CS416: Data Visualization, Final Project Report (Summer 2021)

Marina Polupanova (marinap2@illinois.edu)

University of Illinois, Urbana-Champaign

Project URL: https://violetta-ta.github.io/index.html

Git hub: https://github.com/violetta-ta/violetta-ta.github.io.git

---

This project was done using the data from Kaggle dataset provided by provided by Seungwon Song for Udemy courses:

https://www.kaggle.com/songseungwon/2020-udemy-courses-dataset.

This data was pre-processed for making visualization in the following way:

1. Only 3 raw files were chosen for analysis, for Tech, Business and Finance domains.
2. Those 3 files were processed by Python script, which calculated the number of courses in each domain for a certain “stars” value. (please see the script printout at the end of this report)
3. As I have processed files by the script separately, I have combined the result further in Excel to have the single file with columns “Stars”, “Tech”, “Business”, “Finance”, where “Stars” depicted the number of stars for this row, and the other 3 columns showed the courses count for the respective domain. Data file was stored as “stars_overall.csv”.

I have used the Firefox browser for the visualization check, so the visualization works more reliably in this browser. The work in other browsers was not checked.

Here is the more detailed description on the completed work by sections:

- **Messaging.** What is the message you are trying to communicate with the narrative visualization?

I want to let users explore how “stars” scores are distributed for 3 courses domains, which percentage of the overall courses in this domain have certain “stars” value, and based on this make judgement, which domain has the best rating. So, my message is “Which courses are rated best?”.

- **Narrative Structure.** Which structure was your narrative visualization designed to follow (martini glass, interactive slide show or drop-down story)? How does your narrative visualization follow that structure? (All of these structures can include the opportunity to "drill-down" and explore. The difference is where that opportunity happens in the structure.)

As the major target of my visualization was to explore different options, I have chosen the format of drill-down story.

The user first is landed on the page with the bar chart with the 3 bars, which show the number of courses overall for each of the 3 topics which we want to discover. From that page, he has 3 choices to go: to explore Tech, Finance, or Business courses scores, clicking on one of the respective bars.

On the other 3 pages, where he go from the 1-st page by his click, he has the chance to see the bar chart depicting distribution of the courses count over the “stars” scale for his chosen area, see the customized annotations, and see the exact values of courses count for each “stars” value bar. On top of that, user can click on any bar and see the comparison: which percentage of the courses in each area have a certain “stars” score.

From any of the domain pages he can go back on the main page, and start exploring further, clicking on the other bars and looking at the data for the other domains.

- **Visual Structure.** What visual structure is used for each scene? How does it ensure the viewer can understand the data and navigate the scene? How does it highlight to urge the viewer to focus on the important parts of the data in each scene? How does it help the viewer transition to other scenes, to understand how the data connects to the data in other scenes?

I have used a similar structure for each of the 4 scenes which I have in the visualization. It includes the header, the description text and suggestions for the user with describing what he can do, the graph itself, and the footer with the link to the row data and some more instructions to the user. Also, on the domain scenes, I have the button which let the user go back to the main page. The text instructions in the description and hint on tooltips of the main page can help the user to understand what he can do and what is the connection to next page.

On each scene, I have placed the annotations near the respective interesting bars, which help to focus on important items in the graph. In order to help users better understand which domain he studies, I have kept the same color code over all the pages: each domain has its own code which is used both on the main page and on domain pages. Also, for consistency, the vertical axe scale is same for domain graphs.

- **Scenes.** What are the scenes of your narrative visualization? How are the scenes ordered, and why

I have 4 scenes in the visualization: main page and 3 domain pages. First, user can go to the main page, and then, he can choose which of the domain pages he would like to explore and in which order. The reason for this order is that the topic and message of the

visualization are more to explore the proposed data than to explain to the user some pre-defined answer.

- **Annotations.** What template was followed for the annotations, and why that template? How are the annotations used to support the messaging? Do the annotations change within a single scene, and if so, how and why

I have followed style template for annotations which is close to the tooltips template. It includes same font size, color, same background color and annotations background rounding. On top of that, I have used same width for all annotations and long tooltips, letting only height to expand. The reason for that, is that in this case annotations and tooltips look in a unified way, and also do not hide the parts of the graph which user may want to observe (as other width would hide some objects or other annotations).

Annotations support the messaging in a way, that they draw the attention of the user to interesting parts of the graph, letting him notice how the scores/count change for different domains.

Annotations change with each scene, as each domain appeared to have its own interesting details, which I wanted to highlight. As Business and Tech has similar curve, I used just one annotation per each, noticing only one detail. For Finance, as the graph is quite different, I have left couple of annotations to draw the attention of the user to the difference.

- **Parameters.** What are the parameters of the narrative visualization? What are the states of the narrative visualization? How are the parameters used to define the state and each scene?

I have used following parameters for my visualization:

1. Topics (domains), which have “labels” name in the code. The user actions on the first page depend, on which “labels” parameter value he will chose, clicking on one of the bars. The states of this parameter: “Tech”, “Finance”, “Business”, so corresponding state of the narrative visualization will be counts distribution for the respective domain. When user click on some of the bars, on.click event is being captured, and this event handler function uses “labels” parameter to construct the URL, which will be used to go to the next page with respective counts distribution.
2. On the domain pages, I use the parameter “id”, which is a rectangles attribute, and has the states from 0 to 49 (parameter is calculated in “onClick” function). The state of the visualization which it will allow to see - a clarification tooltip with percentages of the courses on the domain page. In order to trigger calculation of this parameter, user need to click on some of the bars on the domain page, the “onClick” function will calculate respective clarification figures, and the function will display the results as a tooltip.

- **Triggers.** What are the triggers that connect user actions to changes of state in the narrative visualization? What affordances are provided to the user to communicate to them what options are available to them in the narrative visualization?

I use 4 event handling functions, and 3 triggers to connect user actions with the state of visualization:

1. In the main page, I use the .on (“click”) handler in the .bar class rectangles, which constructs the URL to go to the domain page and so, when a user clicks on any of the bars in the main page, the user can see the next scene. Affordances provided to the user: apart from the note in the description of the scene, the user can see on mouseover that the bar becomes larger and more dark, which can mean that it is clickable.
2. “onclick” event. In the domain pages, same trigger as in the first point (click of the user on bars) generate specialized tooltip with extra information for the user. Affordances provided to the user: apart from the note in the description of the scene, the user can see on mouseover that the bar becomes larger and more dark, which can mean that it is clickable.
3. on “mouseover” trigger. On mouseover event, the user is able to see the tooltip for the certain bar. On mouseover the bar becomes darker and larger, so the user can see to which exactly bar he see the tooltip
4. on “mouseout” trigger, the tooltip becomes invisible, and the user can see the clean graph, or explore other bars. On mouseout, the bar becomes lighter and smaller, returning to initial state, so the user can see that no bars are highlighted.

Supplementary materials:

Here is the code of the data summarizing script, which I have used:

```
import pandas as pd
import numpy as np


udemy_tech = pd.read_csv('Udemy_data/udemy_tech.csv')
udemy_finance = pd.read_csv('Udemy_data/udemy_finance.csv')
udemy_business = pd.read_csv('Udemy_data/udemy_business.csv')


udemy_finance_red = udemy_finance.groupby(['Stars']).size().reset_index(name='counts')
udemy_business_red = udemy_business.groupby(['Stars']).size().reset_index(name='counts')
udemy_tech_red = udemy_tech.groupby(['Stars']).size().reset_index(name='counts')


udemy_tech_red.to_csv('tech_reduced.csv', index=False)
udemy_business_red.to_csv('business_reduced.csv', index=False)
udemy_finance_red.to_csv('finance_reduced.csv', index=False)
```