor's movie and genre treemap and some of the actor's statistics retrieved from our dataset.

We decided to represent the actor's movie portfolio and genre diversity using a treemap. Each square is associated to a movie and its size depends on the movie budget. The color of the movie square indicates the main genre of the corresponding movie. This visualization gives us a broad idea of the biggest movies the actor played in and the main genres of those. The legend above the treemap explains the colormap we choose to represent all the genres present in an individual actor's movie portfolio. We also decided later to put the color of the main genre associated to an actor in its node in the connection graph. When hovering over the treemap, you can learn more about each movie from the tooltip that contains the movie poster, its release date, budget, revenue and rating.

2.3 Actor's metrics

To complete the actor's profile, we included some statistics about him/her. Those are the number of movie releases per year, the number of total movies, the average rating of his/her movies, the average budget and the average revenue. An actor's profile would also not be

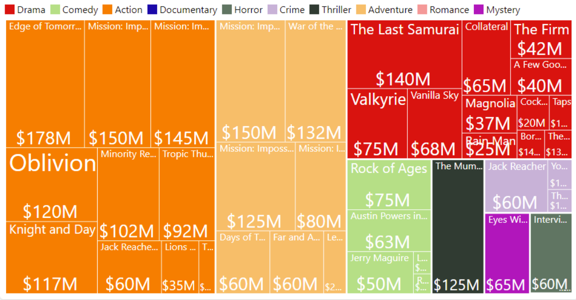


Figure 2: Tom Cruise movie and genre treemap

complete without the actor's picture. All these metrics, except the movie releases, are compared to the average for all the actors in our dataset to have a better idea of the actor's performance and popularity.

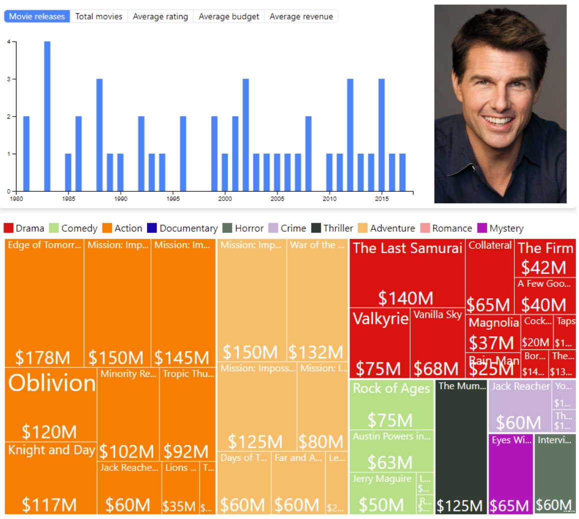


Figure 3: Tom Cruise complete profile

3 Visualizations development

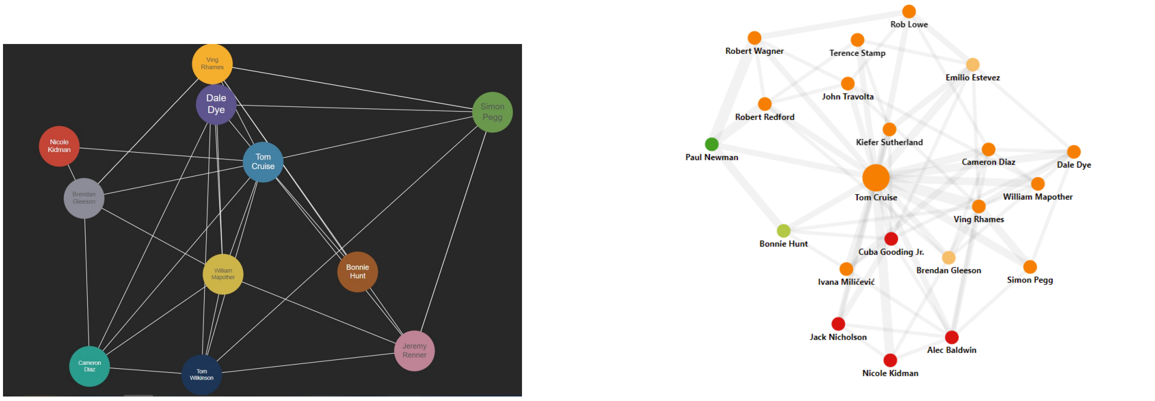
We developed all our visualizations using D3.js and Tailwind CSS to efficiently design our website. We also used Fuse.js for the search bar. Before obtaining the results broadly presented in the previous sections we of course had to start from some basic ideas and sketches, some of which we have already presented in Milestone 2 of this project.

3.1 Connection graph

To build the graph as it is now, we started by creating a simple static version of it, using the D3plus library. We tested several numbers of co-actors nodes to show on the graph to provide as many co-actors and connections as possible by still making it readable and clear for the user. In the end, we decided that 20 nodes is the optimal number and that's what we kept for the later steps. As we wanted to add a lot more interactivity in our graph we saw that D3plus is limiting the possibilities we have, so we switched to a D3 implementation of a force-directed graph. This kind of graph is a commonly used layout for visualizing connections between nodes in a network as it is an aesthetically-pleasing and efficient representation. With such a graph the user can move the nodes around with all the other nodes and connections moving accordingly to it. Another advantage of it is that the main actor node connected to all the other nodes will always remain in the center of the graph, which makes it more intuitive,as we want to focus on a central actor for each of such a graph.

We were also able to customize the edges representation, by adapting their thickness to their weights (number of movies in common) and by adding a tooltip for each of them to show the title of those movies. Later on, we came to a conclusion that adding a tooltip on each node with the actor picture could be helpful, as it would not force the users to always check the actor profile to make sure they know who he/she is. As the last feature, came the color of the actor node associated to the color of the most present movie genre in his portfolio (to be discoverd in the treemap).

Below you can see the development of our graph on pictures:



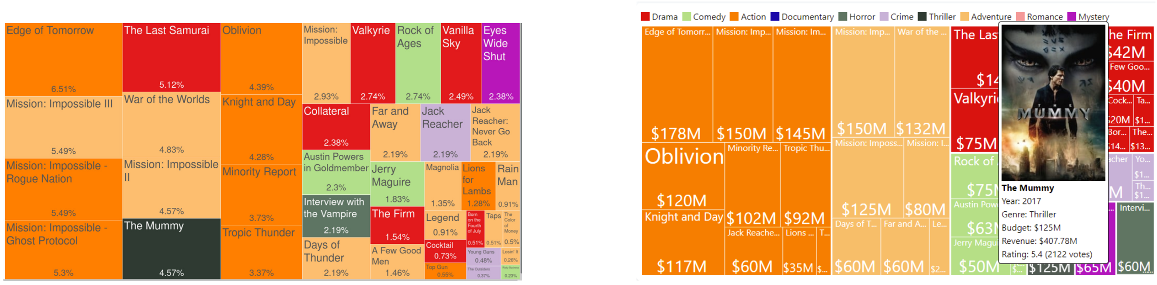
(a) Graph sketch for Tom Cruise and his 10 closest co-actors

(b) Force-directed graph for Tom Cruise and his 20 closest co-actors

Figure 4: Graph sketch in D3plus vs final graph in D3

3.2 Treemap

The treemap was also created at first using D3plus, before changing to D3 for the same reasons we had with the graph visualization. In our first prototype we already defined the movie square sizes to be dependant on the movie budget. In this way we were able to clearly represent the biggest movies in an actor's portfolio. At first we were showing the percentage of the total budget of all actor's movies in each square, but later we changed it to simply the movie budget itself for each window, as it was more intuitive. As we wanted to show the movie genres in this representation as well, we had to experiment a bit to find the most appropriate color palette for each genre. In this way the color of a window in the treemap was indicating what is the main genre of the corresponding movie. To allow the user to clearly understand our choices we add a legend on the top of the treemap to which one can refer to check the genre of the movie. The lest improvement added to this part was the creation of a tooltip for each movie square. In this tooltip we added the movie name, its poster, release date, genre, budget, revenue and rating (with the number of votes associated to it).



The Mummy
Year: 2017
Genre: Thriller
Budget: $125M
Revenue: $407.78M
Rating: 5.4 (2122 votes)

(a) First treemap prototype

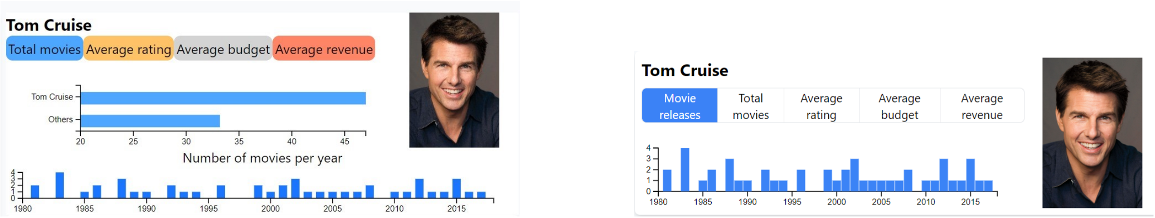
(b) Final treemap with movie tooltip

Figure 5: Teemap evolution

3.3 Metrics bar charts

The actor's metrics is maybe the part that experienced the most changes and improvements. In the beginning the idea was pretty simple - we wanted to include some metrics for each actor to show the number of his movies, the average rating of those, the average revenue, budget and the total budget and revenue as well. Quickly, we dropped the total budget and revenue, as we did not find these information very interesting. We also wanted to represent a time bar chart with the number of movie releases per year. Lastly an actor's picture would complete the profile.

After having a first version of this metrics part with the picture, text metrics and the time bar chart below them, we came to a conclusion that visualizing the number of movies, average rating, average budget and revenues with comparison to the average of all actors would be nicer that just having hard text information. To do this we had to rearrange a bit the space we had left for these statistics and decided to use buttons to change from one bar chart to another. We then visualized the four comparison metrics using a horizontal bar chart for each of them. Adding a tooltip that appear when hovering the mouse over a bar, made then more readable as the tooltip shows the precise value for the metric.

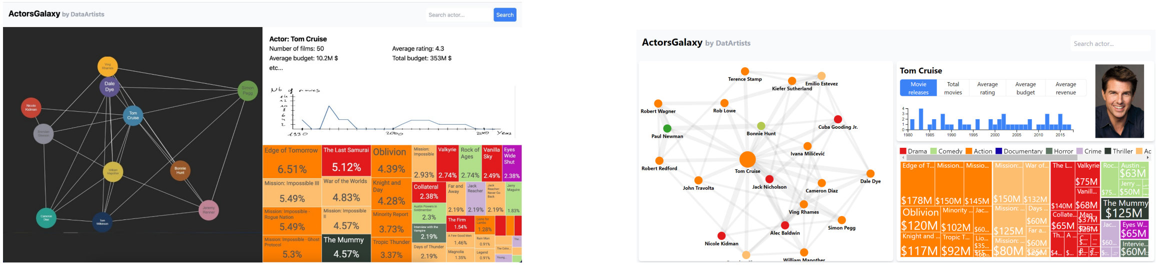


(a) Early version of comparison metrics

(b) Final version of comparison metrics

Figure 6: Actor's metrics evolution

Finally, lets see how our website changed compared to the prototype version we submitted for Milestone 2.



(a) First website prototype

(b) Final website

Figure 7: Website evolution

4 Challenges

The first idea we had was to implement a graph that would represent all the actors in our dataset and their interconnections in a large graph. However, we quickly understood that such an approach might not be the best, as it was difficult to directly scale the whole graph to a global view, making it at the same time readable and comprehensive. Another issue we were concerned about was that we could have discovered some clusters that are almost disconnected from the rest of the graph and would have to be addressed separately. Instead we decided to focus on a subgraph proper to every actor.

Another major challenge we faced was that the url paths to the images provided in the dataset were not valid, so we had to find another way to get the poster and actor images.

In the end we wrote a scraper that provided us valid paths to images that we stored in json files with other data for each actor in our dataset. With the poster and actor images we were able to complete the actors profiles with movie tooltips and profile pictures.

5 Peer assessment

Hugo:
Main frontend developer, responsible for the network graph and improvements in all parts.
Marc:
Treemap expert, movie genre specialist, webscraper and screencast presenter.
Cezary:
Bar chart lover, responsible for the actor's metrics and main process book writer.

All members of the group were equally involved in the final conception of the project, the data preprocessing and idea sharing for the first two milestones.