COM-480: Data Visualization - Process Book

Laurent Brock, Ben Kriesel, Cindy Tang

4 June 2023

1 The path to obtain the final result

The aim of our visualisation was to apply a readable formatting to a real, consequential field of science. As we all know, the pharmaceutical industry is one of hidden truths, large profit margins, and blocking researchers from combining molecules and making key discoveries. As such, we set out to shine a bit more clarity on the field, through visualising information which can be hidden: that of the EMA's (European Medicines Agency) public database of approved medicines. The publicly accessible version of the data which we have collected is simply a XLSX file, which by itself is quite difficult to understand. In order to retrieve the information on a single medicine, it is a good format, but as soon one tries to understand the data at a larger scale, it rapidly becomes a bad format.

As such, we decided to visualise the trends and fields which are most present in the pharmaceutical industry in our visualisation. To achieve this, we sought inspiration from the Graphs lecture, specifically focusing on tree visualization techniques as we want to explore the hierarchical structure of the therapeutic areas. This guided us in developing a zoomable treemap plot as the foundation for our primary visualization. Additionally, we take benefit of the introduction lectures about HTML, CSS, JavaScript and D3.js to gain technical proficiency in coding, and of lectures about data, interactions, perception of colors to gain the best practices for visualization.

Conceptually, our main treemap is linked to all the other visualization in our web page, so that interacting with the main visualisation would then result in the other ones being affected. Therefore, in order to make this plot particularly interactive, we made it so that clicking on the plot zooms it in, therefore sub-setting the dataset. For example, if one is interested in finding which medicines the EMA has approved in the field of Oncology, one can click on it, and the other plots will adjust.

This left us with the task of finding what other visualisation we were going to use to represent the data. We settled in the idea of a histogram, which would represent how fast each medicine's discovery, and addition, happened in the dataset. Thus, one can easily see, just below the icicle plot, adjust in real time, just how many medicines were released in a certain time-span in the field. This allow us to easily realise what fields are under most active development, at which moments in history, or, more cynically speaking, which fields bring in the most money.

One of the essential parts of our visualisation was that it could be updated. Explained below in the challenges section are the details, but we wanted that our project could be updated at any time in order to incorporate the newest data possible, and that it was not fixed in time, like most likely a lot of other projects are.

Ben suggested that we compute a sort of similarity metric between medicines, in order to see just how much the pharmaceutical industry is actually innovating, or on the other hand, recycling compounds. This part of our visualisation mostly has to rely on Python-based pre-processing, as the computational burden for generating this on-the-fly is too great for a JS web page (or at least with the know-how that we have now). This data is then read into the page when necessary. This

design decision meant that we had to expand the Python / pre-processing portion of our project to now include this part. However, this does not stop us from simply running it all together when need be, in one single file.

2 Challenges & Design decisions

- A main constraint of our work was actually the dataset. Although it contains high quality information in each category, there are only a limited amount of categories which can be visualised in a meaningful and impacting manner. This is because the majority of the data is fully text-based, meaning that any sort of visualisation which relies on numbers of out of the question. One can extract numbers from the data, which is what Laurent did for the dates, which were converted to years, however this was a pretty limited approach as it only applied to a handful of columns. Therefore, we decided to restrict what we would be focusing on.

- The treemap itself can show every single category if need be, but this approach would clutter the visualisation greatly. Therefore, Laurent opted to restrict the number of subcategories which would be shown for each category, in a top-k fashion. This greatly increases performance and visual clarity, at the cost of accuracy: it is often small sicknesses or medicines which are the most interesting in this kind of data, and this decision crushes them. However, the purpose of this visualisation is to see the field at large, the market and research efforts over time and where they are being concentrated. If one desires to search for individual or lesser known treatments, the initial, XLSX formatted data might be more adapted. Thus, he had to make the design decision to remove less common points in order to better capture the global trends.

- It was quite difficult to read in the data. This is because the EMA presents the data in a XLSX format, which JavaScript cannot read by default. Therefore, Laurent had to write code which this data, and saved it locally in the CSV format instead. This format can be read by d3.js, and since this was the key framework under which we were going to base our visualisation, why not also use it to process the initial data? This was then read into JS, and special care has to be taken in order to retrieve the data in the format that one wishes. This is because the d3.csv function reads in data line-by-line, and so data had to be appended into a dataset Object every time. Furthermore, the dataset's column names are only located on the 7th line of the XLSX, meaning that extra work had to be done in order to accommodate, while also taking care of promises so that the data can be loaded asynchronously. However, the resulting formatting means that the web page doesn't freeze, and one can easily access each column of the dataset (eg. dataset["Therapeutic area"]), a syntax which closely resembles DataFrames, seen in R or Python. The main use of this effort and extra work is that any version of the XLSX from the EMA can be passed into the function and loaded into our visualisation. As such, by simply running the one-line Python fetcher, the whole visualisation gets updated with fresh data, meaning that our visualisation is not ephemeral, and can be used many years into the future with up-to-date data. To summarise, the data loading was a difficult but essential part of building our visualisation, in order to make work easier and our code useful into the future.

- At the beginning, we had some confusion between the terms and plots of icicle and treemap. Our initial intention was to create a visualization similar to the treemap depicted in Figure 1a. although we used the word "icicle" to define it in the second milestone's report. When we searched in the D3.js library, we found that the proposed treemap lacked interactivity, which was an important aspect we wanted to include in our visualization. As a result, we decided

to explore the icicle plot, as it also showcases the hierarchical structure of the therapeutic areas we aimed to represent. However, when we presented our results to people around us, we realized that the interpretation of the icicle plot was not immediately intuitive. It would have been better to display only one layer at a time, as originally planned. At that point, we discovered the zoomable treemap that offered the perfect combination of the zoomable icicle we had already implemented and the treemap we initially desired. Although this decision was made at the last minute, we found that the zoomable treemap was more comprehensive and relevant to effectively communicate the information we wanted to convey.

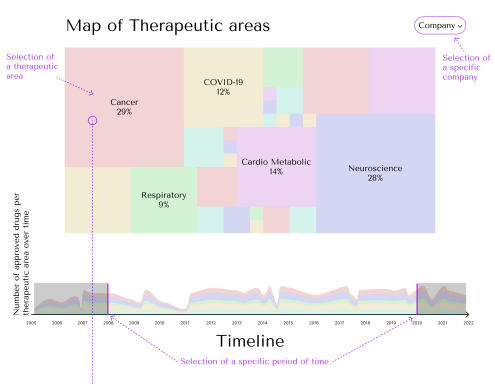
3 Changes from the previous milestone

From the previous milestone, we made some changes to our initial sketches and plans in Figure 1.

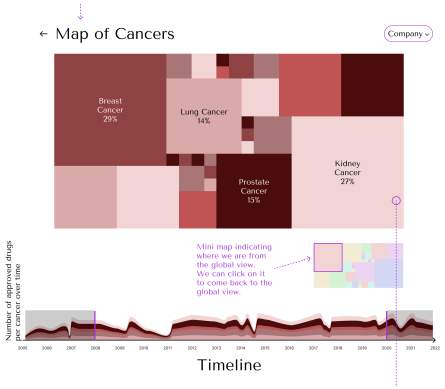
- Firstly, we intended to use different colors for each category in our treemap. However, since there are numerous categories, it became challenging to find a suitable categorical color palette with distinct colors. As a result, we opted to use a single color for all categories in the treemap. To establish a visual connection between the treemap and the histogram, we decided to change the color of the treemap box to match the color of the corresponding bar in the histogram when the user hovers over it. This modification enhances the relevance and cohesiveness of the visualization.
- Originally, we had planned to position the histogram of time below the treemap. However, upon further consideration, we realized that placing them side by side would be more efficient. This arrangement allows users to view both visuals simultaneously, facilitating the connection between both plots.
- Another modification we made was regarding the visualization of drugs when zooming in. Initially, we planned to use circles instead of rectangles to differentiate the drugs themselves from the categories. However, we found that using circles caused confusion, so we decided to stick with rectangles for both drugs and categories.
- Due to time constraints, we had to give up certain options. For instance, we abandoned the idea of incorporating the choice of the pharmaceutical company, the ability to click on a particular year in the timeline to update the treemap, and the ability to hover over drugs to retrieve detailed information. It was unfortunate to let go of these features, but we prioritized focusing on the main aspects of the project. However, these options hold great potential for future improvements and could be considered in subsequent iterations.

4 Peer assessment

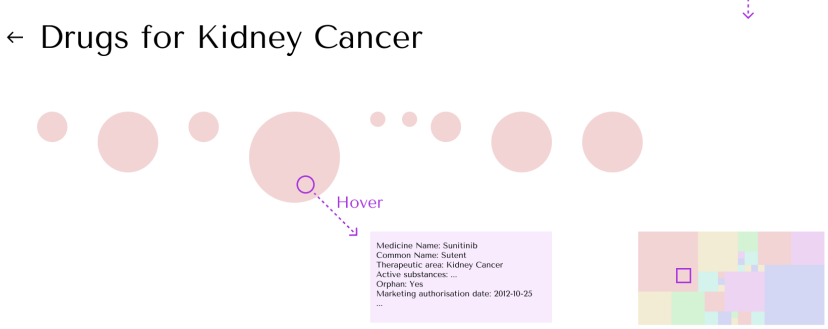
- Laurent:
  - Fetching of the data and converting it to CSV in Python, loading the data into Javascript, and formatting it into a readable and usable format for the rest of the processing (equivalent format to that of a dataframe, an object of arrays) asynchronously.
  - Adding many methods in order to facilitate accessing the data, and processing it further down the line for the other members of my group.



(a) Initial visualisation (with timeline)



(b) Example 1: Zoom in on cancer details on click



(c) Example 2: Zoom into list of drugs for kidney cancer

Figure 1: Plans for Milestone 2

  - Formatting the data into a hierarchical format which can then be easily called in the icicle plot creation function.
  - Modification of the icicle plot so that it would suit our needs (from a given template) making it wider, having the amount of columns we needed, and adding the date data into the dataset creation a posteriori.
  - Skeleton of the histogram function, which would be below, both in HTML and JS, and made sure that it gets the right data, at the right time (on click).
  - Writing a large portion of this report.

- Ben:
  - Finding our amazing dataset.
  - Helping to replace the zoomable icicle plot by the zoomable treemap.
  - Presentation of our website in the screen-cast.
  - Working on the extra deliverable (which we decided not to put in the website because the medicines are all not similar) such as:

    * Writing the Python code to get molecular structures of the drugs, scraping PubChem
    * Calculating molecular structure similarities with the found molecular structures
    * Creating MDS plots to display similarity in molecules to the visitor

- Cindy:
  - Exploratory Data Analysis to gain insights from the dataset.
  - Drawing the initial plans & sketches to put our ideas into place.
  - Design and implementation of our whole website skeleton using HTML and CSS.
  - Integration of the formatted data provided by Laurent and Ben into the visualizations using D3.js library.
    * Zoomable icicle plot (eventually replaced by the treemap).
    * Histogram of marking authorization years.
    * Zoomable treemap.
  - Finalization of the report.

- All Team:
  - Discussion and choice of the most suitable visualizations for our purpose.