NTGAN

Praneeth Nayak & Bhanu Prasad

July 8, 2023

1 Introduction

This project aims to leverage Generative Adversarial Networks (GANs) in computer networks to improve network data analysis and security. By training a GAN on a labeled dataset of network traffic, we can generate synthetic data resembling normal network behavior. Using this synthetic data, we will develop an anomaly detection algorithm to identify deviations from normal patterns, enhancing network monitoring and security capabilities. The project explores the intersection of deep learning and network analysis, with the potential to create more robust and adaptable systems for real-time network traffic analysis.

2 Requirements

The requirements for a project involving computer networks and GANs

- **Programming Skills:** The project requires a good knowledge of programming in Python. The project also requires a good knowledge of the PyTorch library.
- **Knowledge of Computer Networks:** The project requires a good knowledge of computer networks, especially the TCP/IP protocol stack.
- **Familiarity with GANs:** Gain a solid understanding of Generative Adversarial Networks (GANs) and their architecture. Study the GAN training process, loss functions, and techniques for generating synthetic data.

- **Data Collection and Preprocessing:** Collect and preprocess the data for training the GAN. The data should be collected from a real network and should be representative of the network traffic.

- **GAN Training and Optimization:** Train the GAN on the collected data and optimize the GAN to generate realistic synthetic data. Experiment with different hyperparameters, network architectures, and training strategies to achieve good performance.

- **Anomaly Detection Techniques:** Study the different anomaly detection techniques and choose the most suitable technique for the project. Implement the chosen technique and evaluate its performance.

- **Evaluation Metrics:** Choose the most suitable evaluation metrics for the project and evaluate the performance of the anomaly detection technique. Common metrics include accuracy, precision, recall, F1 score, and area under the ROC curve (AUC-ROC).

- **Testing and Validation:** Test the performance of the anomaly detection technique on the synthetic data generated by the GAN. Validate the performance of the anomaly detection technique on real network traffic.

3 Project Roadmap

3.1 Project Planning and Research

- Define the project objectives, scope, and requirements.

- Conduct literature review and research on GANs, network traffic analysis, and anomaly detection techniques.

- Identify potential datasets for network traffic analysis.

3.2 Data Collection and Preprocessing

- Acquire a suitable network traffic dataset, either from publicly available sources or by generating simulated data.

- Preprocess the dataset to clean the data, remove noise, and transform it into a suitable format for training GANs.

3.3 GAN Training

- Design the architecture of the GAN model for network traffic generation.
- Split the dataset into training and validation sets.
- Train the GAN model using the training dataset, experimenting with different hyperparameters and optimization techniques.
- Monitor the training process, evaluate the quality of the generated traffic data, and fine-tune the model as needed.

3.4 Anomaly Detection Algorithm

- Develop an anomaly detection algorithm that leverages the GAN-generated network traffic data.
- Define appropriate features and techniques for analyzing the traffic patterns and identifying anomalies.
- Implement a threshold-based approach or explore machine learning algorithms for anomaly detection.
- Train and validate the anomaly detection algorithm using the validation dataset.

3.5 Evaluation and Performance Analysis

- Evaluate the performance of the anomaly detection algorithm using appropriate evaluation metrics such as accuracy, precision, recall, F1 score, and AUC-ROC.
- Compare the results with existing methods or baselines for anomaly detection in network traffic.
- Perform in-depth analysis to understand the strengths and limitations of the developed approach.

3.6 System Development and Integration

- Design and implement a functional system that integrates the trained GAN model and the anomaly detection algorithm.
- Develop a user-friendly interface for real-time analysis of network traffic and visualization of detected anomalies.
- Ensure the system can process incoming network traffic and provide timely alerts or notifications when anomalies are detected.

3.7 Testing and Validation

- Conduct extensive testing of the system using representative network traffic data.
- Validate the system's performance under different network conditions and scenarios.
- Fine-tune the system and address any issues or limitations identified during testing.

3.8 Documentation and Reporting

- Maintain comprehensive documentation throughout the project, including the dataset used, methodologies employed, and implementation details.
- Prepare a final report summarizing the project objectives, methodology, findings, and any future recommendations or improvements.