International Journal of Computer Applications (0975 - 8887)
Volume 176 - No.35, July 2020

Unsupervised Hybrid Algorithm to Detect Anomalies for Predicting Terrorists Attacks

Francesco Curia
Department of Statistical Sciences, Sapienza University of Rome
Piazzale Aldo Moro 5, 00185, Rome, Italy

ABSTRACT

This work presents a hybrid approach for unsupervised algorithms (**UHA**), in order to extract information and patterns from data concerning terrorist attacks. The reference data are those of the Global Terrorism Database. The work presents an approach based on autoencoders and k-modes type clustering. The results obtained are examined through some metrics presented in the article and it is also considered methodologically how to determine a robust threshold for anomaly detection problems.

Keywords

Terrorism, Unsupervised Learning, Clustering, Autoencoders, Optimization

1. INTRODUCTION

Terrorism is a real threat, which does not exclude any country from the danger of terrorist attacks. From Asia, to Africa, to Europe, the terrorist groups are more and more numerous, from the biggest and sadly more famous groups like the ISIS or Al-Qaeda to small local groups, to lone wolves to the so called **foreign fighters**, the cities are constantly in danger. Technological and scientific progress has provided the tools to counter these threats, artificial intelligence, specifically machine learning algorithms, currently play a major role in this field, and that the military and intelligence systems of the various countries they are endowing these tools is not something to be surprised about. The fight against terrorism cannot be fought only in the field with conventional weapons, but must therefore be preventively countered by designing plausible and more or less probable scenarios requiring the use of sophisticated mathematical-statistical models and the development of complex software that they can in advance provide the necessary moves to counter the terrorist threat. Science placed at the service of governments can provide the analytical support necessary to set up operational actions; in this work, in addition to an overview of the state of the art regarding the modeling used to predict terrorist phenomena, a new hybrid method of analysis is proposed, this method is based on the detection of anomalies within data patterns, specifically the use of unsupervised neural networks such as **autoencorders**, together with the use of a clustering procedure, however based on multiclass and categorical variables as the dataset being studied has variables of this type. To approach this clustering problem from a methodological point of view, a method called **k-modes** will be used. Specifically, a clustering will be carried out in order to assign a label to each event (record) for the group to which the event will be assigned and subsequently, an autoencoder will be applied, whose reconstruction error obtained will be correlated with the information present in the reference cluster, in order to understand if there is statistical evidence within the groups that have a certain value of the reconstruction error. Under the assumption that there is not necessarily a target variable that indicates whether an attack has occurred or not, try to find some data based on the concept of anomaly in the data (therefore without labeling) of something that is not conventionally considered in the norm. It is obvious that these tools must serve the decision maker in order to be able to make considered choices in an analysis phase and therefore it is necessary to give a definition of anomaly, so what is meant by anomalous and how to determine an anomaly threshold, a question that will be addressed in the following paragraphs. The data of the experiment are the well-known ones accessible via the internet, i.e. those of the **Global Terrorism Database** (GTD), in which there are many variables of strong interest, one of which, mostly used by researchers, is the one that indicates whether an attack was committed, in the sense of having been signed or not, and this variable given the dichotomous nature of the event, assumes binary value 0-1. In this work it will be exclude this information, trying to understand if it is possible to extract from the features present in the data set, the information necessary to understand if a given event can be considered at high risk of terrorism with a certain degree of probability. The contributions of this work therefore considering the current state of the art are those of considering, starting from originally labeled data, the possibility of treating this problem as unsupervised, since in most cases the data are not always labeled and fortunately the proportion on the available data of positive events (i.e successful attack) are in much lower numbers, to make the sample unbalanced.

2. LITERACY REVIEW

The state of the art regarding the applications of machine learning and data mining techniques in the fight and the prevention of terrorism, it has made noticeable leaps and bounds, several works have been produced in recent years, many of them set themselves the task of trying to predict with a certain accuracy the probability that a given terrorist event could manifest itself in a certain place and time. This task is difficult as it can obviously be understood, the data instances that allow the predictive analysis of the mentioned phenomena, fortunately, are not very large and the time evolution becomes a non negligible component, in the sense that given the

evolution of the systems of contrast and the fight against terrorism, the means available to intelligence from around the world, today, are not those of 50 years ago, therefore an event of the time cannot be equated with one of today, because unfortunately also terrorism today it makes use of technological progress. Many of the works that make up the state of the art regarding these themes, and also the present work here, make use of data from the well-known Global Terrorism Database published by the University of Maryland [1]. In Kumar, Mazzara et all. [2] the authors apply different data mining algorithms aimed at pattern recognition within the data, reaching an accuracy that varies between 90% and 95%, specifically applying the 'class attack responsibilities' variable which can take on value $claimed$, $not\text{-}claimed$ and $anonymous$: Lazy classifier IBK linear NN, Lazy classifier IBK Filtered Neighbor Search, Lazy classifier IBK, Ball Tree, Lazy classifier K-star, Decision Tree Random Forest, Multilayer Perceptron, Multiclass Classifier and Nave Bayes. The target object of the study refers to the fact that various attacks may or may not be claimed by terrorist organizations. In the work of Bang, Basuchoudhary et all. [3] instead a series of machine learning algorithms are applied to evaluate the probability (or risk) for the different countries studied in the database (GTD) and crossing with other data sources such as (CNTS; Banks, 2015), (DPI; Cruz et al., 2016) and (ICRG; PRS Group, 2015), in which hypotheses are made on latent variables (such as violence for example) not directly observable, through the application of different techniques such as negative binomial regression and regression of Poisson, classification trees, random forests and neural networks. The authors also propose an approach based on indicators that can define the weakness of institutional systems, for example in the political, military and psychological fields that can be considered as just a weak point against possible terrostic attacks. As a target variable, they consider the number of attacks, i.e. the $\mathbf{attacks\ terror}$ variable present in the GTD, thus applying eight models, obtaining an MSE of 69% and 70% respectively with the Bagging and Random Forests models. In another work, by Verma, Malhotra et.al. [4] consider three different predictive models, one regarding the type of attack, another as a target considers the region of attack and the third considers the type of weapon used, using data range from 2013 to 2016 and as classifiers used SVM, Neural Networks, Nave Bayes and Random Forest further uses a linear regression to evaluate correlations between attacks by ANOVA. The third model seems to be the most performing with a very low error rate of 9%. Saha S. et.al. [5] apply different algorithms obtaining good results in terms of accuracy, for attack types they reach 79% for the type of weapon also used 86% using Random Forests, while Coffman T.R., Marcus S.E. [6] they use social network analysis for prediction if a particular subject is a terrorist or not, obtaining an accuracy of 86%. The authors Sachan A., Roy D. in [7] analyze 43355 terrorist events by applying supervised learning techniques such as Support Vector Machine and Random Forest obtaining good results in terms of classification. The literature in this particular area is becoming very numerous and therefore several researchers publish works of considerable interest in this field. In Adnan and Rafi [8], the authors also propose a work based on unsupervised learning, specifically using a procedure defined by the co-clustering model, basing the clustering on textual data by extracting the characteristics from the GTD data, specifically considering bilateral data; bilateral data can be analyzed by describing the connections between two different entities. Co-clustering is a method that allows the rows and columns of a matrix to be grouped simultaneously, this method was developed by Hartigan in 1972 [9] and has been widely used. Another approach, in the work of Skillicorn and Leuprecht [10], is based on a Singular Value Decomposition clustering procedure, analyzing three types of data, including the well known GTD, exploring the impact of individual attributes or fields by superimposing the visualizations. of clusters.

3. METHODOLOGY

Part of the works examined in the previous paragraph mainly refer to supervised learning techniques, that is, for a set of characteristics (variables) detected on the object of interest, there is a so-called target variable, or in classical statistical terminology, a dependent variable $y_i, i = 1, ..., n$ represented by the pair vector (x_i, y_i), or for each observation i, the variable depends on n-variables $x_i, i = 1, ...n$, expressible as $y_i = f(x_i)$, where the form functional of the $f(\cdot)$ can be linear or non-linear. The algorithms used in the state of the art that has been examined previously, vary from the classical non-linear models as in [3] in which neural networks are used or linear models as in [4]. The functional form also depends on the loss function chosen for the update of the weights, for example a function of quadratic type, or a form based on the logarithm, all these choices are clearly made by the researcher in the experiment phase, looking for the best functional relationship which describes the phenomenon. For the second part of works examined for the unsupervised learning algorithms the same thing happens, usually for clustering based approaches, also here the shape of the distances used can be different, euclidean distances can be used, or absolute values, the nature changes also according to the typology of variables that it's present, that is, binary, multiclass, continuous variables and so on. In the approach used in this work, the modeling choice falls on a type of algorithm based on neural networks, but of an unsupervised type, namely the $autoencoders$ [16], a class of algorithms that are used in image recognition, in the reduction of the number of variables [17] or in the anomaly detection [19], as for example occurs with the PCA and not least in the selection of the variables (Han et al. [20]). Therefore in this phase, mathematical optimization plays a major role, since it is well known that both the updating of the algorithm parameters, also as the estimation of the parameters in a regression model, takes place through the minimization of a function , defined as $\mathbf{loss}$. Therefore, depending on the nature of the functional form that will be adopted, the optimization problem will also change, if a quadratic objective function is used a different optimization method will be used with respect to whether the problem is linear, or linear if the problem is binary, for example when using an autoencoder with binary variables, a cross entropy function is usually used. The activation functions, in the field of neural networks are another interesting point, depending on the choice, which is a sigmoid function, Relu or Tanh, the nature of the mathematical programming problem changes. In the hybrid approach proposed in this work, for the clustering part, a method based on k-modes will be used, an extension of the well known k-means; the k-modes algorithm was proposed in a 1998 article by Huang, Z [11]. Instead of using euclidean distance, in this approach, mode is used. Mode is an element carrier that minimizes the differences between the carrier itself and each data object. There are many modes as the number of necessary clusters that are selected. Another interesting approach developed subsequently is a mix between k-means and k-modes, for numeric and categorical data, called k-prototypes [12].

3.1 Contributions

The main contribution of this work mainly concerns a fundamental aspect, the application of a hybrid method, i.e. combining two unsupervised machine learning methods in order to detect

anomalous patterns in the data or intrinsic structures that can explain a behavior considered $suspicious$. The methods proposed in the articles mentioned above, such as the current state of the art approach to the phenomenon, always see a known target variable to be predicted, compared to a set of characteristics present in the data and this in the reality of the decision science is not always possible, there are decisions that must be made in a time window that does not always allow you to have all the information available. Assuming that it is not known whether an attack has had a positive or negative outcome, therefore the knowledge of a binary variable that expresses this condition, the approach proposed here aims to create a plausible anomaly score to identify suspicious behavior. In particular, considering the GTD data and the above works, it will be then compared the score obtained, through the definition of an evaluation metric, with the binary value of the variable **terrorist attacks** which assumes a value of 1 if you are a terrorist event occurred and 0 otherwise.

Aspects fundamentals of this work:

· Hybrid combination of unsupervised methods
· Cross-analysis between reconstruction error and cluster assignment
·Anomalies thresholds determination

4. GENERAL FRAMEWORK

As regards the general framework of the work, it consists of a first part which concerns data processing, or a first part which concerns the application of an unsupervised neural network of the autoencoder type (part 4.1) , in order to extract anomalies through the error reconstruction between input and output, through scaling of continuous variables and a selection of the variables that are considered most important. In parallel, a k-modes (part 4.2) is applied to categorical data, in order to create groups on which to investigate, by crossing the results through the output of the autoencoder. In 4.3 it's presented the pseudocode of the algorithm proposed that combines the two previously steps. Once this information is obtained, an anomaly threshold is determinted (part 6), which serves to determine which values of the reconstruction error exceed the threshold to define a target to be compared with the $\mathbf{success}$ variable in order to subsequently make. All phases of the work are supported by plots that show the results.

4.1 Autoencoders

An autoencoder is a type of neural network which provides an input and an outuput layer with one or more intermediate layers (fig. 1), in which it is try to reconstruct the input value by encoding the data to obtain an estimate in the output, giving priority to the most relevant aspects of the data to be encoded, where the output layer has the same number of nodes (neurons) as the input level and in order to reconstruct its inputs (in which the distance between input and output is minimized) instead of predicting the value of a target variable. This technique has been used for decades, developed in the 80s by Hinton et al., [16] [17] [18] and the most traditional application was the reduction of dimensionality or the learning of functionalities, but more recently the concept of autoencoder has become most widely used for learning generative data models. Autoencoders are trained to minimize reconstruction errors (such as squared errors), often referred to as $loss$ and as mentioned above, the training of an autoencoder is performed via error backpropagation, just like a normal feedforward neural network. Starting

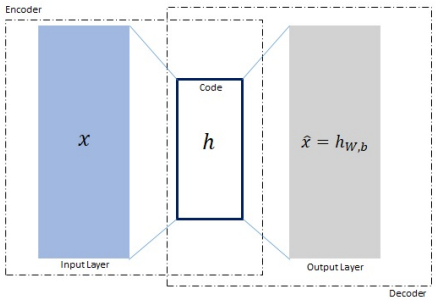

Fig. 1. Architectures

from a set of unlabeled training data $(x_1, x_2, x_3, ..., x_n)$, x_i is N-dimensional, $x_{(i)} \in \Re^N$ and defining the functions $\psi(\cdot)$ and $\phi(\cdot)$ as a function that maps (called encoder and decoder mappings) the inputs: the idea is that the inference process of mapping observations x_i to the corresponding latent variables such that $\psi : \mathcal{X} \to \mathcal{F}$ and $\phi : \mathcal{F} \to \mathcal{X}$ the encoder step of an autoencoder takes the input $\mathbf{x} \in R^N = \mathcal{X}$ and maps it to $\mathbf{h} \in R^p = \mathcal{F}$ where h is the activation function's: define as follow $\mathbf{h} = \sigma(\mathbf{W}\mathbf{x} + \mathbf{b})$, h is usually referred to as code, latent variables, or latent representation. Here, σ is an activation function (i.e Tanh, ReLU, softmax, sigmoid). $\mathbf{W}$ is a weight matrix and $\mathbf{b}$ is a bias vector, for hypothesis follows a gaussian distributions. Weights and biases are usually initialized randomly, and then updated iteratively during training through backpropagation techniques. After that, the decoder step of the autoencoder maps $\mathbf{h}$ to the reconstruction $\hat{\mathbf{x}}$ of the same shape as $\mathbf{x} : \hat{\mathbf{x}} = \hat{\sigma}(\hat{\mathbf{W}} \cdot \mathbf{h} + \hat{\mathbf{b}})$. The train is performed solving the following quadratic optimization problem

$$\min_{x} \quad \|\mathbf{x} - \hat{\mathbf{x}}\|^2 \tag{1}$$

assuming that the differences between inputs and outputs follow a normal distribution.

Architecture	
Number of layers:	7
Number of neurons:	74,50,30,50,74
Activate Function:	Rectifier
Distribution:	Gaussian
Epochs:	100
Loss:	Quadratic

4.2 Clustering: k-modes

In the formulation proposed by Huang [12], the extension of the k-means to categorical variables can be formalized in the following way, considering X and Y two categorical objects described by m categorical attributes, the dissimilarity measure between the variables X and Y can be defined by

$$d(X, Y) = \sum_{j=1}^{m} \delta(x_j, y_j) \tag{2}$$

where

$$\delta(x_i, y_i) = \begin{cases} 0 & \text{if} \quad (x_i = y_i) \\ 1 & \text{otherwise} \end{cases} \tag{3}$$

Now consider a vector M of categorical objects described by m attributes $A_1, A_2, ..., A_m$ the mode of $M = [X_1, X_2, ..., X_m]$ is the vector object $Q = [q_1, q_2, ..., q_m]$ which minimizes the following unconstrained optimization problem:

$$\min_{x} \quad \sum_{X \in M_i} \sum_{i}^{n} d(X_i, Q) \tag{4}$$

where here Q is the mode of the cluster M and the solution to the above problem solves the problem of partitioning a set of n -objects described by m -categorical attributes in k -groups $M_1, M_2, .., M_k$. In this experiment, after applying the k-modes to the categorical data of the dataset, it's obtained as the optimal number of groups, k = 3, with respect to this result it can be now correlate the results of the autoencoder with these of the clustering.

4.3 UHA: Unsupervised Hybrid Algorithm

In this subsection the pseudocode of the proposed algorithm is presented.

$\mathcal{D}$, dataset
i, i-th sample
X_i, i-th features: $X_i \in \mathcal{D}$
$\tilde{C}_{i,k}$, i-th sample in the cluster k
ϵ_i, reconstruction error for i-th sample

Data: Dataset $\mathcal{D}$ with k-features, X_k from GTD,k=1,...,N
Result: Dataset $\tilde{\mathcal{D}}$ with clusters labels and reconstruction error target $\hat{\epsilon}$
initialization;
for *i in $\mathcal{D}$, i=1,...,N* **do**
  compute k-modes, and take $\mathcal{C}_{i,k}$ clusters, k=1,...,M, for i-th sample, i=1,...,N
end
for *i in $\mathcal{D}$* **do**
  compute autoencoder and take reconstruction error ϵ_i, i=1,...,N
end
for *$\mathcal{C}_k$, k = 1,...,M* **do**
  $\tilde{C}_{i,k} = \mathcal{C}_{i,k} \cap \epsilon_i$, $\forall$ i=1,...,N, $\forall$ k=1,...,M
  for *i in $\tilde{C}_{i,k}$* **do**
    if $\epsilon_i \geq$ *threshold* **then**
      $1 \rightarrow \hat{\epsilon}_i$, $\forall i$=1,...,N
    else
      $0 \rightarrow \hat{\epsilon}_i$, $\forall i$=1,...,N
    end
  end
  $\tilde{\mathcal{D}} = (X_i, \tilde{C}_{i,k}, \hat{\epsilon}_i)$
end

Algorithm 1: UHA

5. DATA PROCESSING

The data available for the experiment, mentioned above, are those of the global database on terrorism managed by the national consortium for the study of terrorism and of responses to terrorism (START), this database is made up of various items listed previously put together. The observations occurred at the time period 1970-2017, for the demonstrative purpose of this work the subset

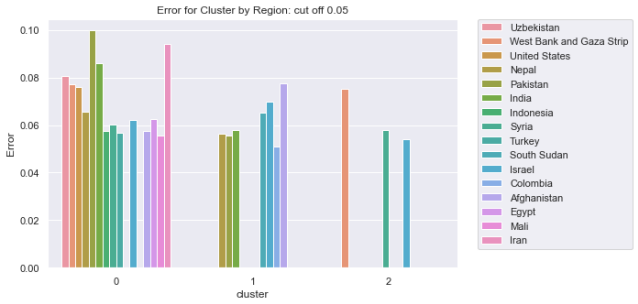

Fig. 2. Error by cluster: Regions

will be considered from 2000 to 2017, since temporal dynamics, cultures and technological progress are inevitably different and do not make much sense considered as explaining a context so temporally distant and different from the current state of the military situation, increasingly in possession of sophisticated means of preventing and combating terrorism, both for internal and external threats. Inside there are 132 attributes of different nature, both numerical (binary, multiclass, continuous) and both textual in nature, variables on location, tactics, perpetrators, targets, and outcomes. For the application of autoencoder it's necessary for analysis and modeling, keep only the variables of a numerical nature and the ID of the event are considered, which is nothing but the composition of the data in which the event occurred plus a part describing the progressive number of the case for the given day. Choice of the subset of requested variables, there is a part of the data cleaning, in which the missing values are imputed, for the different ordinary variables the distribution mode is used and in some cases the median (more robust in the presence of anomalous values) and for the continuous values the missing value was imputed by mean. A subset of 70 features was then chosen for the autoencoder training and the chosen architecture consists of 5 intermediate layers to those of input and output, as a function of loss for evaluation a quadratic type function was chosen in order to reconstruct the error as a difference between input and estimated value. As a function of activation in the first instance a ReLU type function was chosen to then test a Tanh type function. To the features a normalization of the values has been applied through the following scaling

$$x_{norm} = \frac{x_i - min(x)}{max(x) - min(x)} \tag{5}$$

where x_{norm} is the normalized value for observation i, x_i is the original value for observation i, and $min(x)$ and $max(x)$ represent the minimum and maximum values of features $\mathbf{x} = (x_1, ..., x_n)$, respectively. As regards the application of k-modes, categorical, ordinal, binary textual data are considered.

5.1 Features importance

Through the method proposed by Gedeon [13], a selection of the variables was carried out with respect to the importance and the relative score resulting from the following formula

$$P_{jk} = \frac{|w_{jk}|}{\sum_{r=1}^{n,h} |w_{rk}|} \tag{6}$$

where w_{jk} is the connection weight between the input neuron j and the hidden neuron k, w_{rk} is the connection weight between the hidden neuron j and the output neuron k, and $\sum_{r=1}^{n,h} |w_{rk}|$ is the

sum of the connection weights between the N input neurons and the hidden neuron j. P_{jk} represents the percentage of influence of the input variable on the output. Below is showed the table with the first twenty variables with respect to the importance score with cut-off 0.60.

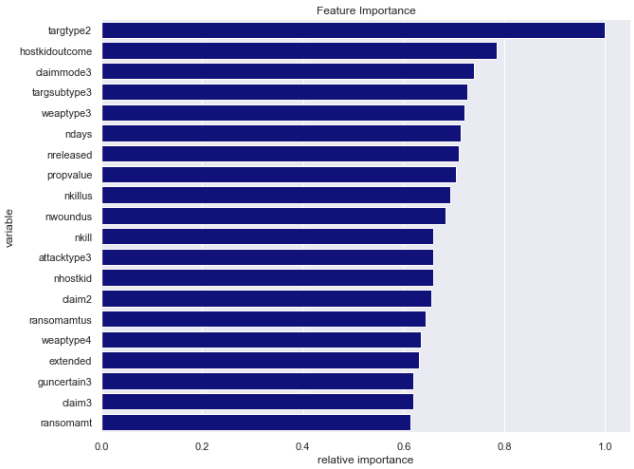

Fig. 3. Features Importance

From (fig. 3) it is possible to note that respect to the score of the importance of the variables, obtained with (6), it's possible note so the most important features relevant in ther data are respectively **targtype2**: the values of this variable refer to military weapons, military aircraft, maritime military personnel and non-combatant personnel, target of the attack by terrorists, **hostkidoutcome**: this feature refers to the possible fate of the hostages and victims of the kidnapping, it takes on seven values and i.e. **claimmode3**: this field refers to the responsibilities inherent to the attack, for example among the values that this variable can be assumed finding email, video and letters.

Fig. 4. Error by cluster: Target Type

6. ANOMALIES THRESHOLD'S DETERMINATION

Perhaps the most important part of an analysis of anomalies is certainly the determination of the threshold that can define what is

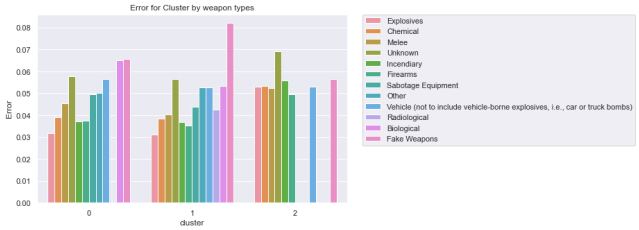

Fig. 5. Error by cluster: Weapons

anomalous with respect to what is not. In this regard, several authors have proposed *scores* for evaluating the anomaly, in order to quantify the degree of anomaly of the phenomenon of interest. Zhao and Saligrama [14], have proposed an evaluation method based on the KNN algorithm, in which anomalies are declared each time the score of a test the sample drops below a predetermined level α, which should be the desired level of false alarm, instead in Gao and Tan [15], it's proposed an interesting method based on the conversion in probability value of the anomaly detection output, or they assume that the a posterior probabilities assumes that the posterior probabilities follow a logistic sigmoid function and the probability estimates allow us to select the appropriate threshold to declare outliers using a Bayesian risk model. Once the k-clusters and reconstruction errors ϵ_i have been obtained, for each sample i, as described in the UHA algorithm, one of the most important steps is how to determine the threshold.

Fig. 6. Error on train and test set for each cluster

From figure 6 it is possible to see the result of the application of the autoencoder on train and test for each cluster, the values that it's are been considered anomalous by setting an initial threshold at $\hat{a} = 0.05$ obtained by application of (7), are values that have less concentration than those more concentrated than theoretically have characteristics that make the score of anomaly less than the others. In figure 7 instead the rappresentation it's by years. In alls two plots it's possibile observe the robustness of the threshold.

6.1 Statistical optimization of threshold's

The main interest is in the determination of which is the best threshold to capture the anomalies present in the data, for this purpose it is considered a certain value a and given the hypotheses of normality on the anomaly scores, the interest is in the calculous of the probability that a value ϵ_i is greater than the value of the threshold a (therefore anomalous) with a certain level of confidence given the uncertainty of the case. In formulas:

$$P\left(\epsilon_i > a\right) = \alpha$$

$\alpha \in (0, 1)$ and $P(\cdot)$ is the probability measure under normality hypothesis of ϵ_i. The goal is to determine the best value of a such that the probability of being above a certain threshold has a certain predetermined value, fixing α it's possible write, standardizing

$$P\left(\frac{\epsilon_i - \mu_\epsilon}{\sigma_\epsilon} > \frac{a - \mu_\epsilon}{\sigma_\epsilon}\right) = \alpha$$

$$P\left(Z_\epsilon > \frac{a - \mu_\epsilon}{\sigma_\epsilon}\right) = \alpha$$

To determine a it is imposed that

$$\frac{a - \mu_\epsilon}{\sigma_\epsilon} = Z_\alpha$$

where Z_α is the α-quantile of normal distribution at level α, then can be therefore derived by

$$\hat{a} = Z_\alpha \cdot \hat{\sigma}_\epsilon + \hat{\mu}_\epsilon \tag{7}$$

With the available data, compared to this experiment, the determined threshold is equal to 0.05, it is possible to see the results obtained through the figure 6 and 7. On the other hand, by approaching the problem of determining the threshold through the addition of the temporal parameter t (can be day, months or years, or also hour, minute and second, depends from the data sampling in the dataset), it is possible see that in plot 7, compared to the year, that the threshold as well as becoming obviously dynamic, is able to better capture the oscillations due to the intra time variation . This solution seems to be better, since assuming the same distribution and the same constant anomaly threshold value in the long run is almost misleading. Always from the same plot note that some events that in the plot with static threshold are not *captured*, with the dynamic threshold instead they are captured with much more precision. Therefore can be written (7) as a function of time, which must be defined on the basis of the type of approach used, if on a monthly, annual, daily basis, therefore by the availability and sampling of the data.

$$a(t) = Z_\alpha \cdot \sigma_\epsilon(t) + \mu_\epsilon(t) \tag{8}$$

with

$$\sigma(t) = \frac{1}{T}\sum_{t=1}^{T} \sigma_t \tag{9}$$

and

$$\mu(t) = \frac{1}{T}\sum_{t=1}^{T} \mu_t \tag{10}$$

In this case the Z_α value becomes a parameter that can either be estimated or empirically set or found through the following optimization problem presented in the next paragraph, i.e. between [5-15], in figures 7 it's was settled on 15.

6.2 Optimization of threshold's

Once estimated $\mu(t)$ and $\sigma(t)$ on temporal based selection, can be done the following reasoning, or think that the value Z_α maximizes the number of events that are correctly predicted by the autoencoder, given the anomaly threshold, there are the following relationships:

$$\hat{Y}_i = \begin{cases} 1 & \text{if} \quad \epsilon_i > Z_\alpha \cdot \sigma_i + \mu_i \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

where $\sum_i^N \hat{Y}_i$ is the sum of all class 1 events predicted by the model, and Y is the true value (i.e. variable $\mathbf{success}$) which take binary values, so $\sum_i^N Y_i$ is the sum of positive values and N is the number of samples in the dataset. Therefore may be interesting in maximizing the following quantity:

$$\max_i \quad \sum_{i \in N} (Y_i - \hat{Y}_i) \tag{12}$$

The (12) is equivalent to saying that you want the number of cases when the target variable is equal to 1, equal to that predicted by the model. As can be shown, the maximization is on Z_α, this from the relation (11). Through the following reasoning can be then thinked that the following formulation also applies:

$$\max_i \quad \sum_{i \in N} \hat{Y}_{1_{(Y_i=1)}} \tag{13}$$

Equivalent to

$$\max_i \quad \sum_{i \in N} (\epsilon_i - Z_\alpha) \tag{14}$$

But for (11) then can be even consider only the values equal to 1 of the $\hat{Y}$, fixing $\mu = 0$ and $\sigma = 1$ and considering only the predicted values of class 1, requesting this implies maximizing on ϵ considering as constraint the fact of wanting an accuracy, that is, that the number of correctly classified cases is as high as possible and for the maximization problem the heuristic solution for the (14) could be take $Z_\alpha^* = \beta \cdot \epsilon_{max}$, β is the level of accuracy to be achieved in the classification, i.e. for at least 70% of accuracy it's $\beta = 70$ (fig. 7, dynamic threshold).

7. DISCUSSION AND GENERAL RESULTS

7.1 Evaluation metrics

Considering the following matrix $M_{n \times n}$, in a binary problems with $n = 2$, where M is the confusion matrix obtained after the comparison between the values of the feature definined $\mathbf{success}$ that indicated if an event it's happen or no (0-1) and the variable obtained by solving (11) and reconstruction error from autoencoder output, so it can be define a binary variable and use the classic evaluation techniques well known for binary classification problems. The metrics defined above are well know in classification problems and seem a right way to evaluate the results of the unsupervised problem. The accuracy for this problem it's 86.5%. Graphically (fig. 9) note that

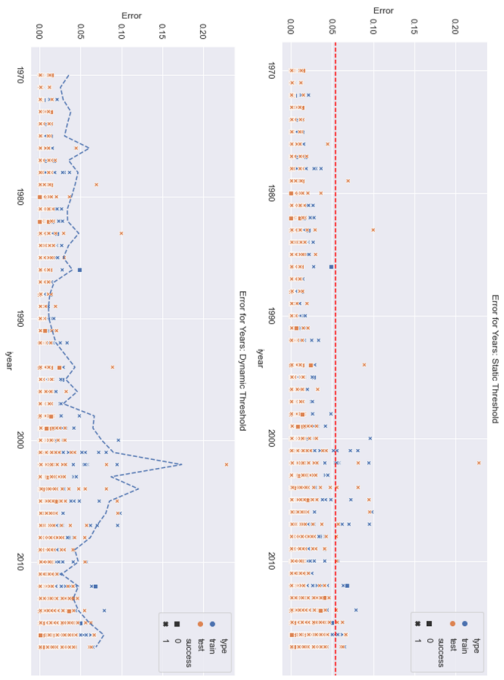

Fig. 7. Error on Train and Test Set: Years

the results obtained are very interesting in terms of precision and recall metrics, respectively 88% and 97%, also in the light of the approach used to determine the anomaly threshold and from the comparison between static and dynamic threshold (fig. 7). The approach is clearly multidisciplinary, as both statistics and mathematical optimization come into play, i.e. the heart of machine learning and deep learning. By the confusion matrix (fig. 8) it's possible see the results obtained, the results are very **robust**, like the value of the **F1-Score** statistic equal to 0.91, therefore the implemented algorithm has brought statistically significant results in the detection of anomalies and the labeling of the binary problem, while in the plots relating to cluster analysis with k-modes (fig. 2,4,5) interesting relationships emerge combining the groups with the value of the autoencoder reconstruction error.

7.2 Scope and future works

In this work an algorithm has been presented that combines two classic and many robust methodologies. The strength in combining two unsupervised methods together, to obtain a supervised classifier, optimizing an anomaly threshold could be the starting point or the evolution of new possibilities, both methodological and application. It is clear that the theme covered in this work is very complex, the main purpose of the application is to understand if, starting from some information (features), without target, it is possible to reconstruct (reconstruction error) the patterns that can identify a certain noise (risk od attacks) in the data. In order to evaluate the methodology, the binary variable indicating if an event occurred or not was initially excluded, and subsequently, having obtained the output of the UHA algorithm, it was used to validate the result of the hybrid classifier. What is certain is that this method of combining classifiers, rather than regressors, could fall within the circle of ensemble methods and this is well know, therefore, the presented work could

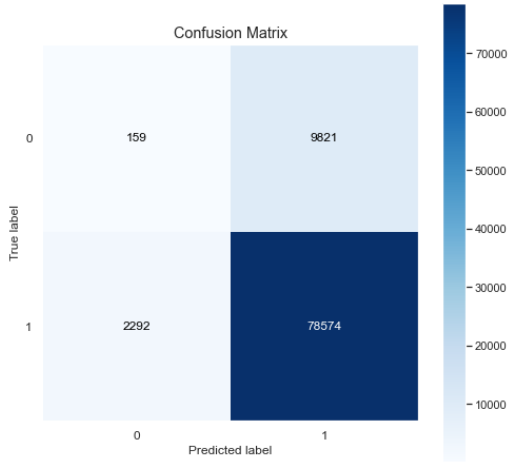

Fig. 8. Confusion Matrix

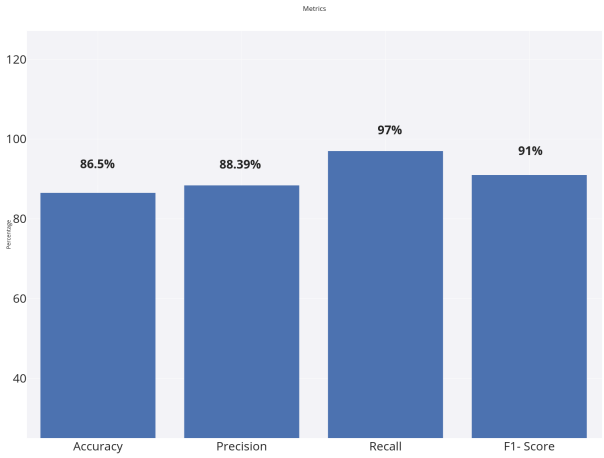

Fig. 9. Evaluation metrics

become of interest and future investigations, improving and testing different methods.

Conflict of interest

The authors declare that they have no conflict of interest.

8. REFERENCES

[1] Global Terrorism Database (GTD), http://www.start.umd.edu/gtd, 2017.

[2] Kumar, V., Mazzara, M., Lee, J., "A Conjoint Application of Data Mining Techniques for Analysis of Global Terrorist Attacks - Prevention and Prediction for Combating Terrorism", Published in SEDA 2018 Mathematics, Computer Science

[3] Bang, J.T., Basuchoudhary, A., David, A. J., Mitra, A. "Predicting Terrorism: A Machine Learning Approach", Working Paper, November 2017

[4] Verma, C., Malhotra, S., Verma, S. Verma, V.,"Predictive Modeling of Terrorist Attacks Using Machine Learn-

ing",International Journal of Pure and Applied Mathematics 119(15), June 2018

[5] Saha, S., Kurian, A., Basu, A., Aladi, H., "Future Terrorist Attack Prediction using Machine Learning Techniques", Working Paper, May 2017

[6] Ozgul F., Erdem Z., Bowerman C., "Prediction of Unsolved Terrorist Attacks Using Group Detection Algorithms", Intelligence and Security Informatics. PAISI 2009.

[7] Sachan, A., Roy, D., "TGPM: Terrorist Group Prediction Model for Counter Terrorism", International Journal of Computer Applications 44(10):49-52 , April 2012

[8] Adnan, M., Rafi, M.,"Extracting patterns from Global Terrorist Dataset (GTD) Using Co-Clustering approach", Journal of Independent Studies and Research-Computing Volume 13 Issue 1 January 2015

[9] Hartigan J.A, . "Direct clustering of a data matrix". Journal of the American Statistical Association. 67 (337): 1239. 1972

[10] Skillicorn, D.B.,Leuprecht, C., "Clustering Heterogeneous Semi-Structured Social Science Datasets", Procedia Computer Science Volume 51, 2015, Pages 29082912

[11] Huang, Z., "Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values", Data Mining and Knowledge Discovery 2, 283304 (1998)

[12] Huang, Z., "Clustering large data sets with mixed numeric and categorical values", Proceedings of the First Pacific Asia Knowledge Discovery and Data Mining Conference, Singapore, pp. 21-34, 1997

[13] Gedeon, T.D., "Data mining of inputs: analysing magnitude and functional measures", International Journal Of Neural System 1997 Apr;8(2), 209-218

[14] Zhao, M., Saligrama, V., "Anomaly Detection with Score functions based on Nearest Neighbor Graphs", Advances in Neural Information Processing Systems 22 (NIPS 2009)

[15] Gao, J., Tan, P.N., "Converting Output Scores from Outlier Detection Algorithms into Probability Estimates", Proceedings of the 6th IEEE International Conference on Data Mining (ICDM 2006), 18-22 December 2006, Hong Kong, China

[16] Rumelhart, D.E., Hinton G.E., and Williams, R.J. "Learning internal representations by error propagation". In Parallel Distributed Processing. Vol 1,Foundations. MIT Press, Cambridge, MA, 1986

[17] Hinton, G.E., Salakhutdinov, R.R., "Reducing the Dimensionality of Data with Neural Networks",Science 28 Jul 2006,Vol. 313, Issue 5786, pp. 504-507

[18] Hinton G.E., Krizhevsky A., Wang S.D. (2011) Transforming Auto-Encoders. In: Honkela T., Duch W., Girolami M., Kaski S. (eds) Artificial Neural Networks and Machine Learning ICANN 2011. ICANN 2011. Lecture Notes in Computer Science, vol 6791. Springer, Berlin, Heidelberg

[19] Oh, D.Y., Yun, D.,"Residual Error Based Anomaly Detection Using Auto-Encoder in SMD Machine Sound",Sensors (Basel) 2018 May; 18(5): 1308. Published online 2018 Apr 24

[20] K. Han, Y. Wang, C. Zhang, C. Li and C. Xu, "Autoencoder Inspired Unsupervised Feature Selection," 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, 2018, pp. 2941-2945