11-2-2018

Visualization, Feature Selection, Machine Learning: Identifying the Responsible Group for Extreme Acts of Violence

Mahdi Hashemi
University of Nebraska at Omaha

Margeret A. Hall
University of Nebraska at Omaha, mahall@unomaha.edu

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 15, 2018, accepted October 27, 2018, date of publication November 2, 2018,
date of current version December 18, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2879056

Visualization, Feature Selection, Machine Learning: Identifying the Responsible Group for Extreme Acts of Violence

MAHDI HASHEMI AND MARGERET HALL
College of Information Science and Technology, University of Nebraska Omaha, Omaha, NE 68182 USA
Corresponding author: Mahdi Hashemi (m.hashemi1987@gmail.com)

ABSTRACT The toll of human casualties and psychological impacts on societies make any study on violent extremism worthwhile, let alone attempting to detect patterns among them. This paper is an effort to predict which violent extremist organization (VEO), among 14 currently active ones throughout the world, is responsible for a violent act based on 14 features, including its human and structural tolls, its target type and value, intelligence, and weapons utilized in the attack. Three main steps in our paper include: 1) the visualization of the violent acts through linear and non-linear dimensionality reduction techniques; 2) sequential forward feature selection based on the generalization accuracy of three machine learning models–decision tree, and linear and nonlinear SVM; and 3) employing multilayer perceptron to predict the VEO based on the selected features of a violent act. Top-ranked selected features were related to the target type and plan and the multilayer perceptron achieved up to 40% test accuracy.

INDEX TERMS Multilayer perceptron, decision tree, SVM, feature selection, visualization.

I. INTRODUCTION

The advance of technology, facility of satellite communications, and ubiquity of the Internet [1] have multiplied the extent and impact of ideology- and politically-motivated acts of violence, have expanded their scope beyond specific locales and regions, and have made them a growing threat against humanity, across the world [2]. With such violence, groups or individuals commit acts of unbelievable brutality against a leader, citizens, an entire city, or nation. The motivation behind it is usually a radicalized interpretation of defending a greater good, politics, or extreme ideology [3]. However, such acts of violence are always disturbing to people's minds, their everyday lives, and destabilizing to societies. Attacks are associated with human and economic tolls, and challenge sustainable development in both modern and developing countries [4]. Groups and individuals committing such violent acts are usually associated with a violent extremist organization (VEO) [5].

The purpose of this study is to investigate the possibility of identifying the responsible VEO based on information available about the violent act. This methodology is useful for assisting in identification of the VEO behind a violent act right after it happens, in cases when no VEO publicly assumes responsibility for a particular act, a VEO assumes responsibility for a particular act but not immediately, or when a particular VEO falsely assumes responsibility for a violent act conducted by another VEO. To the authors' knowledge, the only other attempt to identify the responsible group for extreme acts of violence was performed by Hashemi and Hall [6]. They applied six features, including human casualties and fatalities, level of coordination and expertise, importance of the targeted process, and the extent of its impact on the process. They achieved a 20% cross-validation accuracy using a decision tree. Our study is different in the following aspects: we apply 14 features instead of six applied in their work, we apply a rigorous feature selection process, we apply other machine learning algorithms including a multi-layer perceptron (MLP), and our MLP achieves double the prediction accuracy achieved in their work. The eight additional features applied in our work include: number of hostages, sequential attacks, symbolic nature of the target, uniqueness of the attack method at the time, level of execution, target type, weapon type, and attack type.

Another study related to the theme of this paper is conducted by Tran *et al.* [7]. They classified news articles reporting terrorism events in three southern provinces of Thailand into four classes: demolition attack,

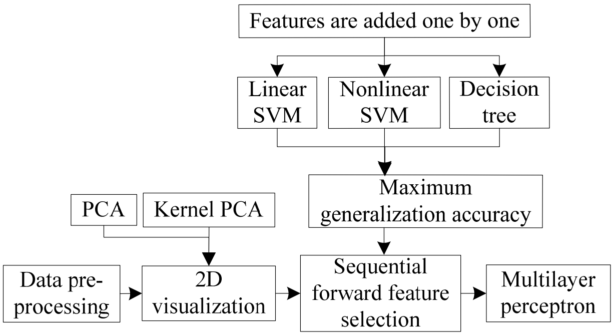

FIGURE 1. The machine learning framework for predicting the responsible VEO for extreme acts of violence.

assassination attack, suicide attack, and unsuccessful attack. They collected news articles from five Thai news websites: Thairath, Dailynew, Naewna, Manager, and Khaosod, from 2007 to 2009. Each news article is represented with a term frequency-inverse document frequency (TF-IDF) feature vector. They applied two classification approaches: fuzzy inference system (FIS) and adaptive neuro-fuzzy inference system (ANFIS). ANFIS resulted in better classification accuracies than FIS. They also classified the same terrorism attacks using the same two methodologies (FIS and ANFIS) but applying a different feature set: the attack's location, the status of victim, and the status of terrorist, instead of TF-IDF feature vectors. This new feature set resulted in better classification accuracies than the TF-IDF feature vectors.

Fig. 1 shows the outline of this paper. Section 2 presents an overview of our data and the application of a linear (PCA) and a nonlinear (kernel PCA) dimensionality reduction technique to visualize it. Section 3 applies the sequential forward method for feature selection based on three classifiers' generalization accuracies. Section 4 employs multilayer perceptron to predict the VEO based on the selected features and discusses the results. Finally, in Section 5 conclusions and future directions are derived. All the accuracies reported in this paper are obtained from ten-fold cross-validation.

II. DATA AND VISUALIZATION

Information about violent acts carried out by VEOs across the world is provided to this study by Radical and Violent Extremism (RAVE) Laboratory at The University of Nebraska Omaha. They developed this dataset by first relying on an open-source database on characteristics of extreme acts of violence, called the global terrorism database (GTD) [8]. Violent acts are included in the GTD if they have a political, social, religious, or economic motive, are intended to coerce, intimidate, or publicize the cause, and/or if they violate international humanitarian law. Among other sources for their dataset are: historical accounts described in open-source data gathered from academic and government sources, scholarly case studies, public-records databases (e.g. Lexis-Nexis), and primary documents from VEOs themselves, such as propaganda and websites. Information was gathered by graduate students with expertise in criminology, industrial and organizational psychology, and information science and technology from a cross functional research center. Coders received 20 hours of training prior to data collection on the nature of VEOs, extremist recruitment, and related manifestations in the context of extremism as well as on search tactics and information filtering.

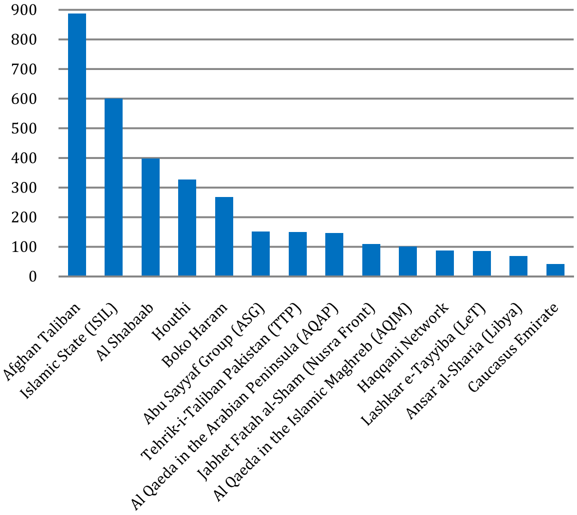

FIGURE 2. Number of violent acts by different VEOs from 1994 until 2016.

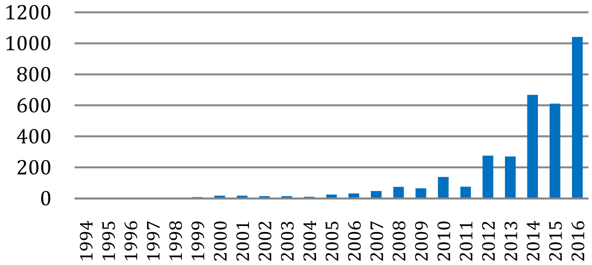

FIGURE 3. Number of violent acts per year from 1994 until 2016.

The dataset contains 3,416 records from the beginning of 1994 until the end of 2016. Fig. 2 shows the number of violent acts carried out by each VEO and the histogram in Fig. 3 shows the number of violent acts per year. As it is indicated, these violent acts have been sharply rising during the past ten years, with the majority of them carried out by well-known VEOs.

There are 17 characteristics associated with each violent act in our dataset, described in Table 1. We refer to them as features in the rest of this paper. The following steps are taken to handle missing values in the dataset:

- Because the physical infrastructural damage is missed for 35% of records, uniqueness of weapon used at the time is missed for 16% of records, and importance of victims to target culture is missed for 16% of records in the dataset, these three features (indicated with a * in Table 1) are removed from the dataset. That leaves us with 14 features.

TABLE 1. Features and their description.

Code name used to refer to this feature	Feature	Type	Description	Percentage of missed values in the dataset
A	Number of casualties	Ratio	Identifying the total estimated casualties.	12%
B	Number of fatalities	Ratio	Identifying the total estimated fatalities.	8%
C	Number of hostages	Ratio	Identifying the total estimated number of hostages taken.	10%
D	Sequential attacks	Ordinal with 3 levels	Assessing the extent to which an attack is part of a larger set of attacks planned and implemented as a collective unit, or is linked to another attack.	0%
E	Level of coordination	Ordinal with 5 levels	Assessing the degree to which the attack method, target, and execution would require a high degree of coordination to implement, either in timing, strategic implementation, or coordinating a number of operatives.	0%
F	Level of expertise	Ordinal with 5 levels	Assessing the degree to which the attack method, target, and execution would require a high degree of expertise to plan, implement, and coordinate, either in terms of weapons development and use, target security, or other facets of the attack.	5%
*Removed from dataset	Physical infrastructural damage	Ordinal with 5 levels	Assessing the degree to which the attack method, target, and execution resulted in significant physical infrastructural damage.	35%
*Removed from dataset	Importance of victims to target culture	Ordinal with 5 levels	Assessing the degree to which the victims targeted or killed in the attack are important to the target culture in terms of psychological or procedural impact.	16%
G	Importance of process affected by attack	Ordinal with 5 levels	Assessing the degree to which the processes effected by the attack (either directly or peripherally) are important to the target culture.	2%
H	Scope of attack's impact on processes	Ordinal with 5 levels	Assessing the degree to which the processes effected by the attack (either directly or peripherally) were impaired as a result of the attack.	2%
I	Symbolic nature of target	Ordinal with 5 levels	Assessing the degree to which the target of the attack is symbolically important to either the target culture, or the organization implementing the attack.	2%
*Removed from dataset	Uniqueness of weapon used at the time	Ordinal with 5 levels	Assessing the degree to which the weapon used in the attack was unique or novel at the time of the attack and within the context of the organization.	16%
J	Uniqueness of attack method at the time	Ordinal with 5 levels	Assessing the degree to which the method used in the attack was unique, novel, or original within the context of the attack and within the typical profile of the organization.	11%
K	Level of execution	Ordinal with 5 levels	Assessing the degree to which the attack was fully completed and effective.	0%
L	Target type	Nominal with 22 levels	Business, Government (General), Police, Military, Abortion Related, Airports & Aircraft, Government (Diplomatic), Educational Institution, Food or Water Supply, Journalists & Media, Maritime, NGO, Other, Private Citizens & Property, Religious Figures/Institutions, Telecommunications, Terrorists/Non-state Militia, Tourists, Transportation, Unknown, Utilities, Violent Political Parties.	0%
M	Weapon type	Nominal with 13 levels	Biological, Chemical, Radiological, Nuclear, Firearms, Explosives/Bombs/Dynamite, Fake Weapons, Incendiary, Melee, Vehicle, Sabotage Equipment, Other, Unknown.	0%
N	Attack type	Nominal with 9 levels	Assassination, Hijacking, Kidnapping, Barricade Incident, Bombing/Explosion, Unknown, Armed Assault, Unarmed Assault, Facility/Infrastructure Attack.	0%

- 52 out of 3,416 records miss at least four features out of 14 features. These records are removed from the dataset.
- The ordinal features' mode is used to replace missing values.
- The average of the second and third quarters is used for missing values in ratio features.
- None of the nominal features have missing values.

Nominal features are coded as dummy vectors [9]. For example, [1,0,0], [0,1,0], [0,0,1] are the dummy vectors for a nominal variable with three levels. All features are normalized to have a zero average and unit variance.

After transforming the training samples using principle component analysis (PCA), we only kept the two principle component scores associated with the largest principle component coefficients (also known as eigenvalues). PCA centers the data and then uses the singular value decomposition (SVD) algorithm [10]. Fig. 4 (left) represents the samples using these two principle component scores which cover 13% of variance in the data. Due to the large number of classes (14), they are not discriminated in this figure.

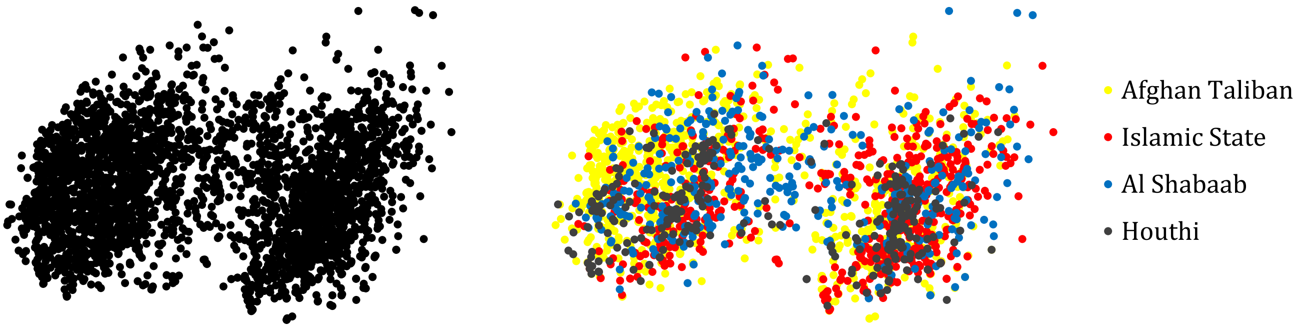

FIGURE 4. Two dimensional visualization using PCA: (left) all samples, (right) only samples from the four largest classes.

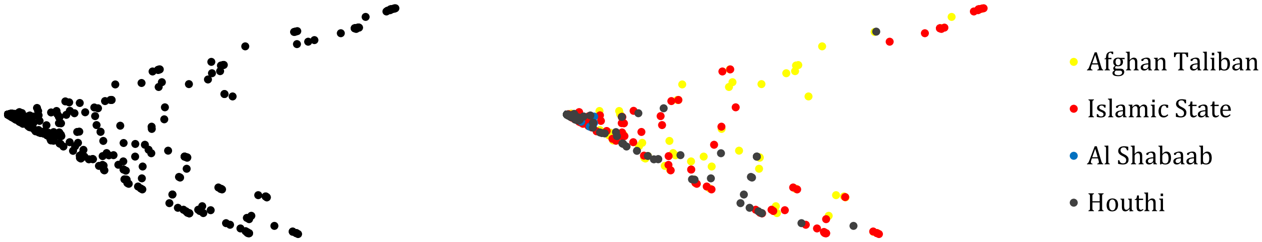

FIGURE 5. Two dimensional visualization using PCA with a Gaussian kernel ($\sigma = 1$): (left) all samples, (right) only samples from the four largest classes.

However, Fig 4 (right) distinguishes among samples from the four largest classes in this two dimensional space. Obviously there are two clusters in this figure but none of the classes fit in any of the clusters. Applying PCA with a Gaussian kernel [11] and visualizing the nonlinearly transformed data using the two principle component scores associated with the largest principle component coefficients, shown in Fig. 5, revealed the same results.

III. SEQUENTIAL FORWARD FEATURE SELECTION

Our criterion for selecting features is the test accuracy of three well-known classifiers with different non-linearities: a linear (linear SVM), a slightly non-linear (non-linear SVM), and a highly non-linear (decision tree) classifier. Table 2 lists the classification accuracy of a linear SVM [12], a nonlinear SVM with a Gaussian kernel, and a decision tree [13] based on individual features. Minimum node size is set to 20 for decision tree, smoothing parameter is set to 0.2 for SVM, and σ is set to 1 for the Gaussian kernel in SVM. These hyperparameters are set by cross-validation inside training data.

For most features, the classification accuracy of the decision tree is close to the classification accuracy of the largest-class-assignment classifier (26.19%), which is the classifier that assigns all the samples to the largest class. This means that either splits in the decision tree hardly result in any reduction in impurity or splits that work well for training data, do not necessarily work as well for the test data. The former is true for the splits at the top levels of the tree and the latter is true for the splits at the bottom levels of the tree. Yet, test samples mostly end up in the largest class. This shows that all that happens in the decision tree is that the majority of test samples land in the most represented class among the training samples and no feature is alone sufficient for proper classification.

Because there is no way that a linear classifier could achieve the accuracy of the largest-class-assignment classifier, if the input features contain considerable randomness with respect to the output class, the accuracy of linear classifiers must be compared with the accuracy of a random classifier (7.14% in our case). The accuracy of linear SVM, for most features in Table 2, is around the accuracy of a random classifier. This shows that any line attempting to separate the classes (in a binary mode), based on one feature, would not perform much better than a random classifier. This indicates the randomness of that feature with respect to the output class. It is noteworthy that SVM hyperplane designs the classifier only based on the closest samples to the hyperplane (support vectors) from the two classes [14]. Statistically, the larger class will have more samples among support vectors than the smaller class but the imbalance among support vectors is much less severe than the imbalance among all samples. The reason is that it is expected that both classes become sparser around the border, assuming that the features are not totally random. This is why SVM is a better choice – when class imbalance is dramatic – in comparison with other linear classifiers that optimize the separating hyperplane based on all samples, e.g. least squares and perceptron.

Nonlinear SVM outperforms linear SVM in 71% of cases in Table 2 and decision tree outperforms nonlinear SVM in all cases. Its interpretation in this application is that the more nonlinear the classifier, the higher the classification accuracy.

TABLE 2. Ten-fold cross validation of the decision tree, SVM, and kernel SVM based on individual features.

Feature	Classifier	Overall accuracy	Max Accuracy
A	SVM with linear kernel	4.64%	26.19%
	SVM with Gaussian kernel	6.36%	
	Decision tree	26.19%	
B	SVM with linear kernel	6.60%	26.72%
	SVM with Gaussian kernel	11.06%	
	Decision tree	26.72%	
C	SVM with linear kernel	1.70%	26.19%
	SVM with Gaussian kernel	1.75%	
	Decision tree	26.19%	
D	SVM with linear kernel	5.83%	26.87%
	SVM with Gaussian kernel	4.85%	
	Decision tree	26.87%	
E	SVM with linear kernel	9.42%	27.44%
	SVM with Gaussian kernel	9.57%	
	Decision tree	27.44%	
F	SVM with linear kernel	9.87%	28.24%
	SVM with Gaussian kernel	6.66%	
	Decision tree	28.24%	
G	SVM with linear kernel	6.66%	26.58%
	SVM with Gaussian kernel	6.21%	
	Decision tree	26.58%	
H	SVM with linear kernel	5.05%	26.40%
	SVM with Gaussian kernel	6.09%	
	Decision tree	26.40%	
I	SVM with linear kernel	5.68%	25.50%
	SVM with Gaussian kernel	10.40%	
	Decision tree	25.50%	
J	SVM with linear kernel	9.54%	26.87%
	SVM with Gaussian kernel	8.32%	
	Decision tree	26.87%	
K	SVM with linear kernel	6.87%	24.94%
	SVM with Gaussian kernel	7.79%	
	Decision tree	24.94%	
L	SVM with linear kernel	8.86%	29.52%
	SVM with Gaussian kernel	12.16%	
	Decision tree	29.52%	
M	SVM with linear kernel	6.33%	27.41%
	SVM with Gaussian kernel	5.65%	
	Decision tree	27.41%	
N	SVM with linear kernel	4.13%	27.62%
	SVM with Gaussian kernel	10.64%	
	Decision tree	27.62%	

This makes it clear that the classes are very nonlinearly distributed in the feature space and a highly nonlinear classifier would be the best choice for this dataset.

An interesting pattern in Table 2 is that the classification accuracies of the three classifiers are fairly correlated. In other words, good features result in a better classification accuracy regardless of the classifier. Another interesting point is that, if the classification accuracy is used as the feature selection criterion, the selected features would highly depend on the choice of the classifier. For example, different features would be selected when SVM's classification's accuracy is used as the selection criterion than when decision tree's classification accuracy is used.

Feature selection is an important step in reducing the demand for large training datasets, mitigating the overfitting problem, boosting the generalization accuracy, and reducing the training time. We chose feature selection over feature generation because we are interested in the nature of the selected features. We now follow the sequential forward feature selection algorithm, as shown below, to select the best features.

Algorithm 1 Sequential Forward Feature Selection

```
$l$: number of features
for $j = 1, \ldots, l$
      for $i = 1, \ldots, l - j$
            temporarily include the $i$-th non-selected
             feature in the feature set
            $c_i = \max\{$Accuracy of decision tree,
             Accuracy of nonlinear SVM,
             Accuracy of SVM$\}$
      $c^j = \underset{i}{max}\{c_i\}$
      if $c^j \leq c^{j-1}$
            return the feature set
      else
            permanently include the $i$-th non-selected
             feature in the feature set
```

The selected features, in the same order as added by the algorithm, along with their classification accuracy are listed in Table 3. The selected features are L, N, F, G, E, M, and I. The relative importance of a feature in the decision tree is defined as sum of the impurity decreases over all internal nodes for which that feature was chosen as the splitting variable [15] and is a measure of how much each feature contributes in training the decision tree. The relative importance of selected features in the decision tree are: 0.2584, 0.1235, 0.1428, 0.1372, 0.02, 0.1214, and 0.1967, respectively. Interestingly, the relative importance of features obtained from the decision tree does not fully correspond with the order in which the features were selected by the sequential forward selection algorithm, despite the decision tree classification accuracy was the main criterion for feature selection in the algorithm. This points to the fact that despite each feature's relative importance in the decision tree is a ballpark reflection of how helpful each feature could be for generalizing the classification, it does not necessarily mirror it. It is noteworthy that classification accuracy is a more legitimate criterion for feature selection than relative importance from decision tree because the former concerns the test data while the latter regards the training data. In general, feature selection methods that involve the test data are more optimal than those involving only the training data.

Beyond the seven selected features, the addition of no other feature would improve the maximum classification accuracy of the three classifiers. For comparison purposes, the last column in Table 3 shows the classification accuracy if all features were used. Decision tree achieves a negligibly better accuracy if only the selected features are used, rather than all features. However, this is not the case for linear and nonlinear SVM.

TABLE 3. Selected features, in the same order as added by the sequential forward selection algorithm, along with their classification accuracy.

	Decision tree	Gaussian kernel SVM	Linear SVM
L	29.52%	12.16%	8.86%
L,N	31.45%	12.13%	5.74%
L,N,F	32.16 %	15.25%	9.66%
L,N,F,G	34.36 %	16.83%	11.77%
L,N,F,G,E	35.85%	16.94%	14.71%
L,N,F,G,E,M	36.06%	19.74%	15.76%
L,N,F,G,E,M,I	36.42%	20.72%	16.91%
All features	36.18%	24.76%	26.43%

TABLE 4. Average test accuracy and its std (obtained from ten-fold cross-validation) for MLPs with six different settings.

Number of hidden layers	Number of nodes in each hidden layer	Average test error	Std
1	25	61.56%	2.05
1	50	61.53%	2.52
2	25	61.50%	2.43
2	50	61.80%	1.64
3	25	62.52%	1.58
3	50	62.39%	2.34

IV. MULTILAYER PERCEPTRON

We employ the power of multilayer perceptron to predict the VEOs based on the selected features. MLPs with six different settings, listed in Table 4, are employed. Batch mode training with backpropagation algorithm is applied to minimize the cost function in Equation 1. The first term in Equation 1 is the quadratic cost function which is the summation of squared errors for each output neuron of each training sample, where $y_j(i)$ and $\hat{y}_j(i)$ are the desired and actual output of the j-th node in the output layer for the i-th training sample, respectively, M represents the number of output neurons and N the number of training samples. The second term, which intends to mitigate the overfitting problem, is the quadratic regularization term which is the summation of squared synaptic weights (w_k), where K represents the total number of synaptic weights in the network.

$$J = \sum_{i=1}^{N} \left\{ \frac{1}{2} \sum_{j=1}^{M} \left(\hat{y}_j\left(i\right) - y_j\left(i\right) \right)^2 \right\} + 0.05 \sum_{i=1}^{K} w_k^2 \qquad (1)$$

The model is set not to update the synaptic weights after those epochs resulting in an increase in the total cost. Adaptive learning is applied where a separate learning rate is used for each synaptic weight in the network. A synaptic weight's learning rate is multiplied by 1.2 if the partial derivative of the loss, with respect to that synaptic weight, keeps the same sign (positive or negative) in two successive epochs and multiplied by 0.7 otherwise [16]. Besides, all learning rates are multiplied by 1.1 or 0.8 after each epoch based on whether the total cost decreases or increases. Each MLP is trained for $h \times 100$ epochs, where h is the total number of hidden nodes in the MLP. Table IV shows the average test error and its standard deviation (obtained from ten-fold cross-validation) for each MLP.

According to this table, all test errors are within one standard deviation of each other. Thus, we choose the simplest MLP, the one with one hidden layer with 25 nodes, for further investigations. Fig. 6 (top chart) shows how the training and test errors evolve over training epochs, when only the selected features are applied. Three following charts in Fig. 6 indicate similar curves, for comparison purposes: (second chart) when all features are used as input, (third chart) when the first 43 principle component scores of PCA are used as input, and (fourth chart) when the first 43 principle component scores of Gaussian kernel PCA are used as input. The number 43 is chosen because the number of columns in the features matrix, corresponding to the seven selected features, is also 43. Some of the features are coded as dummy vectors and occupy more than one column. The charts in Fig. 6 report the average and standard deviation of training and test errors at each epoch. The ten-fold cross-validation has been performed at each epoch. Thus, the training and test errors at each epoch, reported in Fig. 6, are the average over the ten folds. The standard deviation of errors is also calculated over the ten folds at each epoch.

When only selected features are used, the training error falls down to 49% after 2,500 epochs, but when all features are used, the training error falls down to 41%. This large difference in training error is explained by overfitting a complex model to a larger number of features during training. On the other hand, when only selected features are used, the minimum test error is 61%, but when all features are used, the minimum test error is 57%. The difference between the test errors is half the difference between the training errors. This shows that the feature selection process was effective in: (a) picking features that improve the prediction model's generalizability to unseen data, (b) reducing the overfitting problem on the training data, and (c) reducing the processing burden by lessening the number of features.

The MLP with generated features from PCA performs overall similar to the MLP with all features. The reason is that the first 43 principle component scores of PCA capture 97.83% of the variance in the original feature space. This is almost equivalent to using all the original features. Feature generation methods are generally more efficient than feature selection methods in reducing the number of features without losing the variance in the original feature space, but generated features are artificial and hard to interpret [9]. This is the payoff for dismissing principle component scores with close to zero variance in the PCA transformed space.

While the MLP with generated features from PCA performs best, the MLP with generated features from kernel PCA performs worst. The MLP with kernel PCA generated features saturates after around 700 epochs and never gets any better than the largest-class-assignment classifier (with 74% error rate). The main reason is that while PCA transforms the data to a space with equal dimensionality, kernel PCA transforms the data to a space with a much larger dimensionality, equal to the number of samples (3,364 in our case). Because of this much larger dimensionality, the first 43 principle

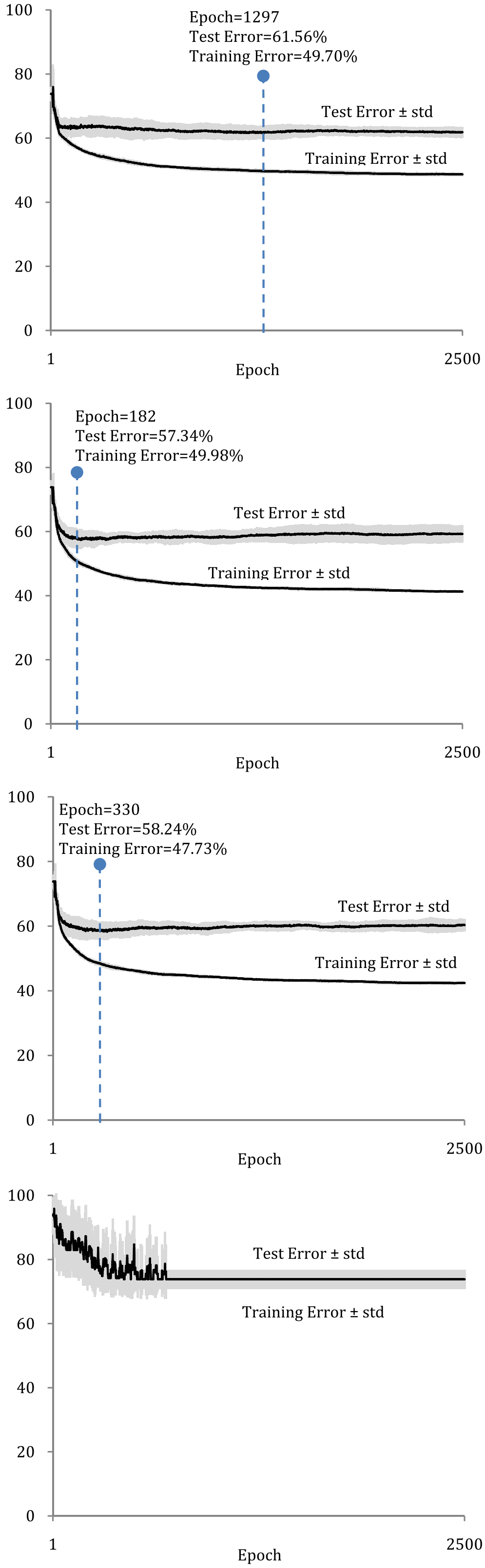

FIGURE 6. Training and test error with one standard deviation of uncertainty for different epochs of MLP using; (top) selected features, equivalent to 43 columns in the feature matrix, (second) all features, (third) the first 43 principle component scores of PCA, (bottom) the first 43 principle component scores of Gaussian kernel PCA.

component scores of kernel PCA capture only 13.87% of the variance in the new feature space. Despite unfavorable performance of kernel PCA in this specific application, there are situations where kernel PCA is the preferred method of feature generation and dimensionality reduction, e.g. when classes form concentric hyper-spheres.

It is noteworthy that sequential forward feature selection is supervised while PCA, either linear or nonlinear, is unsupervised. In other words, we selected features that maximize the generalization accuracy to unseen data but PCA generates features that align with the maximum variance in the original feature space. Keeping features with the largest variance is a heuristic method, though a feature that maximizes the variance does not necessarily separates the classes as well and might not optimize the generalization accuracy. The last attempt on identifying the responsible group for extreme acts of violence was performed by Hashemi and Hall [6]. They applied six features, including human casualties and fatalities, level of coordination and expertise, importance of the targeted process, and the extent of its impact on the process. They achieved a 20% cross-validation accuracy using a decision tree, almost half the accuracy achieved by our MLPs.

V. CONCLUSIONS AND FUTURE DIRECTIONS

This study was the first attempt to predict the responsible VEO for acts of violence based on its human and structural tolls, its target, intelligence, and weapons utilized in the attack. Both linear and non-linear transformation of samples into a two dimensional space clearly indicated two clusters. However, all 14 classes had samples in both clusters and no class was even closely contained in one cluster. Selected features in order of their significance in generalizing the model to unseen data to predict the responsible VEO for an act of violence are: target type, attack type, level of expertise, importance of the process affected by the attack, level of coordination, weapon type, and symbolic nature of the target. Unselected features include: number of casualties, number of fatalities, number of hostages, level of execution, scope of attack's impact on processes, sequential attacks, and uniqueness of attack method at the time. Looking at these two feature sets tells us what aspects of an attack are well-planned by VEOs and what aspects are more of random consequences. For example, the choice of target is the most prominent distinguishing factor among VEOs. In other words, VEOs have more control over choosing their targets, weapons, and how to carry out the attack which makes these factors significant in identifying the responsible VEO. On the other hand, the structural and human tolls and how well the plan was executed are not much dependent on the responsible VEO which could be interpreted as VEOs having less control over these factors.

The generalization accuracy of an MLP for predicting the responsible VEO based on the characteristics of the violent act reached 39%, when only selected features are applied, 43% when all features are applied, and 42% when PCA generated features are applied. This is a considerable improvement over the accuracy of a random classifier (7.14%), a reasonable improvement over the accuracy of the largest-class-assignment classifier (26.19%), linear SVM (16.91% for selected features and 26.43% for all features), kernel SVM (20.72% for selected features and 24.76% for all features), and a slight improvement over the accuracy of the decision tree (36.42% for selected features and 36.18% for all features).

In the dataset used in this research only one VEO is responsible for each violent act. Thus, the assumption is made that VEOs work independently. The prediction accuracy will be adversely affected if an act of violence is carried out as the result of coalition and intelligence sharing among two or more VEOs with similar agendas and motivations. In our future work, we intend to expand the width of our dataset and investigate other features, including location and time. These features are mostly unknown at the time. Additionally, the expansion of our dataset in length is the key to application of deeper MLPs for predicting VEOs.

REFERENCES

[1] M. Hashemi and A. Sadeghi-Niaraki, "A theoretical framework for ubiquitous computing," *Int. J. Adv. Pervas. Ubiquitous Comput.*, vol. 8, no. 2, pp. 1–15, 2016.

[2] R. W. Taylor, E. J. Fritsch, and J. Liederbach, *Digital Crime and Digital Terrorism*. Upper Saddle River, NJ, USA: Prentice-Hall, 2014.

[3] G. Martin, *Understanding Terrorism: Challenges, Perspectives, and Issues*. Newbury Park, CA, USA: SAGE, 2017.

[4] M. B. Altier, C. N. Thoroughgood, and J. G. Horgan, "Turning away from terrorism: Lessons from psychology, sociology, and criminology," *J. Peace Res.*, vol. 51, no. 5, pp. 647–661, 2014.

[5] G. S. Ligon, M. K. Logan, M. Hall, D. C. Derrick, J. Fuller, and S. Church, "The Jihadi industry: Assessing the organizational, leadership, and cyber profiles," START, College Park, MD, Tech. Rep., 2017.

[6] M. Hashemi and M. Hall, "Identifying the responsible group for extreme acts of violence through pattern recognition," in *Proc. Int. Conf. HCI Bus., Government, Organizations*, Cham, Switzerland, 2018, pp. 594–605.

[7] D. Tran, C. Haruechaiyasak, P. Meesad, and U. Inyaem, "Terrorism event classification using fuzzy inference systems," *Int. J. Comput. Sci. Inf. Secur.*, vol. 7, no. 3, pp. 247–256, 2010.

[8] G. LaFree, L. Dugan, and E. Miller. *Global Terrorism Database*. Accessed: 2018. https://www.start.umd.edu/gtd

[9] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. New York, NY, USA: Springer, 2009.

[10] G. H. Golub and C. F. Van Loan, *Matrix Computations*. Baltimore, MD, USA: Johns Hopkins, 1985.

[11] B. Schölkopf, A. Smola, and K.-R. Müller, "Kernel principal component analysis," in *Proc. Int. Conf. Artif. Neural Netw.*, Berlin, Germany, 1997, pp. 583–588.

[12] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[13] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and Regression Trees*, Boca Raton, FL, USA: CRC Press, 1984.

[14] M. Hashemi and H. A. Karimi, "Weighted machine learning," *Statist., Optim. Inf. Comput.*, vol. 6, no. 4, pp. 497–525, 2018, doi: 10.19139/soic.v6i4.479.

[15] L. Breiman and R. Ihaka, "Nonlinear discriminant analysis via scaling and ACE," Dept. Statist., Univ. California, Berkeley, CA, USA, Tech. Rep. 40, 1984.

[16] R. A. Jacobs, "Increased rates of convergence through learning rate adaptation," *Neural Netw.*, vol. 1, no. 4, pp. 295–307, 1988.

MAHDI HASHEMI received the Ph.D. degree in computing and information from the University of Pittsburgh in 2017 and a Post-Doctoral Fellowship in information science and technology from the University of Nebraska in 2018. He is the primary author of 4 conference papers and 20 journal articles appearing in a multi-disciplinary range of conferences and journals, including the IEEE Transactions on Intelligent Transportation Systems, IEEE Access, *Statistics, Optimization & Information Computing*, *Computers & Geosciences*, *Computers, Environment & Urban Systems*, the *Journal of Intelligent Transportation Systems*, *Transactions in GIS*, the *Journal of Computing in Civil Engineering*, and *Fire Technology*. His research interests include machine learning, deep learning, computing with spatial–temporal data, and intelligent transportation.

MARGERET HALL received the Ph.D. degree from the School of Economics and Industrial Engineering, Karlsruhe Institute of Technology, Karlsruhe, Germany, in 2015, where she was a Research Assistant and a Research Group Leader from 2012 to 2015. She was a Research Assistant with the United Nations Office at Geneva, Geneva, and UNHCR in 2011, a Visiting Researcher with the Wharton School, University of Pennsylvania, in 2013, and a Guest Lecturer with the Frankfurt School of Finance and Management in 2015. She is currently an Assistant Professor with the School of Interdisciplinary Informatics, University of Nebraska Omaha, Omaha, where she received a position of excellence in violent, extremist organizations. She is also the Director of Applied Innovations II Lab, focusing on data mining, social computing, text analytics, sentiment analysis, and policy informatics.