Predicting Terror Attacks?
A Data Story

Axel Nilsson, Nicolas Bollier, Elias Le Boudec, Enea Figini
Team 29

January 17, 2019

1 Introduction

Analysing terror organisations and predicting terror attacks is a subject of interest for national security organisations. From data on terrorist relationships and terror attacks, this project aims to assess whether terrorist relationships can be viewed as a social network, and to try to predict terror attacks locations from some known features.
To help reaching these goals, graph theory and data analysis tools provided by the course *A Network Tour Of Data Science* at EPFL will be used.

2 Exploring the Data

2.1 Relationships Dataset

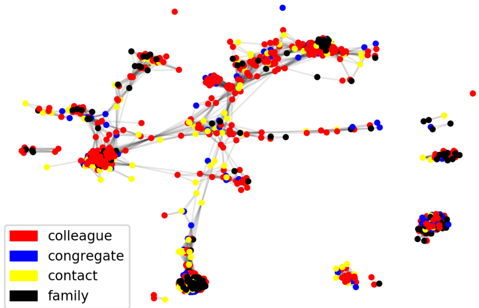
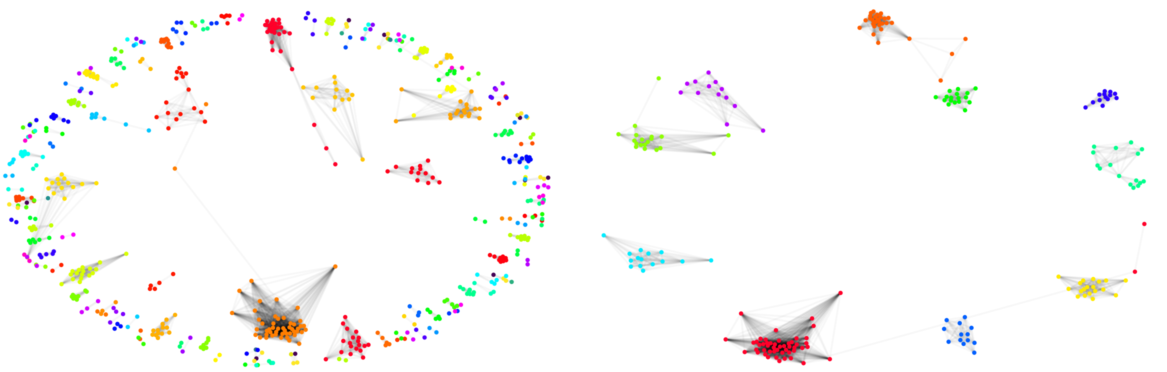


Figure 1: *Terrorist relation dataset graph, colouring by relation type*

The nodes of the network are relations between terrorists. These nodes are connected together if the relations share one terrorist.

2.2 Terror Attacks Dataset



(a) *Terror attacks location graph, colouring by component ID*

(b) *Ten biggest components from the terror attacks location graph*

Figure 2: *Graphs from the terror attacks dataset*

3 Comments on the Data

3.1 Terrorist Relations Dataset

3.2 Terror Attacks Dataset

Multiple issues have been found in this dataset:

Broadness The dataset comprises attacks ranging from 1969 to 2005 and spanning the entire globe. Simple and relevant explanations for the graph formation or properties are not likely to be found, since the mechanisms behind two different attacks can be entirely different.

Structure Half of the nodes are isolated, hence the topological information they carry in the graph is very limited. What is more, because of the transitivity relation described in Section 2.2, connected components are in most of the cases complete, hence isotropic.

Reliability Errors have been found in the data. For example nodes `Djibouti_Youth_-Movement_19900927` and `Armed_Islamic_Group_19950711` have been connected, whereas the first attack took place in Djibouti [1] and the second one in Paris [2]. Hence algorithms using the data must tolerate some error in order to avoid overfitting.

Incompleteness The dataset has been constructed from publicly available sources [3]. Because of the sensitivity of the data behind terrorist attacks and relationships, some of it is classified, making the dataset incomplete.

4 Predictions

The algorithm used to predict the terror attack location is the following:

1. From the dataset, select the 10 biggest connected components (“component” in what follows).
2. Sort the dataset by date of terror attack.
3. At this point, a component represents a location, and the nodes are the terror attacks in chronological order.
4. Select one node per component that is strongly connected to the others, the “lead” node.
5. Find the lead node $l^\star$ that is the most strongly linked to the new node (i.e. the next terror attack).
6. The predicted location of the next terror attack is the location of the component $l^\star$ belongs to.

The determination of the lead node uses the features vector supplied with each node, and a weighting function w. Let w be the application that returns a weight for each pair of nodes (n_1, n_2) in the graph $\mathcal{G}$, defined as

$$w : \mathcal{G}^2 \to \mathbb{R}^+ \tag{1}$$

$$(n_1, n_2) \mapsto f(|n_1 - n_2|) \tag{2}$$

where

$$|n_1 - n_2| = \|\text{features}(n_1) - \text{features}(n_2)\|_2 \tag{3}$$

features(n) is a binary features vector for each node n in $\mathcal{G}$ and $f : \mathbb{R}^+ \to \mathbb{R}^+$ is the node distance weighting. Examples for f are given in Table 1.
For each connected component, the lead node is determined as described below.

Algorithm 1: *Finding the lead node of a connected component with weighted edges*

Data: Connected component C
Result: Lead node n_l
Initialise $s(n)$ to zero. s is a dictionary mapping a score $s(n)$ for each node n
for *each edge e from C* **do**
    Let $e = (n_1, n_2)$, w be the weight of e
    $s(n_1) \leftarrow s(n_1) + w$
    $s(n_2) \leftarrow s(n_2) + w$
end
return $n_l = \arg\max_{n \in C} s(n)$

Finally, the prediction algorithm is presented below.

Algorithm 2: *Finding the predicted location of the next terror attack*

Data: Set of connected components $\{C_i^t\}$, $i = 1, \ldots, 10$, and the features vector of the next terror attack n_{t+1}, i.e. features(n_{t+1}), at each timestep t
Result: Location prediction p_t for time $t + 1$ at time t, at each timestep t
for *each timestep t* **do**
    Compute the lead component $l(C_i^t)$ for each component C_i^t
    $p_t = \arg\max_{i=1,\ldots,10} w(n_{t+1}, l(C_i^t))$
end

4.1 Justification

The design of prediction algorithm is motivated by the following aspects:

- The labels are taken into account by weighting the edges. This allows to completely ignore label signals on the graph and simplify the analysis.
- The determination of one lead node per component allows to smoothen local variations inside a component, thus making the prediction algorithm more robust.
- The choice of one lead component per component is justified by the fact that connected components are almost complete.

4.2 Results

Table 1: *Prediction accuracy for different node distance weightings f*

Weighting		**Best skewness ζ**	**Accuracy**
Gaussian:	$f(d) = e^{-d^2/\zeta} - e^{-1/\zeta}$	0.01	50.5 %
Log-Exponential:	$f(d) = e^{-d} \log\left(\frac{1+\zeta}{d+\zeta}\right)$	0.1	50 %
Linear:	$f(d) = 1 - d$	N.A.	47 %
Square:	$f(d) = \begin{cases} 1 & d < \zeta \\ 0 & \text{otherwise} \end{cases}$	0.1	43 %

5 Conclusion

Section 3.2 explains that the "Terror Attacks" dataset contains flaws that make it difficult to analyze.
However, the results in Table 1 show that predicting the location of an attack with its features is feasible even though the prediction is not very efficient. This result suggests that there is a link between the location of an attack and its characteristics (such as the type of the attack).

References

[1] Amnesty Internation Publications, 1 Easton Street, London, *Amnesty Internation Report 1991*, 1991.

[2] L'Obs, "Attentats de 1995 : chronologie." [fr] Online. https://bit.ly/2ASwNQP, last checked 17 January 2019, October 2007.

[3] B. Zhao, P. Sen, and L. Getoor, "Entity and Relationship Labeling in Affiliation Networks," *Proceedings of the 23rd International Conference on Machine Learning*, 2006.