Department of Mathematics and Computer Science

SDU

Towards Semi-Supervised Subspace Learning for Outlier Detection in Big Data

Muhammad Rajabinasab, Anton D. Lautrup, Peter Schneider-Kamp, and Arthur Zimek

Background: Outlier detection in big data is vital for applications [1]. It is concerned with different challenges, such as the curse of dimensionality and massive amounts of data, which raise the need for scalable and efficient algorithms.

Many attempts have been made in order to address the challenges of outlier detection in big data. Several methods are based on approximate nearest neighbor (ANN) search methods [2]. Distributed approaches are also used in order to conduct outlier detection in large-scale data by leveraging multiple computing nodes to process large-scale, high-dimensional datasets in parallel.

Proposed Method:

- ❑ We propose a semi-supervised subspace learning method to learn a low-dimensional subspace for outlier detection.
- ❑ We construct a novel linear Deep Neural Network (DNN), namely Hybrid Deep Support Vector Data Description (HDSVDD). HDSVDD is trained only on inliers to learn a mapping to a lower-dimensional representation of the data points.
- ❑ Inspired From Deep SVDD [3], The training procedure involves training connected linear DNNs specialized in mapping the inliers into lower-dimensional compact hyperspheres. The distance of the lower-dimensional representation of the data point to the center of the hyperspheres constructs the coordinates of the data points in the lower-dimensional subspace. The one-class training procedure of the neural network learns the representation of the inliers. Hence, when dealing the unseen data points (e.g., test set), it is expected to map the inliers closer to the center of the hyperspheres and the outliers further apart. This process allows the discrimination between inliers and outliers in the lower-dimensional subspace.

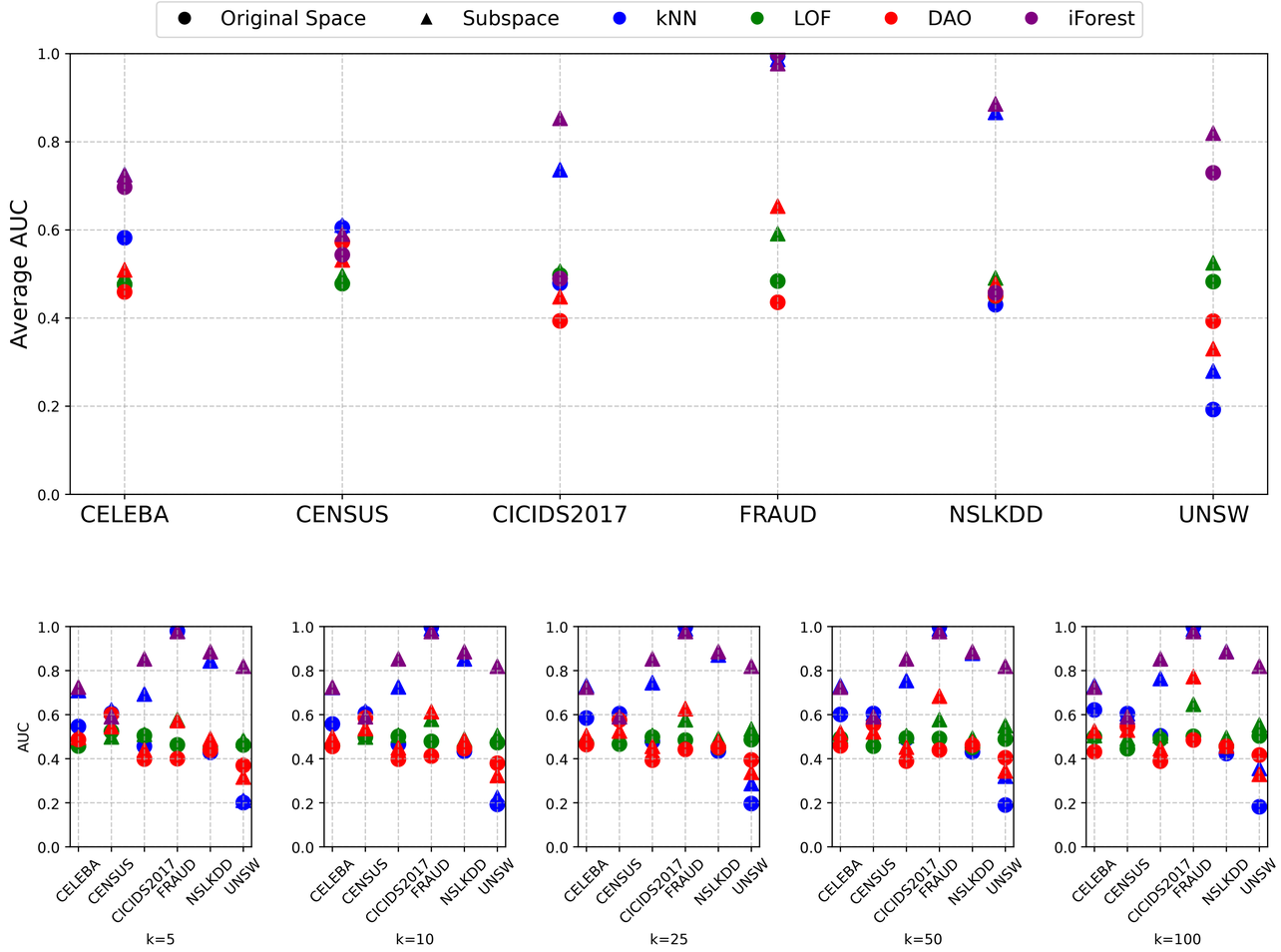

Figure 1 – The comparison of the different outlier detection methods on the original feature space and the subspace learned by HDSVDD. The experiments are done using 5-fold cross-validation. The top plot shows the average results over all values of k. The results for specific values of k are presented in the subplots below.

Algorithm 1 Subspace Learning Using HDSVDD

Input: Dataset D (only inliers), subspace dimensionality d'
Output: Trained neural networks for subspace representation $\{\mathrm{NN}_1, \mathrm{NN}_2, \ldots, \mathrm{NN}_{d'}\}$

Initialize d' autoencoders $\{\mathrm{AE}_1, \mathrm{AE}_2, \ldots, \mathrm{AE}_{d'}\}$, each with a different number of layers.
Set pre-training epochs $E_{\text{pre-train}}$ and joint training epochs E_{joint}.
Ensure a layer with m units in the encoder part for all autoencoders.
for $i = 1$ to d' **do**
    Train AE_i on D for $E_{\text{pre-train}}$ epochs.
end for
Select a subset of encoders' architectures from all autoencoders up to the point it reaches the layer with m units.
for each NN_j in $\{\mathrm{NN}_1, \mathrm{NN}_2, \ldots, \mathrm{NN}_{d'}\}$ **do**
    Project D to output space using NN_j to obtain $Z_j = \{z_{j1}, z_{j2}, \ldots, z_{jn}\}$. ▷ $z_{ji} \in \mathbb{R}^m$
    Compute center $c_j = \frac{1}{n} \sum_{i=1}^{n} z_{ji}$.
end for
for $e = 1$ to E_{joint} **do**
    Compute loss $L = \sum_{j=1}^{d'} \sum_{x_i \in D} \| \mathrm{NN}_j(x_i) - c_j \|_2^2$.
    Update parameters of $\{\mathrm{NN}_1, \mathrm{NN}_2, \ldots, \mathrm{NN}_{d'}\}$ to minimize L.
end for
return Trained Neural networks $\{\mathrm{NN}_1, \mathrm{NN}_2, \ldots, \mathrm{NN}_{d'}\}$

Algorithm 2 Projecting Data into the Subspace Using HDSVDD

Input: Dataset D, trained neural networks $\{\mathrm{NN}_1, \mathrm{NN}_2, \ldots, \mathrm{NN}_{d'}\}$
Output: The coordinates of the data in the subspace D_s

for each data point $x_i \in D$ **do**
    Compute coordinates $[d_{i1}, d_{i2}, \ldots, d_{id'}]$, where $d_{ij} = \| \mathrm{NN}_j(x_i) - c_j \|_2$.
    Assign $D_s = [d_{i1}, d_{i2}, \ldots, d_{id'}]$ as the subspace coordinates of x_i.
end for
return Subspace coordinates D_s.

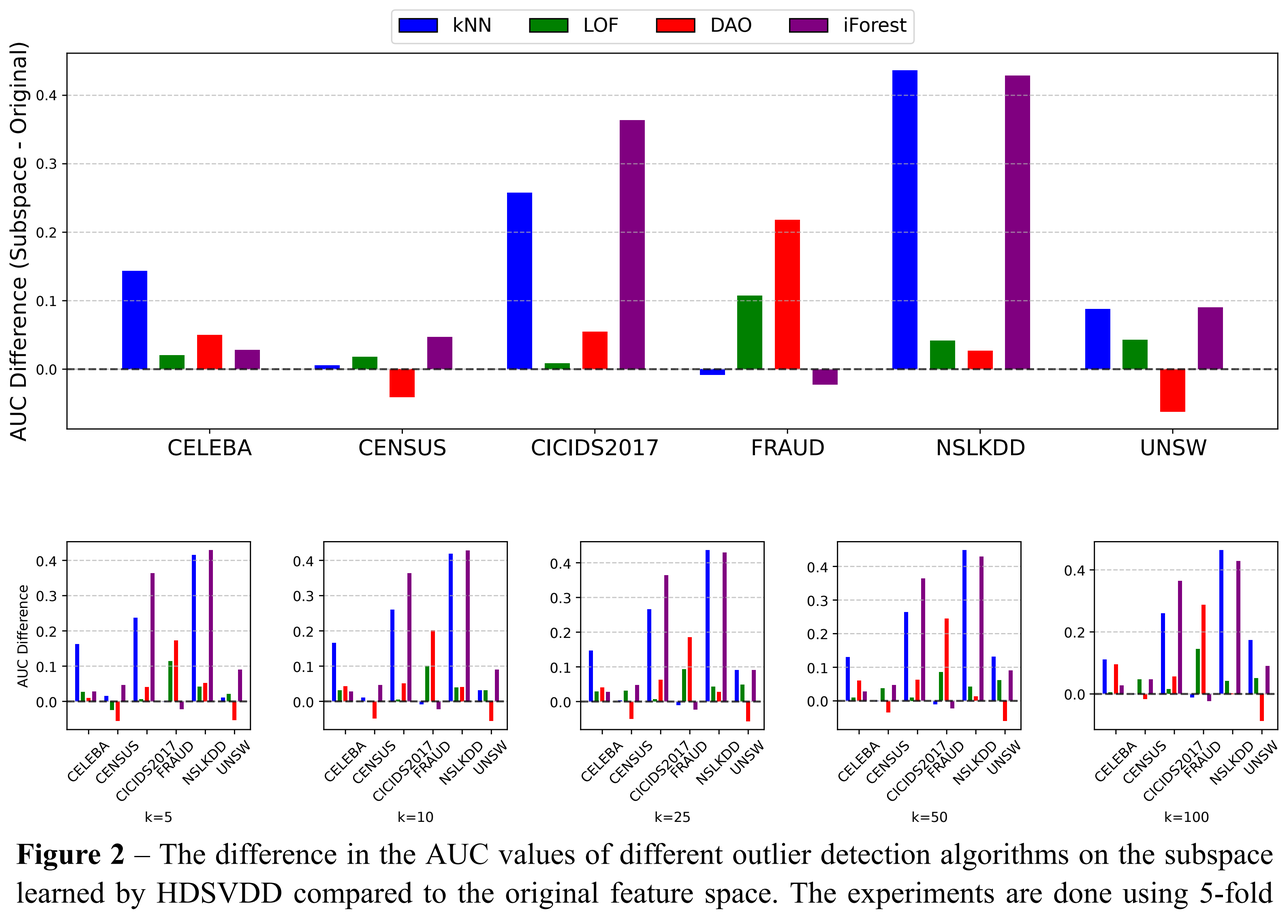

Figure 2 – The difference in the AUC values of different outlier detection algorithms on the subspace learned by HDSVDD compared to the original feature space. The experiments are done using 5-fold cross-validation. The top plot shows the average results over all values of k. The results for specific values of k are presented in the subplots below.

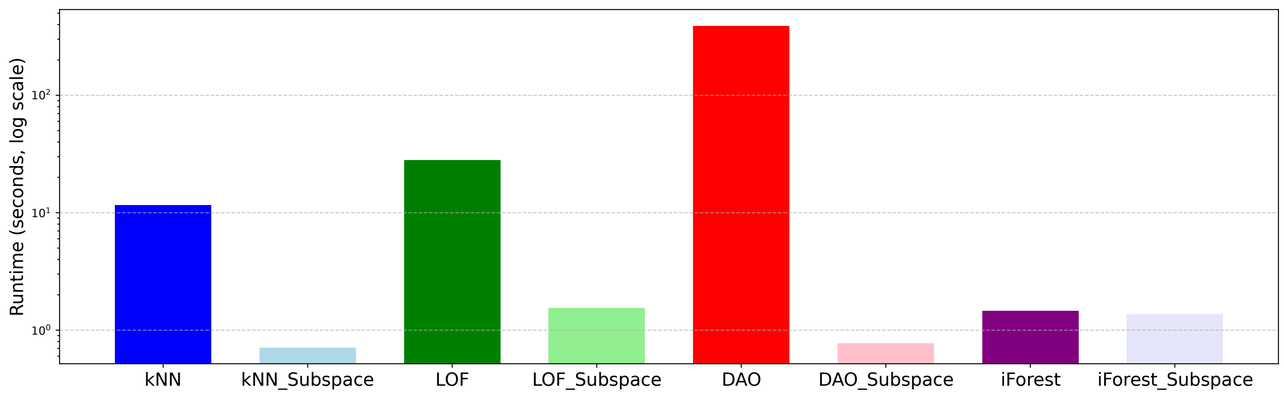

Figure 3 – Runtime comparison on the original feature space and the subspace for different methods on CELEBA dataset. The y-axis is represented on a logarithmic scale.

References

[1] Li, W., Zhang, H., Wang, L.: Anomaly detection in industrial iot with machine learning: A survey. IEEE Internet Things J. 9(15), 12345–12360 (2022).
[2] Okkels, C.B., Aumüller, M., Zimek, A.: On the design of scalable outlier detection methods using approximate nearest neighbor graphs. In: SISAP 2024. pp. 170–184 (2024).
[3] Ruff, L., Vandermeulen, R., Goernitz, N., Deecke, L., Siddiqui, S.A., Binder, A.,Müller, E., Kloft, M.: Deep one-class classification. In: Int. Conf. Mach. Learn. pp.4393–4402. (2018).

sdu.dk
rajabinasab@imada.sdu.dk, University of Southern Denmark