Visual Analytics on Terrorism attacks

Marco Lupia
Francesca Fraioli

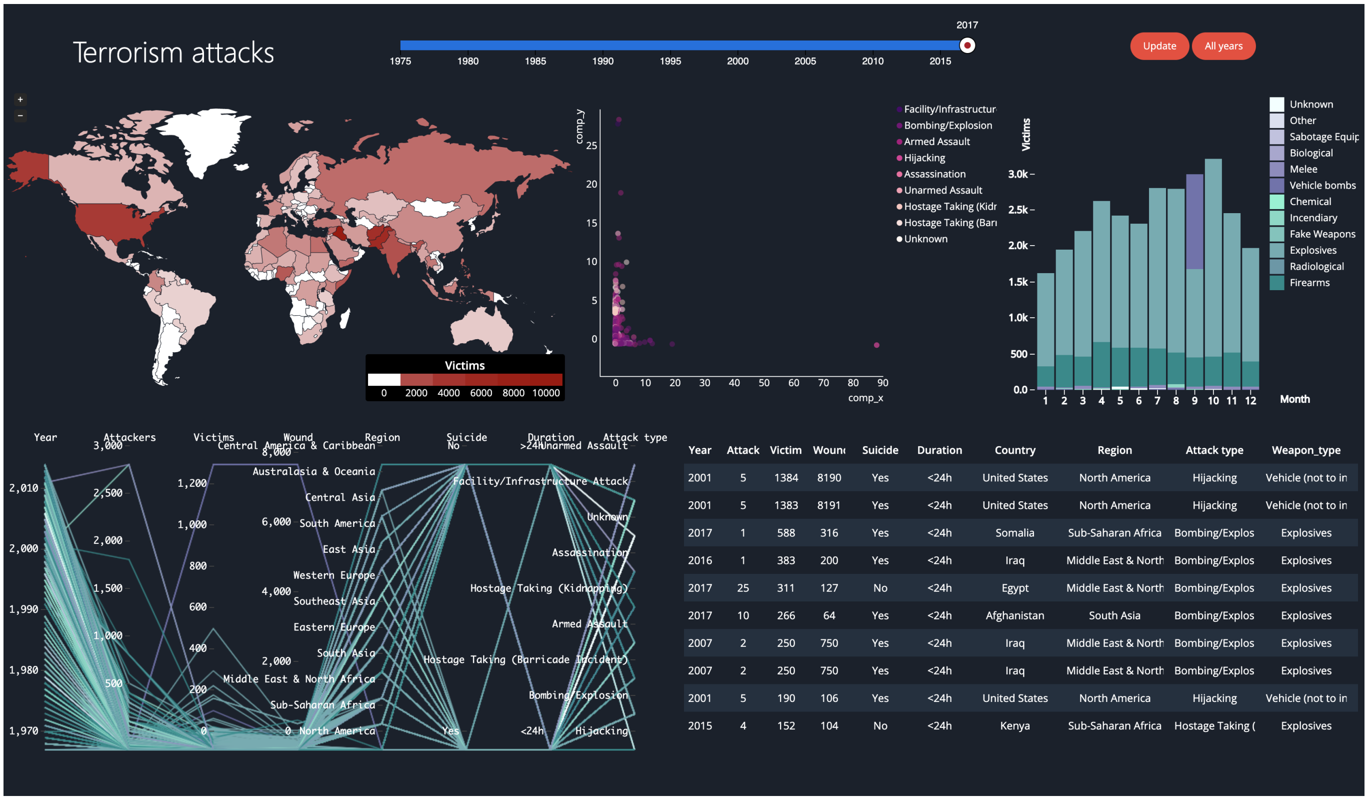

Fig. 1. Entire website interface. Every view is updated when the user makes a selection.

Abstract— This project is about visualizing terrorism attacks happened in in the last 40 years in order to make people more aware and well-informed. The user can see at a glance what are the worst, most deadliest attacks, where they took place and what weapons were used to commit them. The system uses *cosine similarity*, a measure of similarity in order to calculate the five most similar attacks compared to a chosen one.

1 Introduction

We decided to focus our attention on a dataset related to Global terrorism attacks.

Terrorism is usually understood as the use or threat of violence to further a political cause. Acts of terrorism across the globe have increased noticeably in recent decades, especially in particular countries or regions of instability.

The use of terrorism to further a political cause has accelerated in recent years. Modern terrorism largely came into being after the Second World War with the rise of nationalist movements which recognised the ability of terrorism to both generate publicity for the cause and influence global policy.

The attacks of 11 September 2001 marked a turning point in world history and the beginning of the 'War on Terror'. The attack killed more than 3000 people making it the deadliest terrorist incident in human history. The subsequent War on Terror led to the invasion of Afghanistan in 2001 and Iraq in 2003. Terrorism pre-9/11 was concentrated in Latin America and Asia, but shifted to the Middle East post-9/11. More than a quarter of all terrorist attacks between 9/11 and 2008 took place in Iraq. This is apparent in the map we implemented.

Our dataset represents data about the attacks like weapons used, number of victims and wounded and other interesting details. For each attack, the user can view a summary with details and insights about it. He can easily see what are the most disastrous attacks, where they took place and in what year.
We also thought that it is interesting to give the user the possibility to chose an attack from a table chart and show him the five most similar attacks of the same type, among all the others attacks happened in the world. We have made this computation through the *Cosine similarity*.

2 Dataset

The dataset used in this project is taken from Kaggle [5]. It contains about 10.000 tuples, each with 20 attributes. Tuples with `null` values were removed as well as attributes that aren't interest for our analysis. In this way we obtained a consistent and more manageable dataset with an AS index of about 200.000 .
Among the used attributes there are some of them that are categorical: country, region, attack type, weapon type. There are also some numerical attributes: year, month, number of victims, number of wounded,

number of terrorists. These represent information about the attack type and they are used for statistical calculations.

3 Visualization

The visualization that we have developed is composed by 5 views, each showing different aspects of the dataset.
All of them are coordinated with each other, so clicking on an element of a view will update and highlight all the information in the others.
At the beginning the views show the data available for all the years and all countries in the dataset. The user has the ability to choose a specific year from 1975 to 2017 using the slider, and all the views are updated accordingly. He can also choose one or more countries in the map to see changes related to that selection.

In details, the map allows the user to chose a country for which he wants to know more; the scatterplot represents, in each point, a terrorism attack based on the selected filters; the barchart represent the number of the victims for each month of a chosen year (or all years) grouped by the type of the weapon used; the parallel lines represents the most deadliest attacks based on number of victims, with a table that shows the precise value for each field of the attack we decided to represent.

Each charts is color coded with *colorblind safe*[1] patterns, together with its relative legend that explains the color encoding.

A slider allows to select a specific year if it is available in the dataset. Once a year is selected, the user can update all views by clicking on the red *Update* button. The user can also reset all views and select all available years with the red *All years* button.

3.1 Map

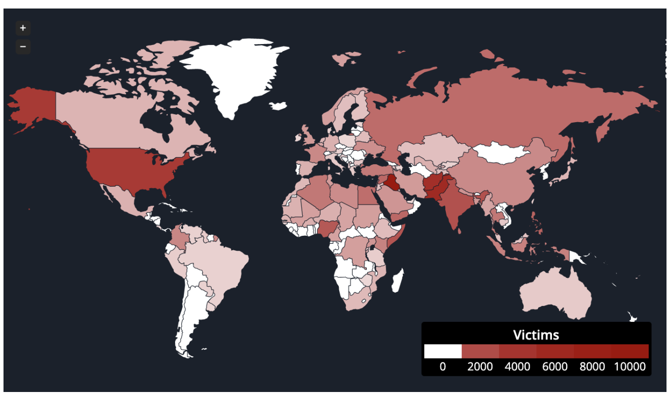

Fig. 2. Map in which the countries are colored by the number of the victims of all the attacks that are happened in that country

The zoomable map shows which countries have been hit by the most disastrous attacks. The color encoding with a red scale represents the number of deaths, as shown the legend below the map.
When hovering with the mouse on a country, a tooltip appears with the country name and the relative number of victims.
By clicking one or more countries and/or choosing a year from the slider (can be one specific between 1975 to 2017 or all of them), the user can see the related results in the other charts: Scatterplot highlights the relative point; barchart highlighted bars, which one shows the sum of the victims based on the weapon available for that countries and year; Parallel Coordinates shows the relative lines and Table reports the 10 worst attacks, based on the number of victims, among all the attacks happened in the chosen countries and/or year.
If the user chooses a country with no data in the dataset, a popup will be shown.

3.2 Scatterplot

Each point of the zoomable scatterplot represents a terrorism attack and the color encodes the attack type (Assassination, Armed Assault,

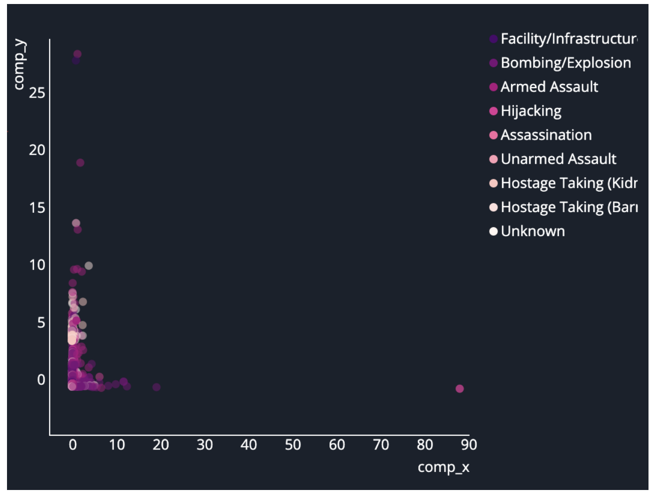

Fig. 3. Scatterplot that represents the whole dataset. Each point is a terrorism attack and the color of the point encodes the classification of each type of attack

Bombing/Explosion, Hostage Taking (Barricade Incident), Hostage Taking (Kidnapping), Facility/Infrastructure Attack, Unarmed Assault, Unknown). In order to do a dimensionality reduction, due to the high dimension of the dataset, we decided to apply *Principal Component Analysis*. It doesn't required a deep knowledge of the problem and it converts a set of correlated variables into a set of `principal components` with maximum variance. We plotted the first two components returned by PCA calculation on the scatterplot. This work is done by the python function `pca.action()` using `pandas` [6] to read and parse the csv original dataset and `sklearn`[7] to compute PCA values. At the end the result is saved in `pca.csv`.

Thanks to 2D scatterplot it's possible to spot possible clusters, in our case the user can easily see where the different type of attacks are grouped together in a particular area (in figure 3) for example is easy to see the Boombing/Explosion attacks in the bottom part of the plot). Points can be zoomed to better distinguish them.
Each points has two handlers: when hovering on it, a tip appears with information about the city, country and the type of attack. When clicking on a point, a popup appears with the summary of the attack.
There is a legend for the scatterplot which explaining the color encoding the type of the attack of each point.

3.3 Stacked Barchart

We analyzed, for each month of all years or of a specific year (depending on the choice made by the user), the number of the victims made by each type of weapons used (Firearms, Radiological, Explosives, Fake Weapons, Incendiary, Chemical, Vehicle bombs, Melee, Other, Unknown).
The chart shows one column for each month: each of them is composed by different weapon types (10 or less) and represent the sum of fatalities based on the type of weapon.
These data are calculated by a python script and read from `barchart_data.csv`.
We added some interactions with this chart: by hovering on a single sub-columns of a bar it becomes highlighted. A tip appears showing the type of the weapon used, the number of victims caused and if the user previously has selected a specific Year or/and a Country this will be also shown. We added this tip because the height of some bars can be very small compared to others, due to the significant difference of number of deaths, so sometimes the number inside could not be easily visible. In order to improve this possible inconvenience we added an interaction on the legend: the user can select a specific weapon type on the legend and see only the related column.

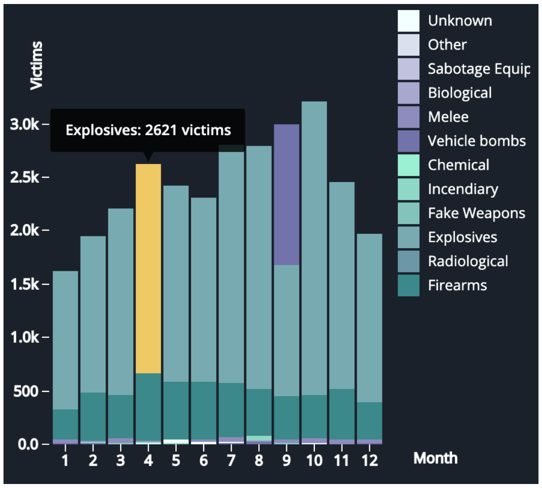

Fig. 4. Stacked Barcharts, represents for each month of the year the number of victims grouped by weapon type

When a bar is clicked all other charts are updated, all the changes are relative to the chosen bar (so to the weapon type).
When the user clicks on a line of the Table or chooses a range on the Parallel's axis, the bars are updated and highlighted for the relative chosen data.
There is a legend for the Barchart which explains the color of the type of the weapon used for the attack.

3.4 Parallel Coordinates

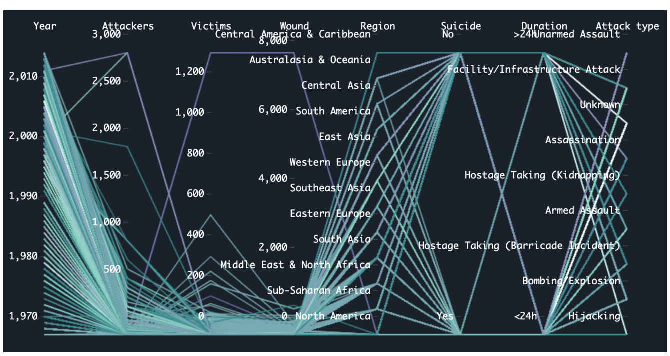

Fig. 5. Parallel Coordinates that represents the whole dataset, one line for each attack. The color encodes the number of the victims which is the same legend we use for the map

Parallel Coordinate view is used to represent multivariate data, it is ideal for comparing many tuples together and seeing the relationships between them, this is possible by crossing lines, each line correspond to a linear relation on some axis.
There are eight axis that we considered to be interested for the terrorism attack analysis.
Each line refers to one single point that represent a specific terrorism attack and the color of each line depends on the type of weapon used for the attack. We use the same colors of the stacked barchart.
There is a brushing event handler that give users the possibility to select a specific range on y-axis that he wants to analyze better so he can see all the changes of the other components for that specific range. He can also change the position of the chosen range on the y-axis in order to analyze other changes he is interested in.
When the user selects specific range on the chart the table next to it is updated with the relative information.
There is the possibility to exchange the order of the y-axis in order to let the user the opportunity to see better the data on the chart as he prefer.
This chart is related with the other chart: Map, Barchart and Table; when the user select a country or/and a year or a column on the Barchart, the Parallel Cordinates is updated with the relative data of that selection.

3.5 Table

Year	Attack	Victim	Woun	Suicide	Duration	Country	Region	Attack type	Weapon_type
2001	5	1384	8190	Yes	<24h	United States	North America	Hijacking	Vehicle (not to in
2001	5	1383	8191	Yes	<24h	United States	North America	Hijacking	Vehicle (not to in
2017	1	588	316	Yes	<24h	Somalia	Sub-Saharan Africa	Bombing/Explos	Explosives
2016	1	383	200	Yes	<24h	Iraq	Middle East & North	Bombing/Explos	Explosives
2017	25	311	127	No	<24h	Egypt	Middle East & North	Bombing/Explos	Explosives
2017	10	266	64	Yes	<24h	Afghanistan	South Asia	Bombing/Explos	Explosives
2007	2	250	750	Yes	<24h	Iraq	Middle East & North	Bombing/Explos	Explosives
2007	2	250	750	Yes	<24h	Iraq	Middle East & North	Bombing/Explos	Explosives
2001	5	190	106	Yes	<24h	United States	North America	Hijacking	Vehicle (not to in
2015	4	152	104	No	<24h	Kenya	Sub-Saharan Africa	Hostage Taking (	Explosives

Fig. 6. Table shows the 10 worst terrorism attack based on the number of the victims

We decided to rank the 10 worst terrorism attacks based on the number of the victims, this chart is related with other charts: Map, Barchart and Parallel Coordinates.
When a user choose a year or/and a country or/and a range in Parallel chart, the Table is updated with the worst 10 among all the available attacks happened in the world, this computation is based on the relative information chosen by the user, year and the country (if there aren't ten attacks it will be shown less).
We have decided to represent other two interesting data in addition to the attack's fields already showed in the Parallel Coordinates chart: the Country, where the attack is happened and the Duration, if it is lasted more than 24 hours.
There are two handlers: by hovering over a row it will be highlight the selected row and at the same time will be highlight the corresponding line in Parallel chart. By clicking a specific row it will be compute the 5 most similar attacks of the same type, computed by using the *Cosine Similarity* algorithm. At the same time Map, Scatterplot and Barcharts are updated relatively to the year of the chosen attack.

3.6 Cosine similarity

Attack	Victim	Wounc	Suicide	Country	Attack typ	spacial_distance
2	105	245	Yes	Turkey	Bombing/	0.999928
1	37	85	Yes	Iraq	Bombing/	0.999883
1	56	134	Yes	Afghanista	Bombing/	0.999823
1	42	100	Yes	Pakistan	Bombing/	0.999809
1	57	123	Yes	Pakistan	Bombing/	0.999803

Fig. 7. Table shows the 5 most similar attacks based on the selected one

When the user clicks on an attack in the Table chart, we show him the 5 most similar attacks, of the same type, to the chosen one. This is computed with *Cosine Similarity*, a measure of similarity between vectors which measures the cosine of the angle between them. The smaller the angle, the higher the similarity.
The cosine similarity between two vectors is defined as $\frac{u \cdot v}{||u||_2||v||_2}$
In our case we have to compare terrorism attacks so we represent each of them as a vector that has as values the number of victims, wounded, terrorist, attack type, and its event id in order to identify the specific event after the computation of similarity has done.
We found it was interesting based this attacks' similarity computation on this specific database's attributes because they are the most significant and representative fields of an attack.

All attacks' vectors are put into an array and thanks to the Python Library *Scipy*[**?**] it will be compute the cosine_similarity value between the chosen attack and the others, the computed value is added into the attacks vectors and allow them are sorted by cosine_similarity value. At the end we pick and show to the user the first 5 elements that will be the five most similar attacks to the chosen one.

4 ANALYTICS

We set up our backend with the python library `a Flask` that provides a server running in `localhost` and can easily comunicate with the views.

Thanks to python versatility in handling data, especially with the `Pandas`[6] library, we decided to use it to do intense computations such as PCA and cosine_similarity calculations, data filtering and data preparation for the map and the various graphs.

When the server starts, all these calculations are executed so each part is ready to display data. Since everything is connected, when the user makes a selection, everything has to be updated in response. All the choice the user makes activate the server with a `getJson()` request. The selected options are passed as parameters to the python programs, with are executed and write the results on csv files. Then a response is sent to the views and everything is updated accordingly.

The choice that triggers the `getJson()` are:

- Selection of a specific year with the slider
- Selection of one or more specific countries in the map
- Selection of a specific weapon type in the barchart
- Selection of a specific row in the parallel lines table

5 CONCLUSION

In this work we presented an valid tool that can be used as a support for the analysis and monitoring of the terrorism attacks events.
It is therefore interesting to analyze the distribution in the world of the attacks type or the weapons type used for it; observe where the highest number of deaths are concentrated or see all the details of the worst terrorism attacks.

It is possible to see how, in recent years, the types of attacks has changed significantly. For once the most common weapon type shifted from firearms to explosives, especially after 9/11 attacks. Moreover it's evident from the map that the most striked countries are the ones in the middle east, with Iraq totalising almost 10.000 victims in the last 40 years.

With all these information the user will hopefully become more aware of the world around him.

REFERENCES

[1] Color brewer. `http://colorbrewer2.org/?type=sequential&scheme=Reds&n=9`.
[2] Colorjs. `https://gka.github.io/palettes/`.
[3] Cosine similarity. `https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.distance.cosine.html`.
[4] D3 api reference. `https://github.com/d3/d3/blob/master/API.md`.
[5] Kaggle database. `https://www.kaggle.com/START-UMD/gtd`.
[6] Pandas. `https://pandas.pydata.org/pandas-docs/stable/`.
[7] Scikit-learn. `http://scikit-learn.org/stable/documentation.html`.