World's Monuments Visualization

Samuele Olivieri Pennesi 1753295
Francesco Ottaviani 1759720
September 2021

***Abstract*— In the last years, the importance to preserve and to protect the most important sites in the world became a fundamental task. Some of them are designated by UNESCO for having cultural, historical, scientific or other form of significance. Our project, named World's Monuments, aims to represent through several visualizations, the most famous places all around the world, divided by country, category and relevance. Our work is available at https://github.com/Programmer100th/Visual-Analytics-Project.**

I. Introduction

In this report we will analyze different types of visualizations for the most important sites in the world, trying to allow people to access various information generated by cross-correlation of data and different interaction that a user can do. The focus is only on several types of categories, like monuments, museums, churches and so on.
With the **World's Monuments** project we want to invite people to take an interest in the culture and to inform themselves about the important sites that are in the world, so through the support of our platform they may be able to navigate through the menus to learn more about the characteristics of each individual country.
We aim to let travellers discover new interesting places and also to make people aware about the importance and the beauty of the cultural sites all around the world. In other words, we want to provide the ultimate travel guide, highlighting interesting aspects thanks to our visualizations.

II. Related Work

During the development of the project we asked ourselves what could be the greatest interests of our users, starting from the data showed and arriving to the different views to propose. Since our project embodies an idea somehow new in the Visual Analytics dimension, we had some difficulty finding something very close to our idea. However, we started looking for something that would help us improve the setting of the views, making them more helpful to people, and we also looked for new ideas to enrich our dataset. Therefore we tried to find additional information to link to the dataset we already had (i.e. GeoNames), so thanks to the paper **An Aspect of Archaeology's Recent Past and Its Relevance in the New Millennium** [1], written by J. A. Sabloff and W. Ashmore, we have thought about the significance of every single site. Here, the authors highlight the importance of a growing sector as archaeological, but the discourse can also be extended to all categories of sites. It is important to underline that collected statistics of this paper mainly concern people's interest in the most famous sites.
Regarding this topic, we thought of a possible value to be assigned to each site, in order to indicate its relevance (or popularity) integrating Pageviews to GeoNames, so as to be able to draw up a report on the most famous places. In confirmation of this, citing the paper **UNESCO World Heritage sites and tourism attractiveness: The case of Italian provinces** [2], the authors described the condition by which people are more inclined to travel to countries (in this case, Italy) to which the world heritage sites belong, according to the list drawn up by UNESCO.
Both of papers listed above try to convey the importance of cultural and historical sites, so they have the same goal as ours. As regards the dataset, looking for a project that somehow used GeoNames, we came across **YAGO** [3], which uses a different approach from our project (since it creates a knowledge graph based on geographical sites, translating them into different languages through

Wikipedia) but it uses precisely the same dataset, with the addition of Wikipedia data.

III. Dataset

The main source for our project is the **GeoNames** dataset [4]. It contains over 25 million geographical sites all around the world. Each entry of the dataset has 19 attributes, but for our purposes we will only use the following ones: **name, latitude, longitude, feature class, feature code, country code**.
Also, since our project focuses on points of interest, we are going to take only the data whose feature class is S, which in GeoNames means spot, building, farm. GeoNames divides the sites into very detailed subcategories (actually we would say too much...). The complete list of 645 feature codes can be checked in the appropriate section of the GeoNames website [5]. For this reason, within this macro-category, **we selected** only few subcategories:
CH **(Church)**, ANS **(Archaeological site)**, HSTS **(Historical site)**, PAL **(Palace)**, BLDG **(Building)**, MUS **(Museum)**, CSTL **(Castle)**, MNMT **(Monument)**, AMTH **(Amphitheater)**, PYR **(Pyramid)**.
GeoNames was only the main source for our project. In fact **the data that we visualize** with D3.js [6], **comes from an important data integration work**, mainly executed with **Talend** [7], but let's procede step by step.
We have also used the **Wikipedia Pageviews** dataset. In a few words, for each existing Wikipedia page, it associates to it the **number of visits** (clicks) that it receives in a specific hour of a specific day. Semantically speaking, we want to use this value as a sort of ***relevance*** of the sites. Since Covid-19 changed our travelling habits (and a Wikipedia page of a monument receives clicks more likely by people who are visiting a place), we took data from the 1st January of 2019. In order to have data reliable as much as possible, we **sum up the visits received by each Wikipedia page in 24 hours**, meaning that the relevance that we associate to each site corresponds to the number of visits that its Wikipedia page has received during the 24 hours of 1st January 2019. In order to reduce the amount of data and make the integration process faster, **we maintained only the pages with more than 50 visits**. Semantically speaking, if a Wikipedia page of a monument receives less than 50 visits in 24 hours, very likely that place is not so relevant for our application.
Coming to the actual integration process, without going into detail, it was useful in particular to **join the Pageviews data with GeoNames on the name of the page/site**, and to make the data more "user friendly" by **substituting the codes (of categories and countries) with their actual names**.
Note that the correspondence 1:1 between Wikipedia page and name of the site in GeoNames is impossible to obtain (here it is important to underline that GeoNames is a free downloadable dataset, and even if on the website it is suggested to use english names for places of interest when possible, this is not guaranteed). When the correspondence is not found, the relevance field of that site will be empty.
At the end of the story, we obtained a dataset in tsv format composed by **67282 sites of interest** all around the world. Each row has the following attributes: **name** (of the site), **longitude** (in decimal degrees), **latitude** (in decimal degrees), **country** (full name), **category** (full name), **relevance** (coming from pageviews) **and country iso** (ISO-3166 2-letter country code).

IV. Visualizations

The page is composed of the menu, in the upper part of the screen, and 5 visualizations: a **choropleth map** representing the entire world, an **heatmap/scatterplot** for the currently selected country, a **starplot**, a **barchart**, and a **scatterplot coming from PCA** algorithm.
The menu allows the user to filter the sites to visualize, by choosing the **country** of interest, **one or more categories**, and a **threshold relevance**. Also, to improve the ease of use, the country dropdown menu is supported by a search bar in which the user can type the name of the country, the category sub-menu provides the buttons select and deselect all, and the threshold relevance is editable using a slider which ensures the maximum granularity.
Note that **any change in the menu will affect all the 5 visualizations**.

A. Choropleth map

The **choropleth map** uses a **TopoJSON file** which contains information about the boundaries and the geographical position of the countries. More specifically we downloaded the file from Natural Earth Data [8].
Our aim is to **assign a color to each state, depending on the sum of number of sites** respecting the categories and the relevance given by the user. We used the threshold scale of D3.js, and for the choice of the colors we have used **ColorBrewer** [9] with a sequential color scheme. By the way, since according to the filters the average of sites per country may drastically change, we have decided not to maintain this scale fixed. Without going into details, the less is the number of sites *"survived"* after applying the filters, the lower is the number of colors used to paint the countries (as well as the numeric range).
Additionally, **we have put a color legend** to help the user in understanding the correspondence between the color of the country and number of sites that it contains. The borders of the selected country are thicker, so that it is easily recognizable in the map.
As regards the interactions, when the user hover a country, it becomes green and basic information such as name of the country and the number of its sites are displayed. When the user click on a country, this will become the currently selected one on the heatmap/scatterplot map, and will consequently influence all the other visualizations. This interaction also helps the user in recognizing and selecting a country even if he doesn't remember its name. Furthermore on the map are active pan and zoom.

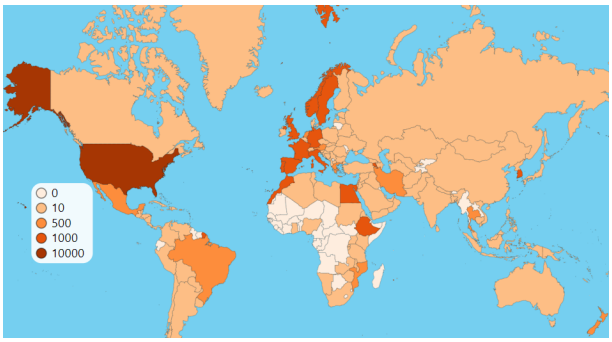

Fig. 1: *Choropleth map of the world.*

B. Heatmap/Scatter plot

Contrarily to the choropleth map, **this visualization focuses on a single country**, the one selected by the user through the menu. Since our goal for this map was to display the precise geographical position of the sites, we needed to use something way more complicated than a simple TopoJSON file. That's why **we used a map from Mapbox** [10]. More specifically, Mapbox Streets, which according to their website "is a comprehensive, general-purpose map that emphasizes accurate, legible styling of road and transit networks".
The information displayed on screen change accordingly to the zoom. Starting from the name of the countries, reaching a certain level of zoom it is possible to visualize even the name of the streets.

Fig. 2: *Heatmap of the chosen country.*

In order to give the most understandable representation possible, **if the country has more than 100 sites** (again, depending on the filters applied by the user), **an heatmap is visualized**, which gives at least an indication of where the sites are more concentrated.
Then, **if the users continues to zoom, the points which correspond to the actual coordinates of the sites will be displayed**. In this way we avoid overplotting. **If the number of sites is less or equal than 100, instead, the points are directly visualized** since the heatmap is not needed.

Fig. 3: *Scatter plot of the chosen country.*

The color of the points depends on the category of the sites. We used the scale ordinal of D3.js and for the colors, the qualitative scale of ColorBrewer. Apart from zoom, pan, and rotation, automatically made available by Mapbox, when the points corresponding to the sites are displayed, the following things happen on hover. The point becomes bigger, and both the name of the site and the corresponding relevance are displayed. As regards the other visualizations, in the star plot the category of the site becomes bigger and green. If present, the bar corresponding to that site in the bar chart becomes green, while in the scatter plot the corresponding point becomes bigger. Clearly, these effects are cancelled on mouse out.

C. Star plot

With the **star plot** (or radar chart) it is possible to compare the **percentage number of sites within a country**, divided by category.
On the axis the world's sites are represented through the logarithmic scale, in a resulting range between 0 and 5, in which the higher the value, the more sites there are in that category than the others. So this range represents a sort of *score* for each category of the selected country.
For each axis of the chart there is a label that defines the category, therefore if the label is bold means that it is one of those selected by the user. Furthermore for each label the user can hover over it or click on it. If the user hover a label, this one becomes bigger and green and on the other graphs will be visualized only information about the current category, while the sites that are not included will be temporarily darkened (such effect will finish as soon as the user will mouse out).

When the user click on a label, instead, the chosen category becomes the only one selected in the complete category list, so this behavior affects the entire project making visible only places of that category.

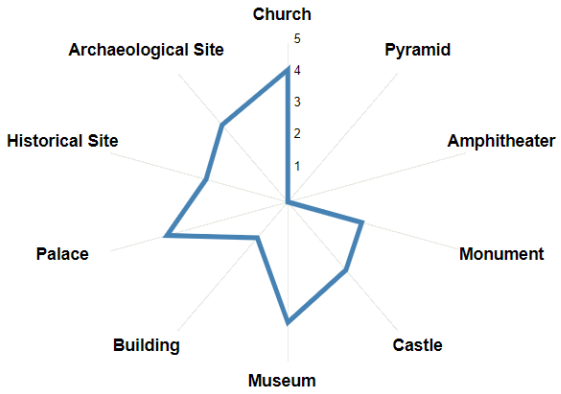

Fig. 4: *Star plot of the chosen country.*

D. Bar chart

The **bar chart** is used to visualize the **10 sites with highest relevance** respecting the filters chosen by the user. Clearly, they are ordered according to relevance.

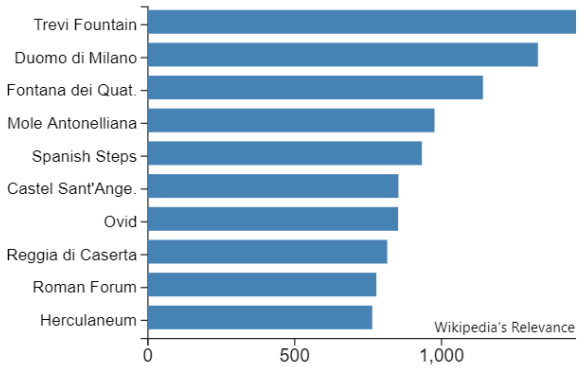

Fig. 5: *Bar chart of the selected country.*

For each bar, both hover and click are defined. When the user hover a bar, the name, the category, and the relevance of the site are displayed, and the bar becomes green. Regarding the other visualizations, the category of the sites becomes green and bigger in the star plot, and the circle of the corresponding site becomes bigger in the scatterplot. The map for the single country is centered on the coordinates of that site, in both heatmap/scatter plot mode. When the user clicks

on the bar, instead, the single country map not only is centered on those coordinates, but is also zoomed, so that the user can understand exactly the position of that site.

E. Scatter plot from PCA

Principal component analysis (PCA) [11] is the process of computing the principal components and using them to perform a change of basis on the data, sometimes using only the first few principal components and ignoring the rest.
We want to represent on a scatter plot the **similarity between country's sites in a 2-dimensional space**. Taking into account that we consider three columns from our reference dataset (**latitude, longitude, relevance**), and only the sites with relevance value not null, we want to represent through this algorithm the proximity between the places of the chosen country, not only as geographical proximity but also as popularity, so we expect that geographically distant sites in the country, but with the same value of popularity, may somehow come near within the graph. Interactions on this graph are identical to those defined in IV-B.

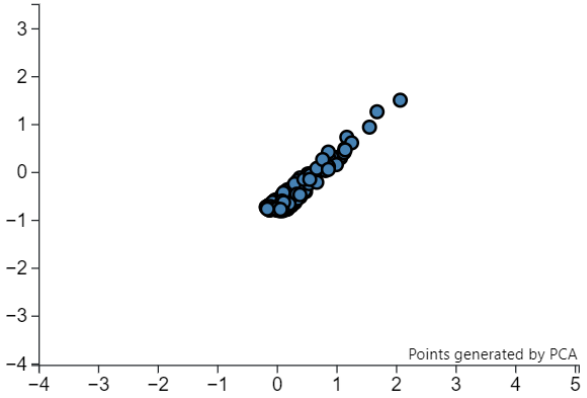

Fig. 6: *Scatter plot from PCA*.

V. Analytics

We make use of several analytics.
One of these is used to **assign a color to the countries** in the world map.
More specifically, it works in the following way. The user select the desired categories and the threshold relevance through the menu. The sites contained in the TSV file (our dataset) are filtered according to the given user input. At this point, we call the rollup function from D3.js, which basically **performs a group** according to a specific attribute (in this case the iso code of the country) **and then a sum**. In this way we obtain an association of country - sites, which will be exactly the one used by the threshold scale to assign the right color.
The other analytic regards the **score of the categories in the star plot**. When the user selects various options of country, categories and relevance in the menu, it is called the function for the filtering of the data, based on the chosen parameters. After that, we call the rollup function from D3.js to link each category to the number of sites of that category in the chosen country. In this way, obtaining a percentage value by dividing the number of sites of each category by the total number of sites in the chosen country, we can compute in a **logarithmic scale** the final score of all the categories.

VI. Insights

In this section we present two possible scenarios in which someone uses our service for a specific purpose, obtaining unique information by combining the various interactions available.

A. Scenario 1

The first scenario concerns an **archeology student**, who wants to find the country with the largest number of archaeological sites in it, in order to visit that country and then **take a tour of the cities with the highest density for this type of places**. So choosing the desired category (i.e. *Archeological sites*), the choropleth map will be modified according to the action made by the user, then with the support of the color scale the user can identify the country (or more, if they have the same number of occurrences) that has the highest number of sites. Looking at the bar chart he can get information about the most important sites in that country, so the hover action on the bars could help him to understand where the sites are located, since doing this action will make the points on the scatter plot of the single country map more evident. In this way he knows which are the best cities to visit, and finally he can organize the tour.

B. Scenario 2

The second scenario exploits the entity of ***Ministero dei Beni Culturali***. The goal is to **remodulate the price of museum tickets**, making the

most requested ones a little bit more expensive, and the least requested a bit cheaper. Using our application, chosen the category *Museum*, and looking at the bar chart, it is very easy to find the 10 most requested museums. Also, even if a certain museum is not in the top 10, but has an high relevance, it is possible to look at the scatter plot map of the single country, to check whether in that region there are other sites with a comparable relevance. If not, this means that the concerned museum is more valuable.

VII. Conclusions and future work

Our project make possible to analyze the data in a way that without visualizations and only considering the classical database structure, wouldn't have been possible. But something that we consider to improve in the future is the dataset itself. In fact, there are mainly 2 problems.

The first one depends on the association between the Wikipedia page and the name of the site in GeoNames. Unfortunately, many sites have not an english name, and this brings to not find a correspondence when, for example, a Wikipedia page for that site exists only in the english language.

Other times, instead, the differences between the two names are very little, let's say an accent or an article. An example of this is the UNESCO World Heritage Site Taj Mahal, which in the GeoNames dataset has an accent, not present in its wikipedia page. One way to solve at least partially this problem could be to use the approximate string matching, a technique of finding strings that match a pattern approximately.

Still other times, the GeoNames dataset has a very particular name for a certain site. Just to make an example, the name given to the well know Sagrada Familia, located in Barcelona, is Temple Expiatori de la Sagrada Família.

A way to solve these problems (for what concerns the most famous and important sites) could be to actively change the names in Geonames, which could also be helpful for the community.

The second problem regards exclusively the GeoNames dataset. As we have already mentioned, it is without doubt the most complete source we could have found. But this has a drawback: the presence of too many sites, even of those that are not actually interesting.

Note that, even with this problems unsolved, our project shows results that we would define absolutely good and valid for the purposes we described in the previous paragraphs.

References

[1] An Aspect of Archaeology's Recent Past and Its Relevance in the New Millennium - *https://link.springer.com/chapter/10.1007/978-0-387-72611-3_2*

[2] UNESCO World Heritage sites and tourism attractiveness: The case of Italian provinces - *https://www.sciencedirect.com/science/article/abs/pii/S0264837718318155*

[3] YAGO: A Multilingual Knowledge Base from Wikipedia, Wordnet, and GeoNames - *https://link.springer.com/chapter/10.1007/978-3-319-46547-0_19*

[4] GeoNames dataset - *https://download.geonames.org/export/dump/*

[5] GeoNames feature codes - *http://www.geonames.org/export/codes.html*

[6] D3.js - *https://d3js.org/*

[7] Talend - *https://www.talend.com/it/products/talend-open-studio/*

[8] Natural Earth Data - *https://www.naturalearthdata.com/downloads/50m-cultural-vectors/*

[9] ColorBrewer - *https://colorbrewer2.org/*

[10] Mapbox - *https://www.mapbox.com/*

[11] Principal component analysis: a review and recent developments - *https://royalsocietypublishing.org/doi/10.1098/rsta.2015.0202*