OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG

Generative Adversarial Networks for Time Series

Autonomous Multisensor Systems Group
Institute for Intelligent Cooperating Systems
Faculty of Computer Science
Otto von Guericke University, Magdeburg

10.07.2024

Team Members: Archana Yadav, Gowtham Premkumar, Shweta Bambal

Autonomous Multisensor Systems

Contents

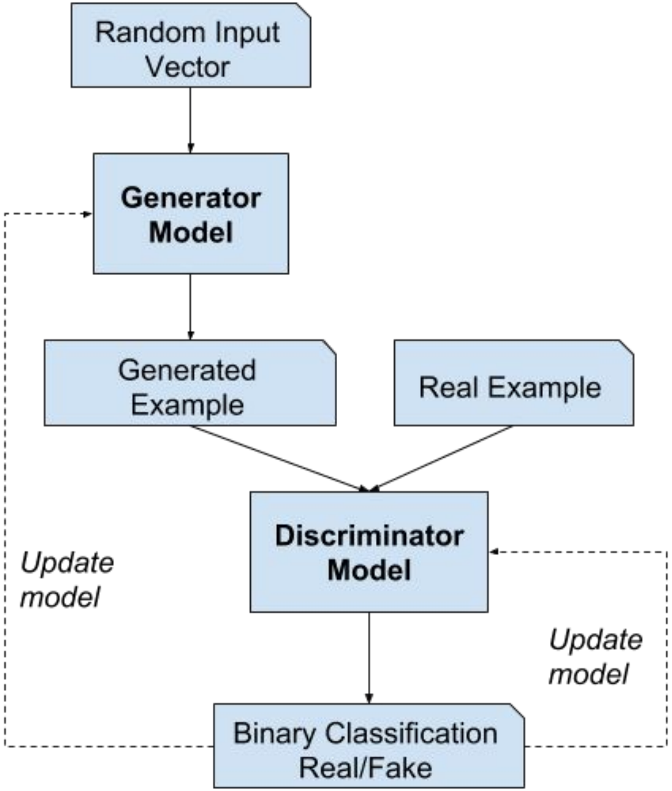
Autonomous Multisensor Systems

OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG

Introduction

Generative Adversarial Networks (GANs) are made up of two neural networks, **a discriminator and a generator.** They use adversarial training to produce artificial data that is identical to actual data.

Figure 1: Images generated by a GAN created by NVIDIA.



https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/

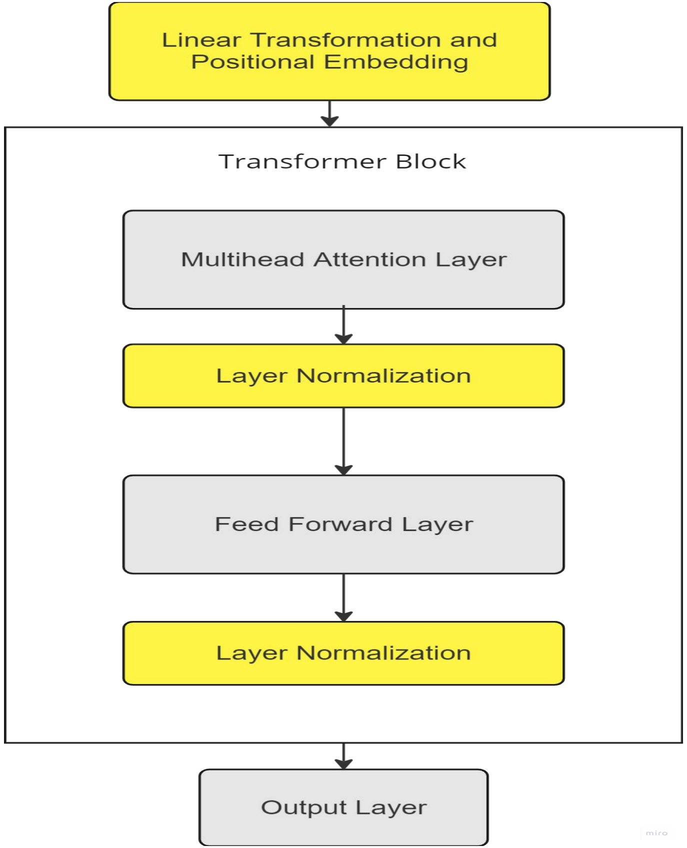
Transformer GAN for Time Series

- Transformers initially designed to handle long sequential data without vanishing gradient problem.
- Consequently, Transformer GAN models are theoretically superior to RNN-based models for time-series data.
- A leading model in deep learning, outperforming CNNs for images and RNNs for sequential data, due to its self-attention layers
- Enhances synthetic data quality, improve training efficiency, especially in image and text generation.

Transformer GAN - Advantages

Advantages	Description
Long-range dependencies	Better captures complex temporal patterns and relationships over long sequences
Scalability	Scales well to handle larger datasets and longer sequences
Attention mechanism	Focuses on relevant parts of input for more coherent outputs
Parallel processing	Can process input sequences in parallel, improving efficiency
Performance	Consistently outperforms existing methods in generating high-quality synthetic data

Transformer GAN - Model Architecture



https://arxiv.org/abs/2202.02691

Transformer GAN - Evaluation Results

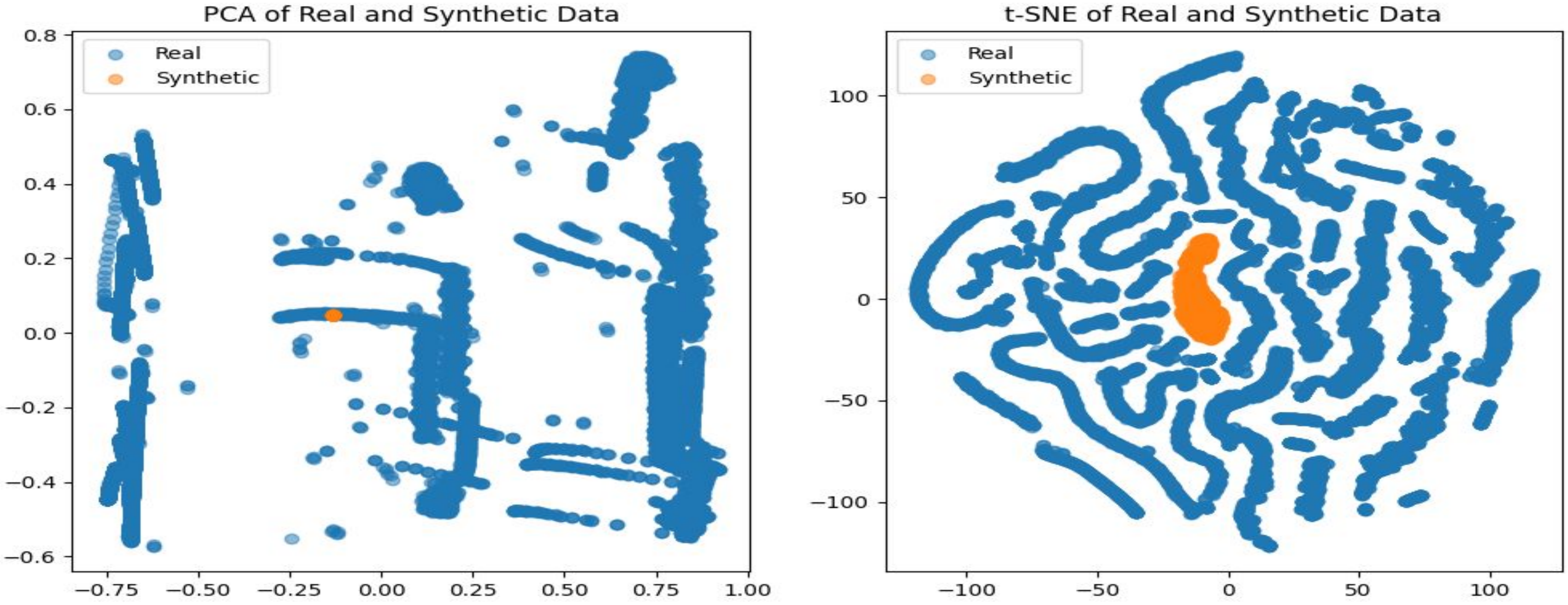


Fig : Results for 1000 iterations

TimeGAN

Synthetic Data

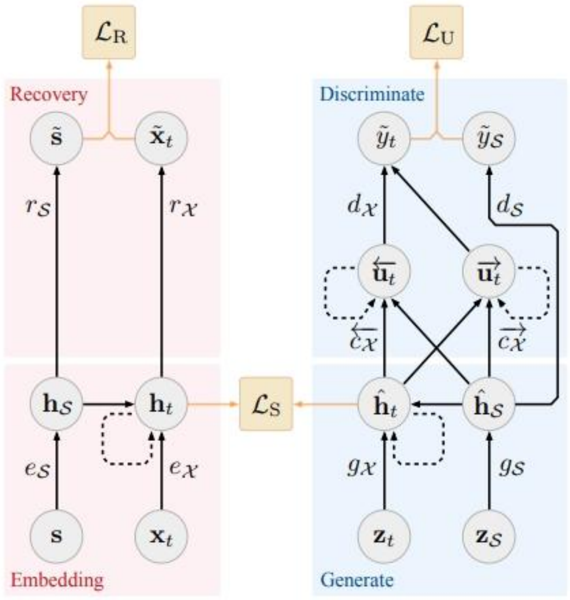
Embedder
Maps the original high-dimensional feature space to a lower-dimensional latent space.

Recovery
Maps the latent representations back to the original feature space.

Generator
Generates latent space representations from random noise.

Supervisor
Generates the next sequence in the latent space from the previous sequence.

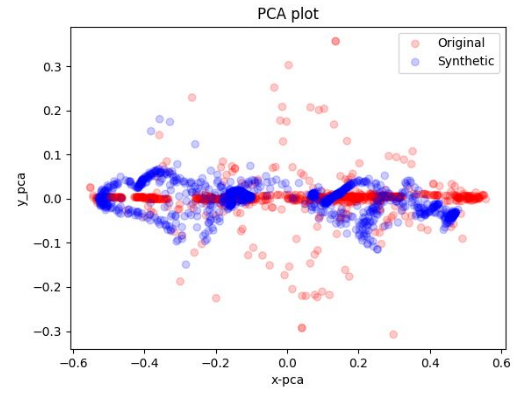
Discriminator
Distinguishes between real and synthetic time-series data.

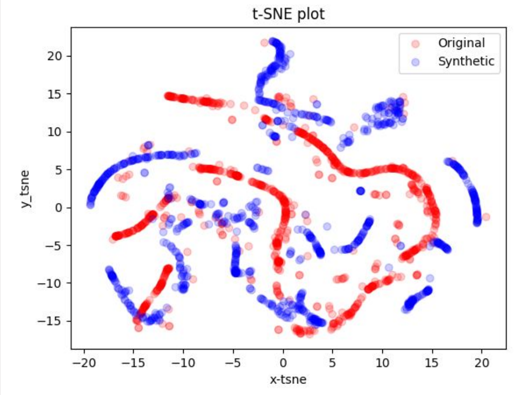


(a) TimeGAN

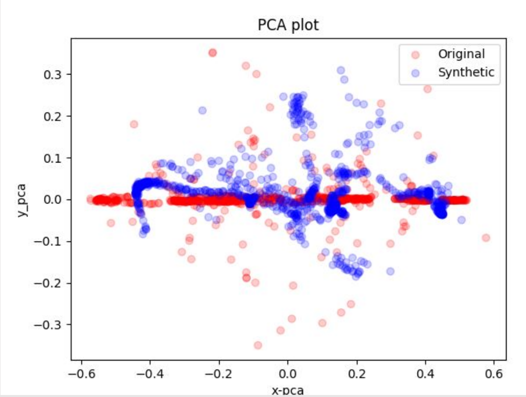
TimeGAN - Results for LSTM

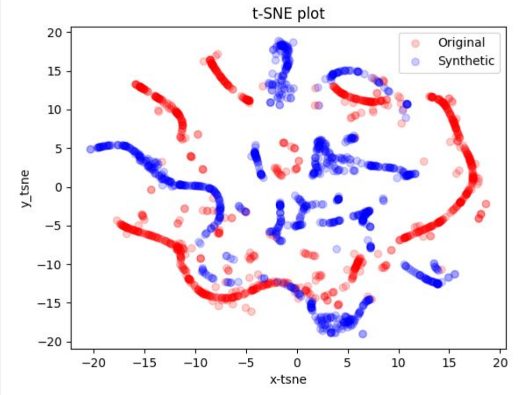
Results for 10k Iterations





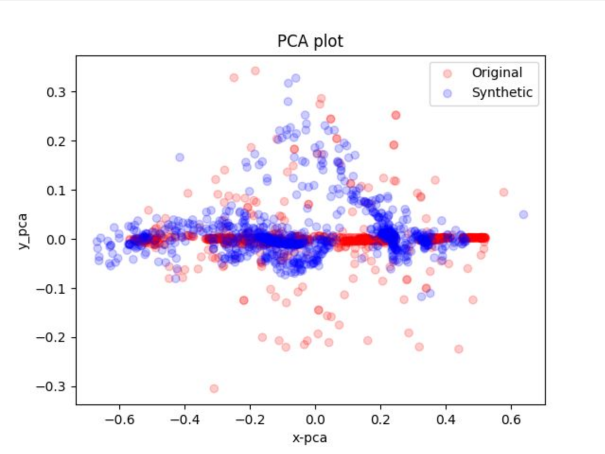
Results for 50k Iterations

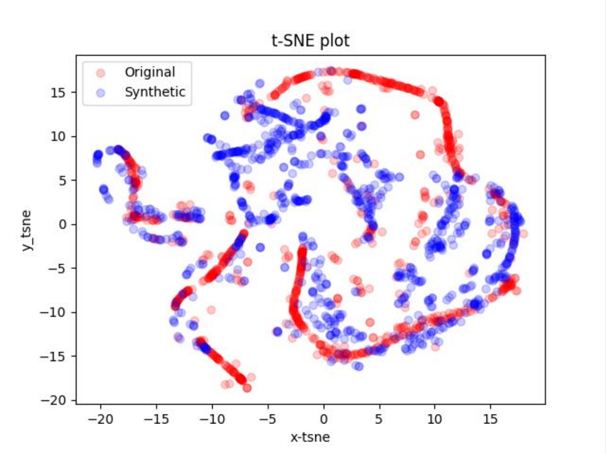




TimeGAN - Results for GRU

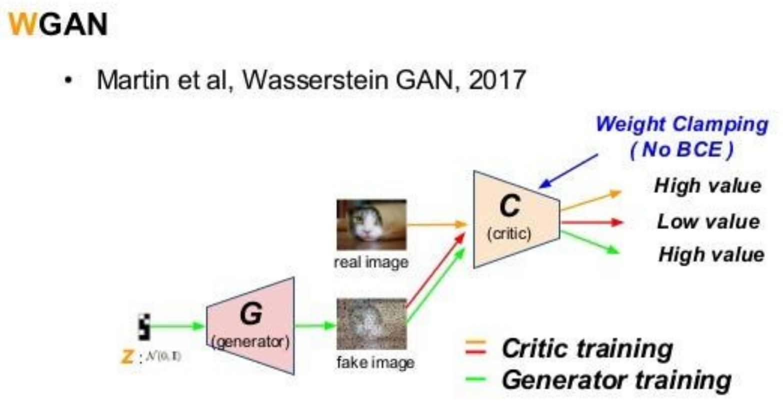
Results for
10k Iterations





Wasserstein GAN (WGAN)

- **Generative Adversarial Networks (GANs):** A deep learning framework with two neural networks:
  - **Generator:** Learns to create new data instances that resemble the real data.
  - **Critic :** Tries to distinguish between real and generated data the can be any real number rather than [0,1]



- **Loss Function**: Wasserstein loss, which measures the Earth-Mover distance between real and generated data distributions, providing better gradients for training the generator.

- **Time Series Application:**
  - Trains on existing time series data.
  - Generator creates new time series that capture the statistical properties of the real data.

WGAN

Wasserstein Distance: Also known as the Earth-Mover distance, it measures the cost of transporting mass to transform one probability distribution into another.

$$W(\mathbb{P}_r, \mathbb{P}_g) = \inf_{\gamma \in \Pi(\mathbb{P}_r, \mathbb{P}_g)} \mathbb{E}_{(x,y)\sim\gamma}[||x - y||]$$

Eq. 1: Wasserstein Distance between distributions P_r and P_g.

Stable Training:

- Reduced mode collapse and vanishing gradients ensure coherent sequence generation.
- Suitable for capturing long-term dependencies in time series data.

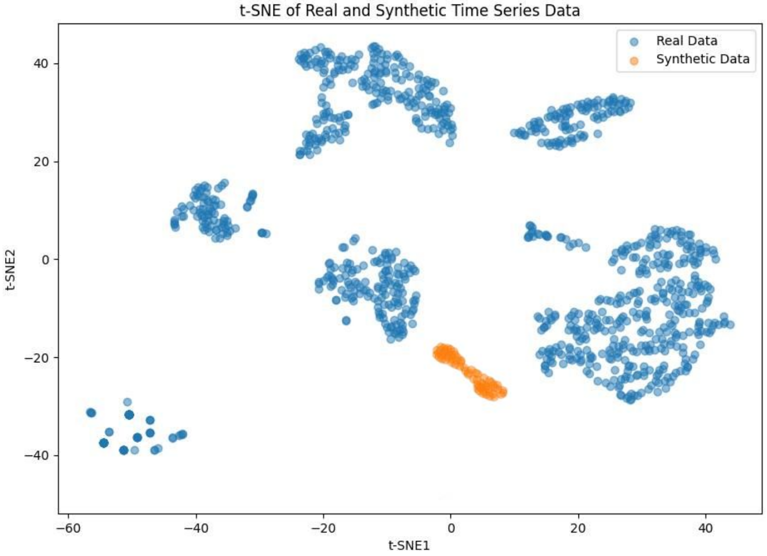
Flexibility:

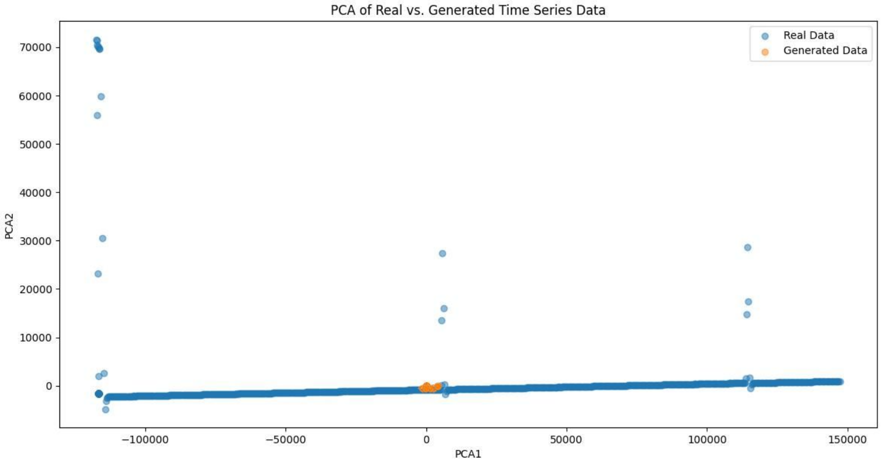
- Suitable for both univariate and multivariate time series data.

High-Dimensional Data:

- Effective in generating realistic high-dimensional time series data.

WGAN - Results





Way Forward

1. Fine tuning the hyperparameters to generate better results for the given dataset.

2. Identify advanced **evaluation metrics** specifically designed for **GAN-generated time-series data** and use them to assess the quality and realism of the generated time-series data.

3. Develop simple and clear **visualizations** to compare the real and generated time-series data.

References

1. https://www.geeksforgeeks.org/generative-adversarial-network-gan/
2. Xiaomin Li, Vangelis Metsis, Huangyingrui Wang, Anne Hee Hiong Ngu. (*2022*) TTS-GAN: A Transformer-based Time-Series Generative Adversarial Network. https://arxiv.org/abs/2202.02691
3. Yoon, Jinsung & Jarrett, Daniel & Schaar, Mihaela. (2019). Time-series Generative Adversarial Networks. https://www.researchgate.net/publication/344464212_Time-series_Generative_Adversarial_Networks
4. Padmanaba Srinivasan, William J. Knottenbelt. (2022) Time-series Transformer Generative Adversarial Networks https://arxiv.org/abs/2205.11164
5. https://www.techtarget.com/searchenterpriseai/tip/GAN-vs-transformer-models-Comparing-architectures-and-uses
6. https://blog.seduca.ai/id/57363/

OTTO VON GUERICKE
UNIVERSITÄT
MAGDEBURG

Generative Adversarial Networks for Time Series

Thank You For Your Attention!

Autonomous Multisensor Systems