
Title
A deep dive into what makes a great movie
Creators
Maja Stamenkovic
Jeremy Mion
Johan Barthas
Professor
Laurent Vuillon
Date
May 2020
Course
COM - 480
Institution
EPFL

Introduction

Nowadays, **movies** are part of everyone's life. Whether you're young or old, you prefer watching your favorite movie at home or the theater, alone or with other people. You might like watching movies just for the sake of **fun** and **relax**, and see that as a **bonding** activity with your friends and family. Movies might be your way to temporarily **escape** from real life and problems that surround you. Or you might be more serious movie lover, which appreciates movies as an **art** form and always tries to learn **lessons** and acquire **knowledge** and **inspiration** from them. Whichever group you belong to, **you will find something intriguing in our visualization**.

Our goals

Determine patterns of what makes a great movie: We want to enable the users to make their own conclusions and discover patterns among movies. By visualizing the data we have for each movie, we give the user the opportunity to interactively explore and make connections, using their own logic and creativity inspired by our representation of the data.

Visualize connections that exist among the movies: The key factor for success of the movies are the actors playing in them. We would like to visualise how the actors are connected and which group of actors tend to work together more often.

Create an entertaining experience on the website: Make the whole experience of reading and consuming our data story engaging, interactive and visually appealing.

Is this visualization for you?

If movies are just visual entertainment for you: you will appreciate elegant and interactive visualization we create which will give you a broad overview of the movie industry in the past century. Moreover, you will discover many trivia and fun-facts that you can share with your friends.

If you are a real film aficionado: you will be overwhelmed with the possibility to get into the core of the most popular movies and the complex relationships that exist between them - from genres, production houses, budgets and revenues to connections between actors.

Design

Dataset

We had the idea to explore movies. The Kaggle TMDB dataset published in 2016 seemed to be the ideal dataset to use. It contains information about the movies such as revenue, popularity, title, overview, release date, genre, production companies, keywords and popularity… We then explored the options that TMDBs API offered with regards to accessing more details for specific movies.

Knowing what data was present in our dataset and what was possible to access through TMDBs API allowed us to brainstorm potential ideas for visualisations to make.

Brainstorming

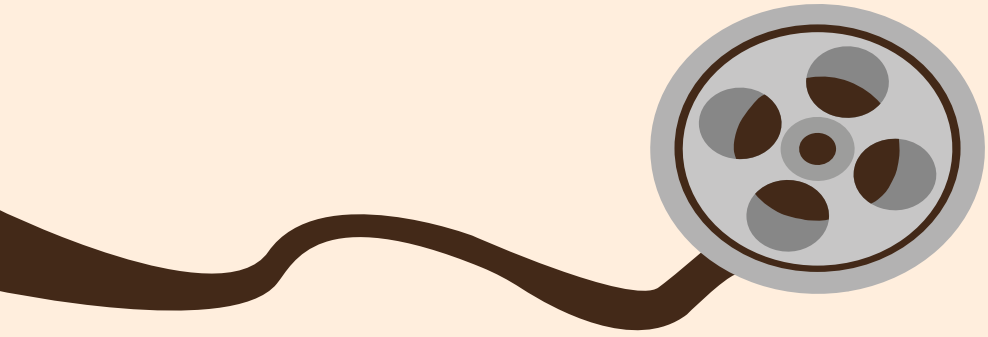
When we were sure which information about the movies we can get, we needed to select the key dimensions on which we will base our analysis and visualization. The key attributes of the movies that we decided to analyse are release date, genre, production house and actors that played in the movie. We had to give up on many interesting movie features like keywords, movie length and spoken languages. Although we knew that removing these attributes from the analysis will hide many interesting insights about the history and the evolution of the movie industry, we know that overwhelming the user with all movie features would actually make it harder to get to the conclusions by themselves.

Choosing the visualizations

We had ideas for several types of graphs: Bubble chart, Dependency wheel, Area Chart, Network graph, Sequence Sunburst… We know that choosing the visualisation type must be done carefully and we wanted to know our options. We also tried to have variety in the visualisations, so each one is related to different data relationships. We decided to do three visualizations:

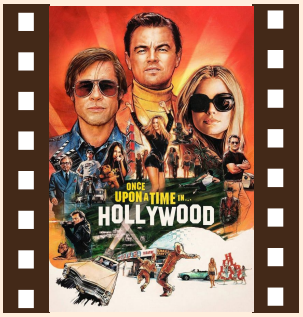
- The first one is the **bubble chart**, which is used for representing the correlation between revenue, popularity and budget of the movie.
- The second one is **100% stacked area chart**, which shows change-over-time, and with **additional streamgraph** adds part-to-whole relationship between genres and production houses.
- The final one is a **network**, which is used for showing the interconnectedness of actors based on the movies in which they played.

In the end, we managed to find a balance between the recommended ways to visualize the data relationship we wanted to show and our own ambitions and interests.

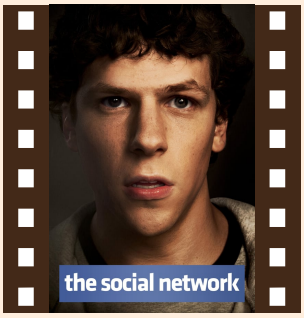


What are we visualizing? The initial challenge was to pick which information about the movie should we visualize. We decided to settle on visualising the difference between a revenue, production budget, and popularity. The revenue and popularity will be on the x and y axis respectively. The budget will be the area of the bubble. Note that the axis use a log scale, this allows for better visualisation of the movies with a low revenue or popularity. The colors will represent the genre and will match the timeline chart below it.
How is it connected to our goals? This visualization allows for elegant detail visualization of movies with providing some interesting insight on what makes a good movie. It's primary role is to increase user engagement and introduce them to the movie story.

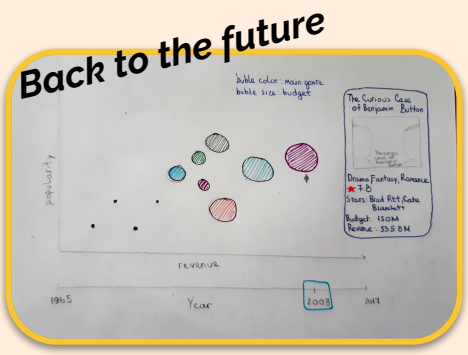
What are we visualizing? We decided to do a deeper analysis on genres and production houses and show how they evolved over time. In 100% stacked area chart the y axis shows what is the percentage of movies produced in a particular year that belong to each genre or production company, while the streamgraph shows the number of movies per year for each category for a selected genre or production house.
How is it connected to our goals? This visualization is meant for users to get insights and discover patterns about the trends in the movie industry. Users can see which genres were the most popular throughout the previous 50 years and which production houses were most successful, as well as recognize the connections that exist between genre production and movie companies.

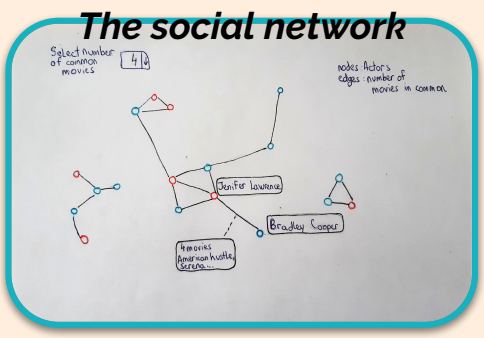


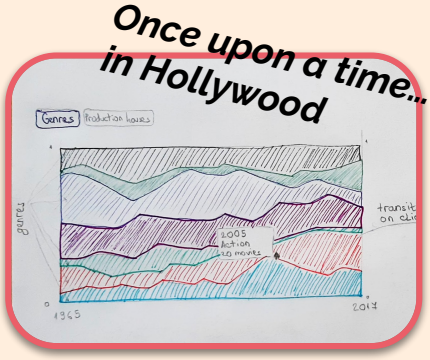
What are we visualizing? We wanted to show the connectedness of actors given by the number of common movies they played in. Therefore we decided to set the nodes as the actors and the edges as the common movies the connected actors played together. To make the network insightful we used tooltips both for nodes and edges.
How is it connected to our goals? This visualization allows the user to play around to find clusters of actors given by the movies they played in so that the user can get a better idea of why some actors have a lot of movies in common (e.g.: trilogy, …) We decided to use a force-directed layout to make the experience enjoyable and interactive.



In order to keep the spirit of the topic in our work, we decided to **name our visualizations** using some of the popular movies which seemed adequate for our graphs. For the visualization of the movies across years we decided to go with "**Back to the future**", the evolution of popularity of genres and production houses is a modern blockbuster "**Once upon a time... in Hollywood**", and the network of actors is literally "**The social network**".



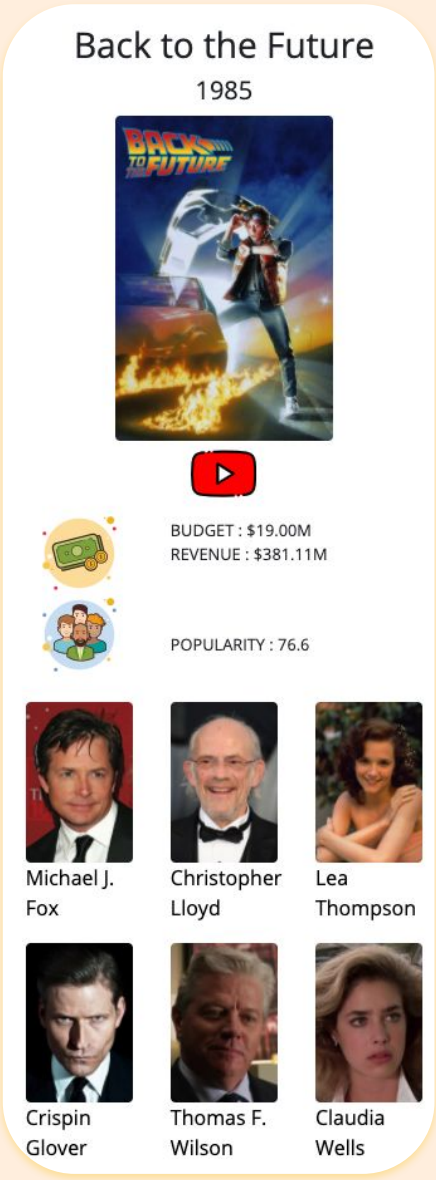




Initial sketches of the graphs and their names

Chart type: Bubble chart
Main idea: Visualize all the movies popularity, budget, and revenue through years
Technologies: d3.js, jQuery-UI

"Back to the Future" chart



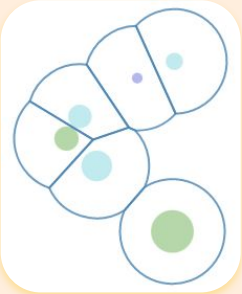
Detailed movie info

Inspiration: The bubble chart was inspired by the incredibly talented Hans Rosling, who co-founded the Gapminder data exploration tool. Gapminder is an incredible tool for data exploration but not that much for a nice user experience to visualize data. We needed to restrict interactivity to key dimensions.

Design evolution: The design changes were extra features that we decided to add such as the play button for the timeline, youtube link to trailer, headshots of actors, clickable legend for genres colors, and search and autocomplete of movies names.

Main Challenge: The main difficulty for this graph was collecting and linking many **different data sources**. Our initial data source came from Kaggle TMDB data dump. This data is sufficient for rendering the bubble chart. This is important because there is too much data to fetch from TMDB since the API only offers single movie queries. To get the missing information such as the cast, youtube link, pictures,... JavaScript code had to be written to make the requests and fetch the appropriate data. The difficulty with this task is to choose relevant information to share with the user while maintaining a great user experience.

Other Challenges: Selecting movies with small budget can be complicated if you need to hover on top of the bubble. To **improve the user experience** we decided to help the user select movies. To do so we created a **Voronoi diagram** with a maximum radius. On the figure on the right you can see the selection zones that will trigger the movie info to be displayed. Since these selection zones are larger than the small data points it improves the user experience of exploring movies with small budgets. Using a Voronoi diagram allows the borders to be positioned in a natural way that feels **intuitive** for the user.



Voronoi diagram

Chart type: 100% stacked area chart + streamgraph
Main idea: Visualize the popularity of genres and production houses through years
Technologies: d3.js

"Once upon a time... in Hollywood" chart

Red: danger, violence	**Thriller**
Pink: femininity, beauty	**Romance**
Blue: calm, depressed	**Drama, Music**
Purple: mysterious, cruel	**Horror**

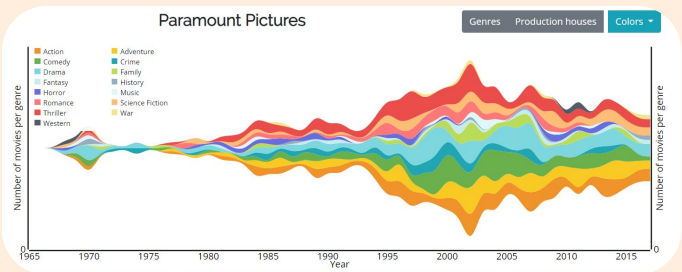
Examples of colors, emotions and associated genres

Inspiration: To visualize how the number of produced movies changed for the 15 most popular genres and 10 most popular production houses we found a big inspiration in the visualization of another type of art - music. The timeline perfectly explains how music genres evolved over time, and the interactivity enables users to get even more details on the genres they find interesting.

Design evolution: Initially our idea was to visualize the genres and allow users to have a detailed view by decomposing the genre production by production houses. After implementing it we realized that the reverse decomposition would also be an interesting feature. During the implementation, we also decided to include the option to choose between the default and colorblind-safe palette. From the initial ideas that we dropped, we agreed that making the complex transition between the two views didn't bring enough benefits and we decided to go with simpler, fading transition.

Main Challenge: One of the most difficult aspects for this plot was **choosing the colors**. We have 15 distinct genres and it is challenging to fit **15 distinct colors** on the fully colored plot like 100% stacked area chart in a way which is visually appealing. Furthermore, most online solutions for color choices include up to 12 colors in a palette. To make this color picking problem even more difficult, we ambitiously wanted to not only find a color palette and ordering that would look nice, but to also make genre-color combination adequate using a **color psychology concept**. - keeping in mind that colors can affect us emotionally, psychologically and even physically, often without us becoming aware. Color choice was very important for us: e.g. pastel colors would not be the right pick for the cinema topic. In the end we decided to use Aussie palette, which we enriched with 3 more colors.

Other Challenges: Preparing the data in a proper manner for the plot: even a small mistakes in the data format can lead to completely random graphs and the problem is hard to debug.



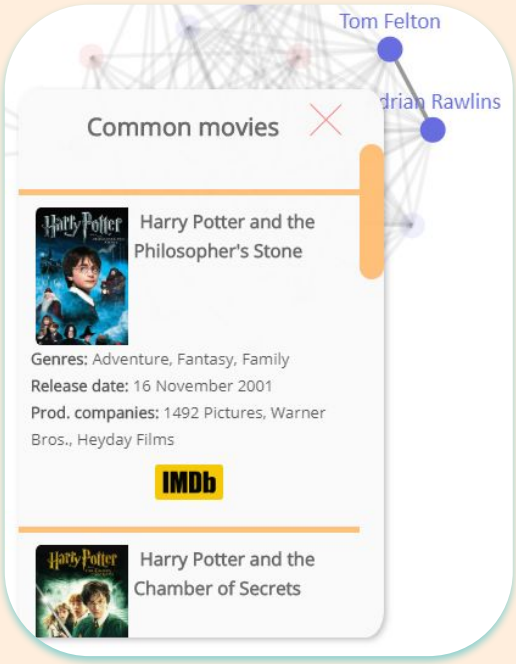
Streamgraph with 15 different colors

Chart type: Force-directed layout
Main idea: Visualize actors that played together in several movies
Technologies: d3.js, canvas

"The social network" chart

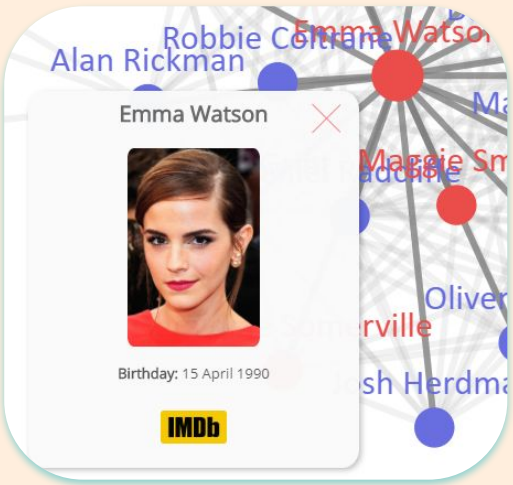
Inspiration: We were amazed by the Popcha! movie network who represented the nodes as movies and edges as similarity relationship. But our idea was rather to take the actors as nodes, the list of common movies as edges, and also make it a **force-directed layout** to improve the interactivity.

Design Evolution: The main changes were done on the content that we wanted the user to have access to, in the **tooltips** dedicated to actors and list of common movies. Moreover, we added extra features such as to be able to search for an actor, instructions to make our plot understandable easily, and also setting up mouse over elements and right-click binded tooltips.



Mouse over edge: tooltip for the list of common movies

Main Challenge: The main difficulty came from the idea that we wanted our graph to be very smooth and also interactive. Nonetheless, as mentioned in the course the **computation** for force-directed layouts is **huge** and therefore the practical svg approach which provide already implemented interactivity was too slow for our huge networks. Thus, we decided to go on a **canvas-based approach** because it's **way faster** than svg to render big networks but the drawback is that we had to implement the support for event handlers using a hidden canvas because canvas are just drawings. This hidden canvas is basically not rendered on-screen but is a duplicate with distinct colors for each element of our network so that we know the object under the mouse of the user.

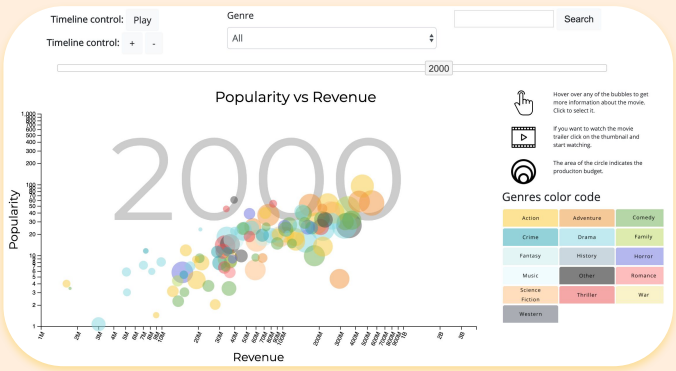


Mouse over node: tooltip for the selected actor

Other Challenges: Other challenges were to find ways to work with canvas to make basic moves such as translating the whole network and zooming in and out. To solve this issue we combined a lot of solutions from other d3.js force-directed layouts + canvas that had fewer functionalities than we have and mostly implemented from scratch.

Finishing up

After all the graphs were done, there remained the main task - **put everything together** in a coherent and insightful way. The website had to be simple, keeping the focus on the graphs rather than the design of the website. We used the same color palette as for our visualizations, and we decided to use simplistic yet elegant fonts Raleway and Open Sans. The content of the website is completely oriented to the visualisations we created - explaining and inviting users to try out everything we implemented. To make our project a true **visual data story**, we used data scientist approach and in our data story we share some of the things that we discovered by playing with the visualizations. However, we didn't want to make the user wonder around where to look and what to click, that is why on the website for each visualization we make it clear where to start, what to look and when. We hope that the data story tackles the interest of the audience and that it motivates them to increase their engagement.

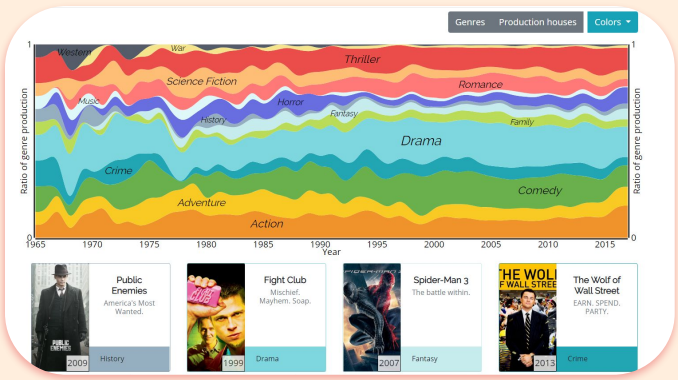


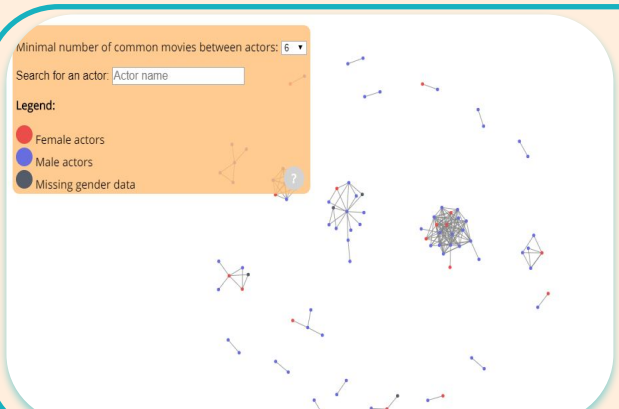
Back to the Future - main functionalities

- Select a movie to find out more (click on dot)
- Search by movie name (Search bar)
- Search by genre (click on the color code)
- Selecting the year and playing the timeline
- Click on a dot to keep the movie information up and available.
- View the trailer on youtube by clicking on the youtube icon
- More info on TMDB by clicking on the poster
- More info on actor by clicking on the picture

Once Upon a Time... in Holywood - main functionalities

- Switching between primary category: genre or production house
- Clicking on any primary category generates a streamgraph for it with detailed view based on the secondary category
- Hovering over the category shows the number of movies for the selected year, and below shows most popular movies for that category
- Switching between default and colorblind palette





The Social Network - main functionalities

- Hovering over the nodes to get the name of the actor and its connected actors
- Hovering over the edges to get the name of the connected actors
- Right-clicking on the nodes/edges to get a tooltip with additional information related to this element
- Select the minimal number of common movies
- Search for an actor on the network

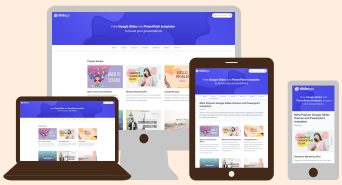
Reflexions and contributions

What have we learned?

This project allowed us to practice presenting the data in an interactive and effective manner with a customer-oriented approach. During our other courses at EPFL, we mainly focused on the backend part of the product. This course was the opportunity to switch the perspective and focus not only on the technical aspects, but also on the artistic aspect. For some of us, this was also the opportunity to learn Javascript, HTML, CSS, D3, jQuery, and Bootstrap.

How this work can be improved?

The D3 charts could be reworked to be mobile compatible. Large screen devices work properly but mobile devices required the entire layout of the visualization to be redesigned. Mobile layout is more than a question of resizing component. There are parts of the dataset which we didn't have time to explore, like the keywords of the movies or people involved in the creation of movies other than actors like directors and screenwriters.



Github repo

Website link : https://movievizz.xyz/

Thank you for taking the time to explore the movie history with us. Your vissybussy team!

Jeremy

Worked on:

- Website structure
- "Back to the future" viz
- Process book
- Website

Team experience:
A fantastic team that was incredible to work with.

Maja

Worked on:

- Milestone 1 & 2 text
- "Once upon a time… in Hollywood" viz
- Process book
- Website

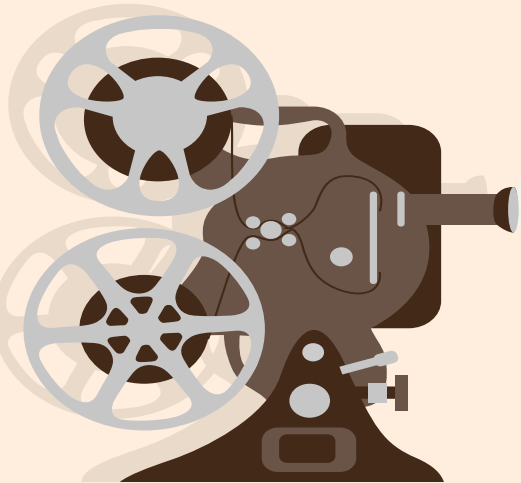
Team experience: Great team! I appreciate Johan's attention to details and Jeremy's proactiveness.

Johan

Worked on:

- Data analysis
- "The social network" viz
- Process book
- Website

Team experience: Amazing teamwork to leverage D3 and innovate on graphs.



Credits

Icons: Flaticon by Freepick, Kiranshastry, Pixel perfect

PDF template: Slidesgo

Dataset: kaggle TMDB 5000 movie dataset