International Journal of Advance and Innovative Research
Volume 12, Issue 2 (XVII): April - June 2025
ISSN 2394 - 7780

ANOMALY DETECTION IN NETWORK TRAFFIC USING A HYBRID MODEL OF K-MEANS CLUSTERING AND AUTOENCODERS

[1]Apurva Kishor Gawde and [2]Sunita Jena
[1]Student and [2]Assistant Professor, MSc. Big Data Analytics Department, Jai Hind College

ABSTRACT
Anomaly detection is generally understood to refer to rare objects, events or observations that deviate significantly from the bulk of the data and do not conform to well-defined notions of normal practice. The model uses the k-means power of clustering to identify potential anomalies based on the distance between cluster centroids and optimizes the search by using an autoencoder to detect reconstruction errors and then, hybrid method combining kmeans and autoencoder, compared with individual methods . The model was tested on real-time network traffic data collected through Wireshark, which demonstrated its effectiveness in detecting anomalous networks. KMeans and Autoencoders have been shown to perform well in detecting anomalies in network traffic, where both methods show the same number of anomalies and in addition, the hybrid method detects fewer anomalies and outperforms both KMeans and Autoencoders.

Keywords: *Anomaly detection, K-Means, cyber security, autoencoders.*

1. INTRODUCTION
With the rapid expansion of digital communications and the reliance on network management systems, the detection of anomalous behavior in network traffic has become increasingly important to maintain security and system integrity New vulnerabilities emerge daily and quickly used in daily attacks. Abnormalities in network traffic are often indicative of security threats such as malware, denial of service (DoS) attacks, or data breaches. Anomaly detection is generally understood to be identification of data that deviates from the normal behaviour. Systems capable of anomaly detection are important tools in many fields. Anomaly detection is a critical task in identifying outliers or deviations from expected behavior in data. [8] They are used to detect financial fraud, network intrusions, unusual traffic situations and other rare events.

These rare events may have an impact on a specific system, but they are hard to find. [8]. Traditional anomaly detection methods may struggle to cope with today's network data volume, speed, and complexity. Cybersecurity attacks are hard to detect. These attacks can result from malicious or benign behavior, internally or externally through malware, targeted attacks, or APTs. However, internal threats are much more powerful and potentially more damaging than external threats because they have already penetrated the network. These activities pose unknown threats and can steal, damage, or alter assets or operations.

Therefore, it is a major concern for industry to enable anomaly detection for cyber network security. Abnormal behavior detection provides real-time cyber-attack threat detection. It monitors unethical user behavior protecting companies from threats.

In this study, we propose a hybrid anomaly detection model that exploits autoencoders and KMeans clustering capabilities. The hybrid model aims to diagnose anomalies in automated traffic, combining the ability of automatons to identify data patterns with the clustering capabilities of KMeans to organize similar traffic patterns. The proposed model improves the detection rate of network anomalies by utilizing both reconstruction error from autoencoders and clustering distance from KMeans.

2. RELATED WORK
Anomaly detection in network traffic has been extensively studied, various methods have been developed to deal with complex and diverse network data Traditional methods, such as statistical illegal methods, are limited by sophisticated or new discoveries.[1] Machine learning models such as support vector machines and random forest have improved detection accuracy, although these methods may lack robustness in detecting complex, nonlinear anomalies [2] especially, K-means are collected to detect anomalies by clustering common behavior and identifying outliers s are used However, they are often ineffective in capturing complex network behaviors in their own right [3]. The k-Means clustering method is first used to partition the training instances into k clusters using Euclidean distance similarity. [7]

Meanwhile, deep learning models, especially autoencoders, have shown promise at the learning level of normal network traffic, and enable deviations to be flagged efficiently [4]. Hybrid approaches combining clustering and autoencoders have emerged to address the weaknesses of individual models. For example, a study has shown that pre-clustering data with K-Means before training the autoencoder can increase anomaly detection accuracy, and reduce false positives [5]. However, challenges remain, including the scalability of this model and

ISSN 2394 - 7780

optimization of threshold values to balance detection accuracy and false positive rates [6]. These works lay the foundation for exploring hybrid models that harness the power of clustering and deep learning for improved anomaly detection in network traffic. Clustering techniques are widely used in anomaly detection to identify patterns in data and separate normal from abnormal behavior. [9] Machine learning techniques enable the development of anomaly detection algorithms that are non-parametric, adaptive to changes in the characteristics of normal behaviour in the relevant network, and portable across applications. [10]

3. ANOMALY DETECTION WITH KMEANS AND AUTOENCODERS

In this section, we briefly discuss the k-Means clustering and the autoencoders methods that are used for anomaly detection.

3.1Anomaly Detection with k-Means Clustering :

This structure mirrors our concept, detailing the clustering and anomaly detection processes, including the distance calculations and threshold-based anomaly classification.

1) Set K : Choose the number of clusters, K, based on the analysis requirements.

2) Initialize the Centroids: Select K initial centroids C = {c1, c2, ..., cK}, either randomly or with a method like k-means++.

3) Assign Points to Clusters: For each point xi, calculate the distance to each centroid ck using:

$$d(x_i, c_k) = \sqrt{\sum_{j=1}^{m} (x_{ij} - c_{kj})^2}$$

$$c_k = \frac{1}{|S_k|} \sum_{x_i \in S_k} x_i$$

4) Update Centroids: For each cluster k, recalculate the centroid as the mean of points in the cluster:

3.2Anomaly Detection with Autoencoders :

This structure depicts an unsupervised anomaly detection pipeline using an autoencoder that learns to reconstruct normal data and flags significant reconstruction deviations as anomalies.

1) Build an autoencoder model with input and output layers equal to the feature size and smaller hidden layers in between.

2) Train the autoencoders on normal data to minimize reconstruction error.

3) For each test point Z:

- Pass Z through the autoencoder to get the reconstructed output.

- Calculate the reconstruction error (e.g., mean squared error between Z and its reconstruction.)

4) Set an anomaly threshold based on reconstruction errors. (e.g., 95th percentile).

5) Id the reconstruction error for Z exceeds the threshold, classify Z as an anomaly, otherwise, classify as normal.

ISSN 2394 - 7780

4. PROPOSED SCHEME

In this section, we proposed a hybrid model of Kmeans clustering and Autoencoders. In this hybrid anomaly detection model, we combine the power of k-Means clustering with autoencoders to improve anomaly detection accuracy. Here's how each step works.

1. Separate training of k-Means and Autoencoder:

- We first train a k-Means clustering model for cluster normal data into clusters, each with a centroid. This chart helps identify points that are far from the norm, and can indicate abnormalities.
- We also train an autoencoder, a neural network designed to encode and then reconstruct input data. Because it is trained on normal data, the autoencoder learns to reconstruct these data points correctly, resulting in less reconstruction error for the normal data.

2. To examine each case with two examples:

- For the new data point Z , we use the two models separately:
- k-Means: Calculate the distance between Z and the nearest cluster center. If Z is far from any focal point, a discrepancy may occur.
- Autoencoder: We feed Z through an autoencoder to get the reconstructed version of Z

. The difference between Z and its return (reconstruction error) helps us to measure how constant the autocoder perceives Z to be. High errors indicate inconsistency.

3. Setting thresholds for similarity:

- Define a threshold for k-Means and autoencoder based on the highest percentage of normal data distances and reconstruction errors (e.g. 95%) This means that we expect to expect the top 5% (or only the other percentage) of values will exceed this threshold under appropriate circumstances.

4. Hybrid Anomaly Detection:

- If both its distance to the nearest cluster center point (k-Means) and its reconstruction error (autoencoder) exceed their thresholds, the data point is flagged as an anomaly
- This combination of criteria is more robust than using either model alone because it reduces false positives—only points that are abnormal in clustering and construction are considered anomalies.

5. METHODOLOGY

The dataset for this study was collected using Wireshark, which captured raw network traffic on a WiFi network. The data contained attributes such as source, destination, protocol, and packet length. After preprocessing, the data were incorporated into a hybrid anomaly detection model by adding the following features.

1. **Autoencoder:** Autoencoder was used to reconstruct normal network traffic patterns. It consists of an encoder and a decoder. The encoder compresses the input data into a low- dimensional representation and the decoder reconstructs the data. The error of reconstruction, that is, the difference between the reconstructed input and output, was calculated for each data point. Network packets that exhibited significant reconstruction errors were flagged as potential anomalies.
2. **KMeans Clustering:** The data were clustered using KMeans, after being processed by the encoder of the autoencoder. KMeans assigns each data point to one of two groups (normal or anomalous). The clustering was based on the reduced dimensional data generated by the autoencoder, so that the clustering process could focus on compressed features Each point distance between cluster centers was used as an additional search feature anomaly detection.
3. **Hybrid anomaly detection:** The anomalies were detected based on two factors:

- Return error from autoencoder.
- KMeans Distance to the cluster centroid.

Data points that exceeded a predefined threshold were flagged as non-abnormal for the item. Final discrepancies were identified by combining the results of both methods to reduce false positives. The hybrid model was trained and tested using the collected network data. The number of epochs was set to 50, and the model's performance was monitored using the training and validation loss.

5. RESULT AND ANALYSIS

ISSN 2394 - 7780

In anomaly detection analysis, we compared the effectiveness of using KMeans only, Autoencoder only, hybrid model (combination of KMeans and Autoencoder) to detect anomalies in network traffic data. The results show that hybrid model shows improved robustness in detecting anomalies, such as overlap, distance measures, and identifies errors in the reconstruction.

```
Average distance of potential anomalies (KMeans): 2.764255496262255
Average distance across all data (KMeans): 1.4172036034040245
Average reconstruction error of anomalies (Autoencoder): 1.0785305217452026
Average reconstruction error across all data (Autoencoder): 0.5506624113388701
```

Fig 5.1: The Result

Reported Variance Overlap:
The Venn diagram shows that 50% of the anomalies detected by KMeans are anomalies detected by Autoencoder, and vice versa. This combination shows that although the two models agree on some similarities, each model also finds a different variance (50% is different for each model). This diversity suggests that the hybrid approach takes advantage of the unique identification capabilities of the two models, and allows for better identification of potentially different features in the data set.

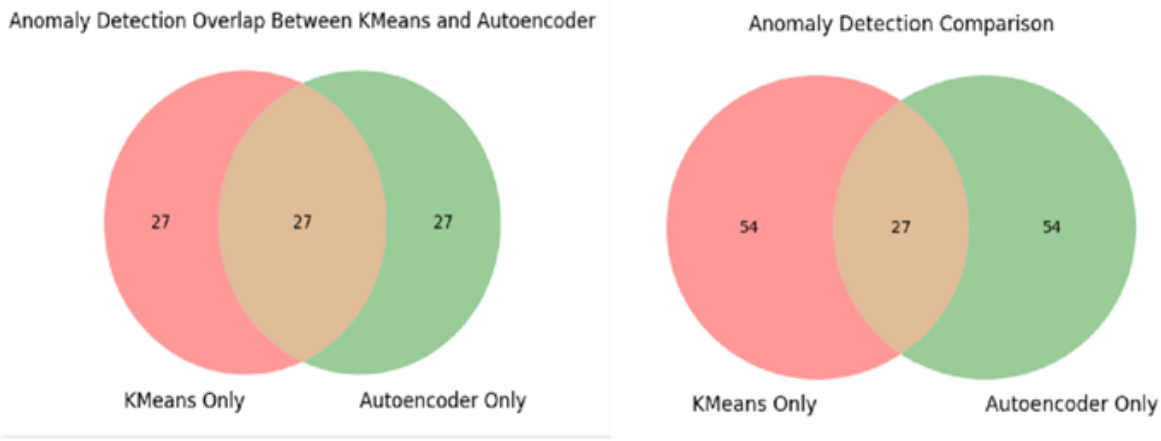

Fig 5.2: Anomaly Detection Overlap Between KMeans and Autoencoders with Comparison

Average distance in KMeans:
The mean distance of the KMeans-only anomalies to cluster centers is 2.764, which is notably larger than the mean distance over all data points (1.417) This high distance confirms that the anomalies detected by KMeans are indeed outliers in the feature space, but beyond that they have no further evidence of their abnormality.

Average reconstruction error of the autoencoder:
The average reconstruction error for the Autoencoder-only anomalies is 1.079, which is significantly higher than the reconstruction error for the entire data set (0.551) This high error indicates that these points deviate significantly away from Autoencoder's known patterns, and improves their classification as anomalies.

Advantages of the hybrid model:
When KMeans and Autoencoder detections are combined, the hybrid model collapses for anomalies to those detected by both methods, resulting in more refined anomalies These combined anomalies lie at higher distances from Chemeans cluster centers and high reconstruction errors, making it likely that they represent genuine anomalies.

This approach avoids false positives that can result from exposure to distance or repetition errors alone and highlights the robustness of the hybrid model to anomalies missed by KMeans or Autoencoder alone in the 19th century.

6. CONCLUSION

The results support the use of a hybrid model, as it combines the capabilities of KMeans and Autoencoder for more accurate and reliable anomaly detection The hybrid approach removes inconsistencies, resulting in anomalies with spatial divergence and structural distortion occurs. This dual validation reduces the risk of false positives and improves the overall model in detecting real anomalies in network traffic data without the need for ground truth labels.

7. FUTURE SCOPE

ISSN 2394 - 7780

This research could be extended in the future in several ways to improve the efficiency of anomaly detection in network traffic data and to use First, more advanced clustering techniques, such as DBSCAN or Gaussian Mixture Models, to handle complex information, nonlinear data structures and improve detection accuracy (GMM) could be investigated Furthermore, generative models such as variational autoencoders (VAEs) or generative adversarial networks (GANs), can capture a more nuanced picture of normal data, enabling the detection of subtle anomalies.

The inclusion of a semi-supervised learning may allow the model to benefit from smaller data sets, if available, for the limited detection limits between normal and abnormal cases improve Future work LSTMs or Temporal Convolutional Networks (TCNs) to capture sequential patterns of network traffic) and other temporal patterns can also be considered, improving anomaly detection for time-series data.

Another promising strategy is real-time anomaly detection, where the model is optimized to handle databases for immediate detection and response to network threats Besides, automatic hyperparameter tuning for boundaries and configurations of the hybrid model can be optimized and applications on different datasets.

Testing the model in different networks, such as enterprise networks or IoT, will prove to be generalizable. Enhancing pattern interpretation through interpretable AI (XAI) techniques such as Shapley Additive Explanations or LIME could improve the clarity of anomaly detection, strengthening the confidence of security analysts. The development of unsupervised performance simulations will also enable rigorous evaluation of anomaly detection models without relying on a ground truth label.

REFERENCES

[1] Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly Detection: A Survey. ACM Computing Surveys (CSUR), 41(3), 1-58.

[2] Ahmad, I., Basheri, M., Iqbal, M. J., & Rahim, A. (2016). Performance Comparison of Support Vector Machine, Random Forest, and Extreme Learning Machine for Intrusion Detection.

[3] Li, Y., Chen, D., Jin, Y., & Lin, W. (2019). Anomaly Detection in Network Traffic Based on Combination of Clustering and Deep Autoencoder.

[4] Lyu, X., Meng, X., & Li, Y. (2020). Anomaly Detection of Network Traffic Based on Deep Learning Models. Journal of Information Security and Applications, 53, 102529.

[5] Ahmed, M., Mahmood, A. N., & Hu, J. (2016). A Survey of Network Anomaly Detection Techniques. Journal of Network and Computer Applications, 60, 19-31.

[6] Basant Agarwal*, Namita Mittal. Hybrid Approach for Detection of Anomaly Network Traffic using Data Mining Techniques.

[7] Amuthan Prabakar Muniyandia , R. Rajeswarib , R. Rajaram. Network Anomaly Detection by Cascading K-Means Clustering and C4.5 Decision Tree algorithm

[8] Jaroslav Kopčan, Ondrej Škvarek, Martin Klimo. Anomaly detection using Autoencoders and Deep Convolution Generative Adversarial Networks.

[9] Gerhard Munz, ¨ Sa Li, Georg Carle, Traffic Anomaly Detection Using K-Means Clustering

[10] Tarem Ahmed, Boris Oreshkin and Mark Coates. Machine Learning Approaches to Network Anomaly Detection