**Anomaly Detection Using Deep Learning**

**The Challenge:**

* Anomaly detection is critical in network security, as it enables the identification of unusual patterns that may indicate security threats or network performance issues.
* This project uses the KDD Cup dataset, which is widely regarded for benchmarking anomaly detection algorithms in network traffic analysis.
* By leveraging deep learning techniques, this project aims to effectively detect anomalies in network data, providing a robust method for enhancing cybersecurity.

**Project Objective:**

* The objective of this project is to compare and analyse six deep learning models—Principal Component Analysis (PCA), Autoencoders, One-Class SVM, Variational Autoencoders (VAE), Seq2Seq, and Bidirectional Generative Adversarial Networks (BiGAN)—in their ability to classify network traffic data and detect anomalies.
* Each model is designed to identify normal patterns and flag unusual or suspicious data points, facilitating real-time detection of network intrusions.

**Dataset Details**

**KDD Cup Dataset:**

The KDD Cup dataset is a widely recognized dataset for anomaly detection and intrusion detection tasks. It contains network traffic data that is pre-labelled as either normal or one of several specific attack types, making it suitable for evaluating anomaly detection models. The dataset is structured to simulate a real network environment, providing a comprehensive benchmark for testing various detection models.

1. **Data Composition:**
   * + **Total Samples:** 4,898,431 records in the full dataset.
     + **Feature Count:** 41 features, representing various aspects of network traffic (e.g., protocol type, service, source bytes, destination bytes).
     + **Classes:** The dataset has two primary classes: normal and attack. Attacks are further divided into four main types:
       - **DoS (Denial of Service):** Overloading a system to deny access to legitimate users.
       - **R2L (Remote to Local):** Unauthorized access from a remote machine.
       - **U2R (User to Root):** Attempting to gain superuser privileges from normal user access.
       - **Probing:** Scanning for vulnerabilities in a network.
2. **Data Structure and Format:**

* **Training Set:** A balanced sample of normal and attack records.
* **Testing Set:** Includes additional attack types not present in the training set to evaluate model generalization.

1. **Feature Categories:**

* **Basic Features:** Basic attributes of the network connection, such as duration, protocol type, and service.
* **Content Features:** Attributes extracted from the data payload within a connection (e.g., number of failed login attempts).
* **Time-Based Traffic Features:** Features derived from network traffic patterns within a specific time window.
* **Host-Based Traffic Features:** Measures of network traffic on the host level, such as connection count over a larger time period.

**Preprocessing:**

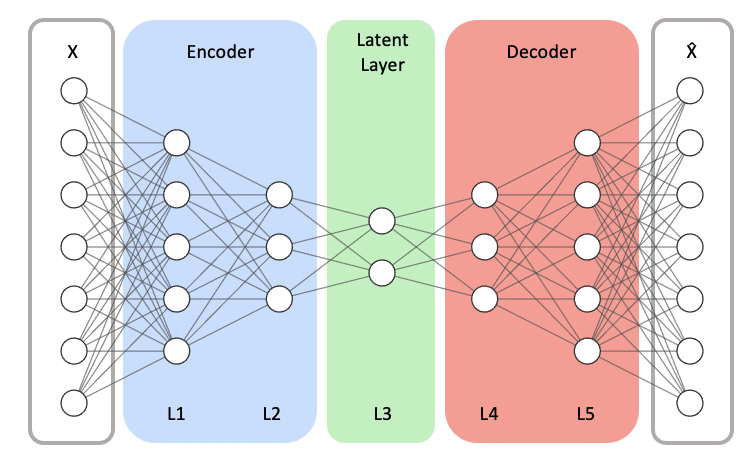
* + **Normalization:** To ensure consistent input, features were normalized to a similar scale.
  + **Label Encoding:** Attack types and protocol types were encoded to numeric values for model compatibility.

**Approach:**

**Model Architectures and Techniques:**

1. **Autoencoders:**

* Autoencoders are neural networks with two main parts: an encoder and a decoder.
* The encoder compresses the data into a lower-dimensional latent space, while the decoder reconstructs it back to the original form. When trained on normal data, the model becomes highly accurate at reconstruction.
* Anomalies, however, are harder to reconstruct and result in high reconstruction errors, signaling deviations from normal patterns.
* **Architecture of Autoencoders :**

****

1. **Variational Autoencoder (VAE):**

* The Variational Autoencoder (VAE) is a generative model that combines neural networks and probabilistic inference to learn efficient representations of data.
* It consists of an encoder that maps input data to a latent distribution (mean and variance), from which samples are drawn, and a decoder that reconstructs the original data from these latent representations.
* VAEs are particularly effective for anomaly detection, as they identify instances with high reconstruction errors, indicating that these anomalies do not conform to the patterns learned during training. This allows VAEs to capture complex data distributions and generate new, similar samples.
* **Architecture of Variational-Autoencoders(VAE) :**

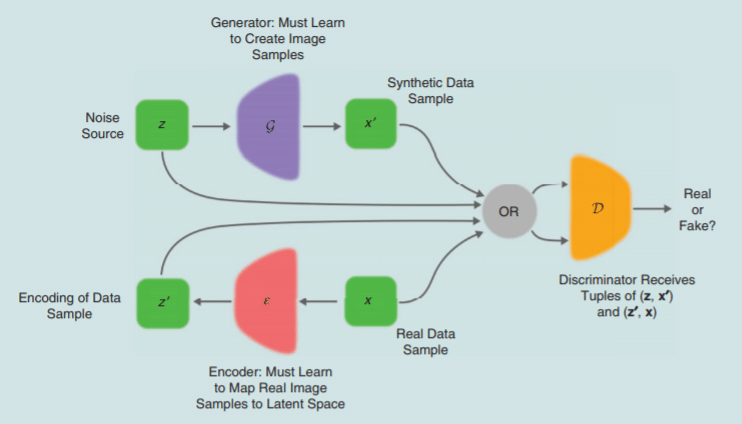
**A diagram of a algorithm

Description automatically generated**

1. **Bidirectional Generative Adversarial Networks (BiGAN):**

* BiGANs extend traditional GANs by including an encoder that maps data to a latent space, creating a bidirectional process. The generator learns to create realistic “normal” data samples, and the discriminator differentiates between real and synthetic samples.
* Anomalies are detected when the discriminator identifies data that doesn’t align with the patterns it learned. BiGANs are highly effective for complex anomaly detection because of their ability to understand the underlying distribution of normal data.

**Architecture of BiGAN :**



BiGAN is an advanced GAN structure that simultaneously learns a generation and an inference process, which is particularly valuable for anomaly detection in complex, high-dimensional datasets like KDD Cup. The BiGAN architecture is composed of three key components: the generator, discriminator, and encoder.

1. **Generator (G):**

Input: Random noise vector, zzz, from a latent space.

Output: Synthetic network traffic features that resemble the normal class in the dataset.

The generator learns to produce data samples that mimic normal network patterns, thus establishing a standard for comparison.

1. **Encoder €:**

Input: Real data samples (network traffic features).

Output: Latent vector representation of each sample.

The encoder maps real network data into a latent space, capturing the underlying data structure. This latent representation is crucial for distinguishing between normal and anomalous patterns.

1. **Discriminator (D):**

Input: Pairs of real data with encoded latent vectors and generated data with input noise vectors.

Output: Probability distinguishing real (normal) data from generated (potentially anomalous) data.

The discriminator’s objective is to identify whether a data-latent pair comes from real network traffic or the generator. It thus learns to capture the subtle differences between normal and anomalous data representations.

**Anomaly Detection:**

During testing, BiGAN generates latent representations for real network samples. Samples that deviate significantly from the normal latent space (based on a threshold) are flagged as anomalies, allowing the model to detect complex attack patterns efficiently.

1. **Seq2Seq :**

* The Sequence-to-Sequence (Seq2Seq) model is a deep learning architecture designed for tasks where input and output sequences may differ in length, such as language translation, summarization, and anomaly detection in time series data.
* It typically comprises an encoder, which processes the input sequence into a fixed-size context vector, and a decoder, which uses this context to generate the output sequence step-by-step.
* By capturing dependencies across sequences, Seq2Seq models are well-suited to detecting irregularities in sequential data by analyzing patterns and variations over time.

**Architecture of Seq2Seq :**

A diagram of a computer network

Description automatically generated

1. **Principal Component Analysis (PCA):**

* PCA is a statistical method used for reducing dimensionality by capturing the principal components (patterns) that account for the most variance in the data.
* In anomaly detection, PCA helps by compressing data into fewer components that represent normal patterns.
* Anomalies are identified when data points have high reconstruction errors, meaning they deviate significantly from the principal components of normal data.
* **Architecture of PCA :**

A diagram of a graph

Description automatically generated

1. **One-Class SVM (OC-SVM):**

This variation of Support Vector Machines creates a decision boundary around normal data by transforming it into a higher-dimensional space, allowing a hyperplane to separate normal points from outliers. OC-SVM works well when trained on data representing typical behavior, using this boundary to flag points lying far from the decision boundary as anomalies.

1. **Architecture of OC-SVM :**

**A diagram of a circle and a square

Description automatically generated**

**How to Decide on a Modeling Approach ?**

* Given the differences between the deep learning methods discussed above (and their variants), it can be challenging to decide on the right model.
* When data contains sequences with temporal dependencies, a sequence-to-sequence model (or architectures with *LSTM layers*) can model these relationships well, yielding better results.
* For scenarios requiring principled estimates of uncertainty, *generative* models such as a VAE and GAN based approaches are suitable. For scenarios where the data is images, AEs, VAEs and GANs designed with *convolution layers* are suitable. The following table highlights the pros and cons of the different types of models, to provide guidance on when they are a good fit.

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Strengths | Weaknesses | Best Use Cases |
| Autoencoder | - Efficient at reconstructing normal patterns. - Identifies anomalies via reconstruction loss. | - Struggles with complex, high-dimensional data. - Sensitive to hyperparameter selection. | - Scenarios with well-defined normal data patterns. |
| Variational Autoencoder (VAE) | - Adds probabilistic modeling to capture data uncertainty. - Capable of generating new samples. | - More complex, with hyperparameter tuning challenges. - Requires larger data for effective performance. | - Use cases with uncertain data distributions. |
| BiGAN (Bidirectional GAN) | - Combines generative and discriminative models for robust feature learning. - Produces high-quality data generation. | - Training is complex and can be unstable. - Requires balanced tuning between generator and discriminator. | - Complex data scenarios with data generation needs. |
| Seq2Seq Model | - Ideal for time-dependent anomaly detection. - Captures sequential dependencies. | - Computationally intensive for long sequences. - Challenging to implement on large datasets. | - Financial or IoT applications with time-series data. |
| One-Class SVM | - Interpretable, effective in high-dimensional spaces. - Straightforward implementation. | - Sensitive to kernel selection. - Less robust in highly imbalanced datasets. | - Scenarios with limited labeled anomalies. |
| PCA (Principal Component Analysis) | - Reduces dimensionality, focusing on key variance. - Straightforward, interpretable results. | - Assumes linear relationships in data. - Limited to capturing variance, not complex patterns. | - Preprocessing for high-dimensional datasets. - Useful for exploratory data analysis and dimensionality reduction prior to model training. |

**Recommendations for Use :**

* For datasets with complex sequential or temporal relationships, Seq2Seq is recommended.
* For multidimensional datasets without inherent temporal structure, Autoencoders and VAEs provide reliable anomaly detection.
* BiGAN offers additional flexibility for data with unknown or diverse distributions, allowing synthetic sample generation.
* PCA is valuable as a preprocessing step to reduce dimensionality and noise, enhancing the efficiency of downstream models.

**Model Performance Comparison:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | **Model** | **Accuracy (Your Results)** | **Accuracy (Research -Papers)** | | **AutoEncoder** | **84.6** | **89.8** | | **Variational AutoEncoder (VAE)** | **93.0** | **95.6** | | **BiGAN (Bidirectional GAN)** | **90.3** | **92.3** | | **Seq2Seq** | **93.8** | **94.0** | | **PCA (Principal Component Analysis)** | **91.6** | **94.0** | | **OCSVM (One-Class SVM)** | **94** | **94.1** | |

**Conclusion**

**Overall Model Performance Comparison :**

* BiGAN stands out as the best model for anomaly detection in this project. It achieves a high accuracy (0.923) and AUC-ROC (0.947), along with robust precision, recall, and F1 scores. Its generative capacity enables it to capture complex, subtle patterns in the data, making it particularly well-suited for modeling both normal and anomalous behaviors.
* OCSVM also performs exceptionally well, with the highest accuracy (0.949) and AUC-ROC (0.958). This model is strong in distinguishing outliers, though it may be less effective with very complex anomalies compared to BiGAN.
* VAE follows closely behind, with a good balance of accuracy and AUC-ROC scores, benefiting from its probabilistic approach that allows for effective generalization.

In conclusion, BiGAN is the best overall choice due to its ability to model complex data patterns and detect subtle anomalies, especially in high-dimensional, nonlinear datasets. OCSVM and VAE are also strong options, with OCSVM being a solid choice if simpler boundary-based anomaly detection is sufficient.

**Model Limitations:**

* + PCA: Susceptible to noise in high-dimensional data, limiting its anomaly detection accuracy.
  + Autoencoders: Can suffer from overfitting if not carefully regularized, which may limit generalization.
  + One-Class SVM: Computationally expensive on large datasets due to boundary definition complexity.
  + BiGAN: Resource-intensive due to its adversarial training, though it produces robust anomaly detection for complex data.

References and Research Papers

1. <https://ar5iv.labs.arxiv.org/html/2212.13558>   
   Kalyan Veeramachaneni *MIT*Cambridge, USA
2. <https://arxiv.org/abs/2212.13558> **AER: Auto-Encoder with Regression for Time Series Anomaly Detection 2022**
3. <https://arxiv.org/abs/2408.13561> **Variational Autoencoder for Anomaly Detection: A Comparative Study 2024**
4. <https://www.researchgate.net/publication/346066479_An_improved_BiGAN_based_approach_for_anomaly_detection/fulltext/5fb9ea7c458515b797603346/An-improved-BiGAN-based-approach-for-anomaly-detection.pdf> An improved BiGAN based approach for anomaly detection 2020
5. <https://arxiv.org/abs/1409.3215> **Sequence to Sequence Learning with Neural Networks**
6. <https://arxiv.org/abs/1312.6114> **Auto-Encoding Variational Bayes**
7. <https://arxiv.org/abs/1312.6114> **Auto-Encoding Variational Bayes**