Enhancing IoT Security: Anomaly Detection using Deep Support Vector Data Description and Contractive Autoencoder

THE UNIVERSITY of NEW ORLEANS

Sharmin Aktar, PhD Candidate

Department of Computer Science, University of New Orleans, New Orleans, LA, USA

Introduction & Motivation

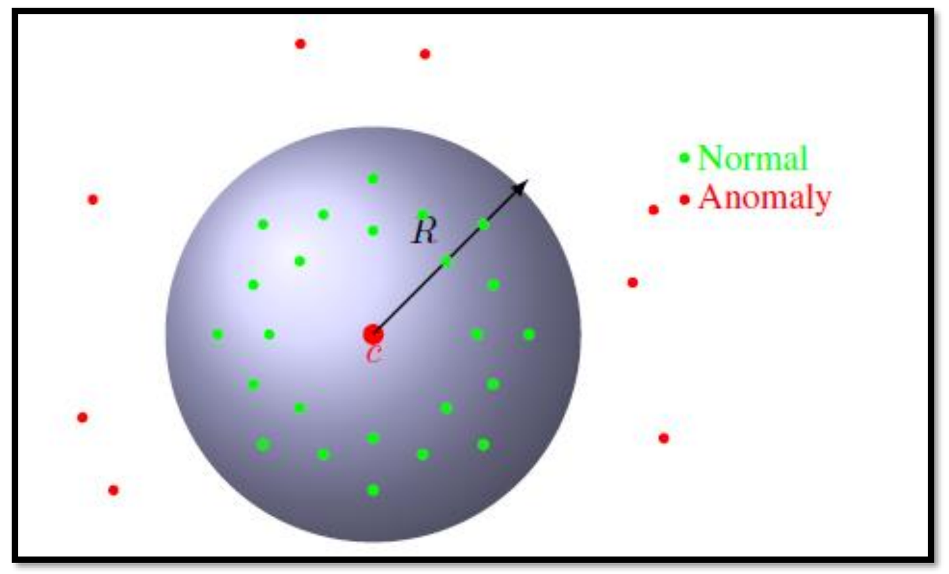
- ❑ **IoT security challenge:** Detecting diverse, heterogeneous attacks, both known and novel
- ❑ Current models struggle with handling heterogeneity and lack dedicated anomaly detection objectives
- ❑ **Our solution:** Integrating DSVDD with Contractive Autoencoders (CAE), with an objective function based on anomaly detection
- ❑ Outperforms traditional AI in subtle anomaly detection

Methodology

- ❑ Combined Deep Support Vector Data Description (DSVDD) with Contractive Autoencoders (CAE)
- ❑ CAE learns robust, low-dimensional features from normal data
- ❑ DSVDD identifies anomalies outside a minimal hypersphere in feature space[1]
- ❑ Semi-supervised learning using only normal data for training
- ❑ Threshold optimized using F-score on validation set
- ❑ Objective Function:

$$J_{DSVDD-CAE}(\theta) = J_{CAE}(\theta) + \alpha J_{DSVDD}(\theta)$$

Methodology



Visualization of SVDD hypersphere with normal and anomalous data points.

Algorithm 1 Anomaly Detection using DSVDD-CAE

```
function PREDICT(xtrain, xtest, xval, yval)
    score-normal ← ComputeAnomalyScore(xtrain)
    threshold ← GetOptimalThreshold(xval, yval, score-normal)
    anomaly score ← ComputeAnomalyScore(xtest)
    is anomaly ← anomaly score ≥ threshold
    return is anomaly
function GETOPTIMALTHRESHOLD(xval, yval, score-normal)
    Define a range of potential_thresholds from min(score-normal) to max(score-normal)
    best_f1 ← 0
    best_threshold ← potential_thresholds[0]
    for each threshold in potential_thresholds do
        Predict anomalies for xval based on current threshold to get predicted_yval
        Compute F1 score using actual yval and predicted_yval
        if current F1 score > best_f1 then
            Update best_f1 and best_threshold with current F1 score and threshold
    return best_threshold
function COMPUTEANOMALYSCORE(x)
    z ← EncodeInput(x)
    distance ← ComputeDistance(z)
    x recon ← DecodeInput(z)
    recon error ← ComputeReconstructionError(x recon, x)
    anomaly score ← CombineScores(distance, recon error)
    return anomaly score
```

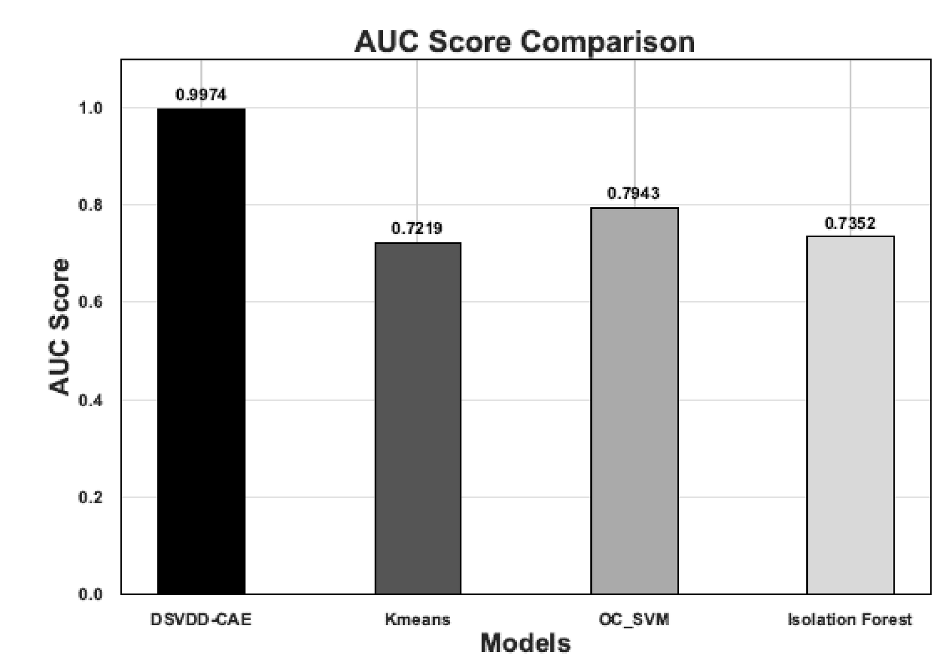
Results

Method	Precision	Recall	F1-score	Accuracy
KMeans	66.13%	66.16%	66.15%	87.81%
OCSVM	69.40%	69.43%	69.41%	88.98%
Isolation Forest	70.51%	68.95%	69.68%	89.50%
Proposed Method	**98.25%**	**99.80%**	**99.01%**	**99.64%**

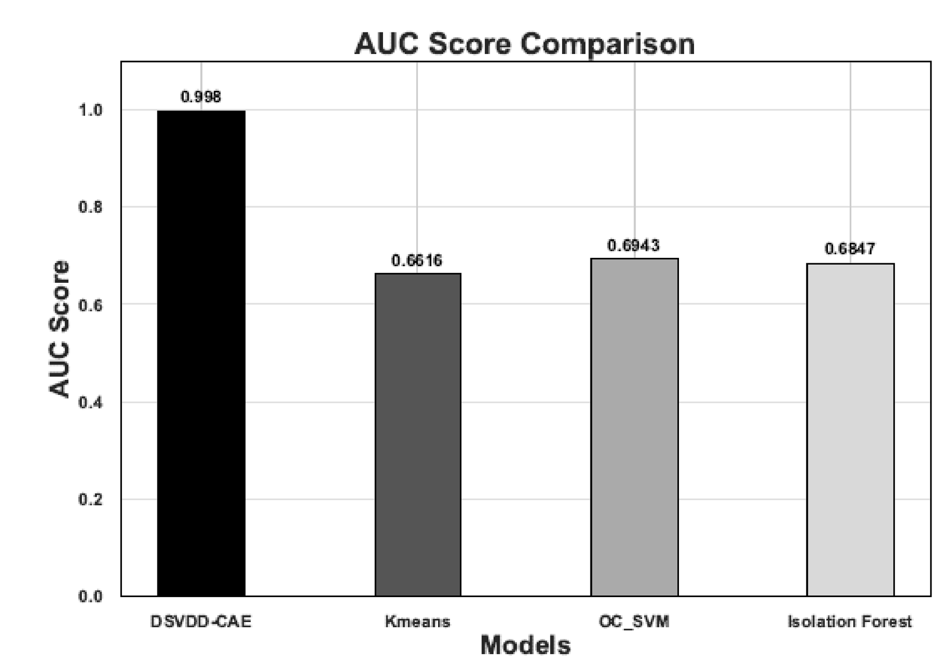
IoTID20 dataset

Method	Precision	Recall	F1-score	Accuracy
KMeans	84.77%	72.19%	76.28%	88.92%
OCSVM	91.34%	76.38%	81.35%	91.29%
Isolation Forest	86.86%	83.52%	77.89%	89.68%
Proposed Method	**98.77%**	**99.74%**	**99.25%**	**99.57%**

ToN-IoT dataset

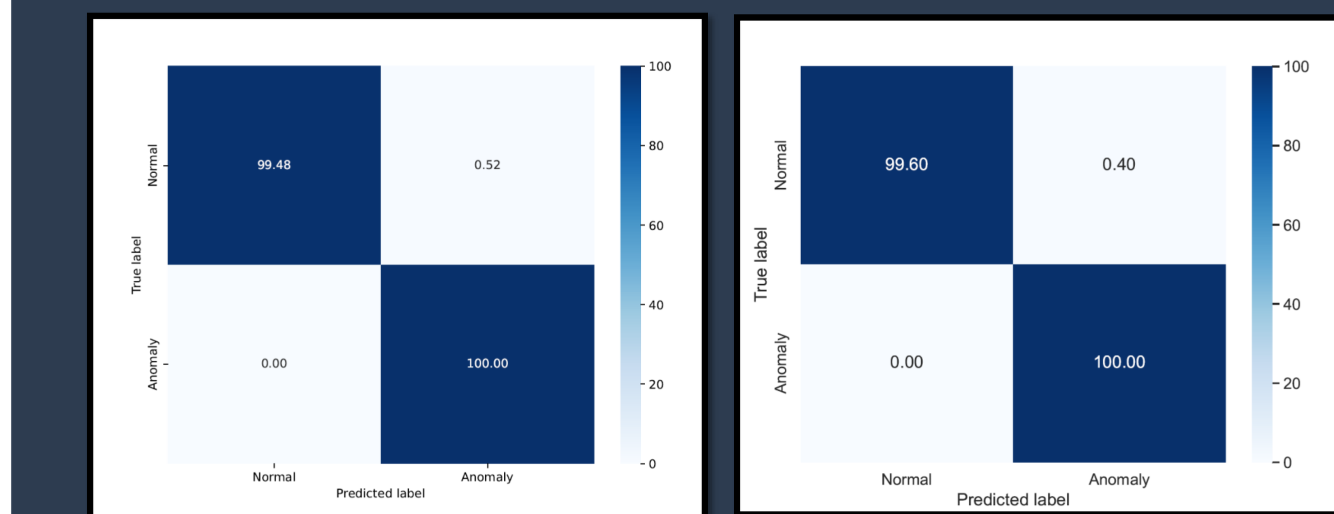


(a) AUC Score Comparison (TON-IoT dataset)



(b) AUC Score Comparison (IoTID20 Dataset)

Results



Confusion Matrices on ToN-IoT and IoTID20 Datasets

Conclusion

- ❑ **Proposed DSVDD-CAE model** for anomaly detection in IoT networks
- ❑ Achieved a **99.25%** F1-score on the ToN-IoT[2] dataset and a **99.01%** F1-score on the IoTID20[3] dataset, outperforming traditional methods
- ❑ **Future work:** Expand testing on additional IoT datasets to further assess generalization and model scalability

References

[1] Tax, D.M., Duin, R.P. Support Vector Data Description. Machine Learning 54, 45–66 (2004). https://doi.org/10.1023/B:MACH.0000008084.60811.49

[2] N. Moustafa. (2020). TON-IoT Dataset. [Online]. Available:https://cloudstor.aarnet.edu.au/plus/s/ds5zW91vdgjEj9i

[3] Ullah, Imtiaz and Mahmoud, Qusay H., A scheme for generating a dataset for anomalous activity detection in iot networks, Canadian Conference on Artificial Intelligence, Springer International Publishing, 2020.

Access the Full Paper