Learning Traffic Anomalies from Generative Models on Real-Time Observations

Fotis I. Giasemis, Alexandros Sopasakis

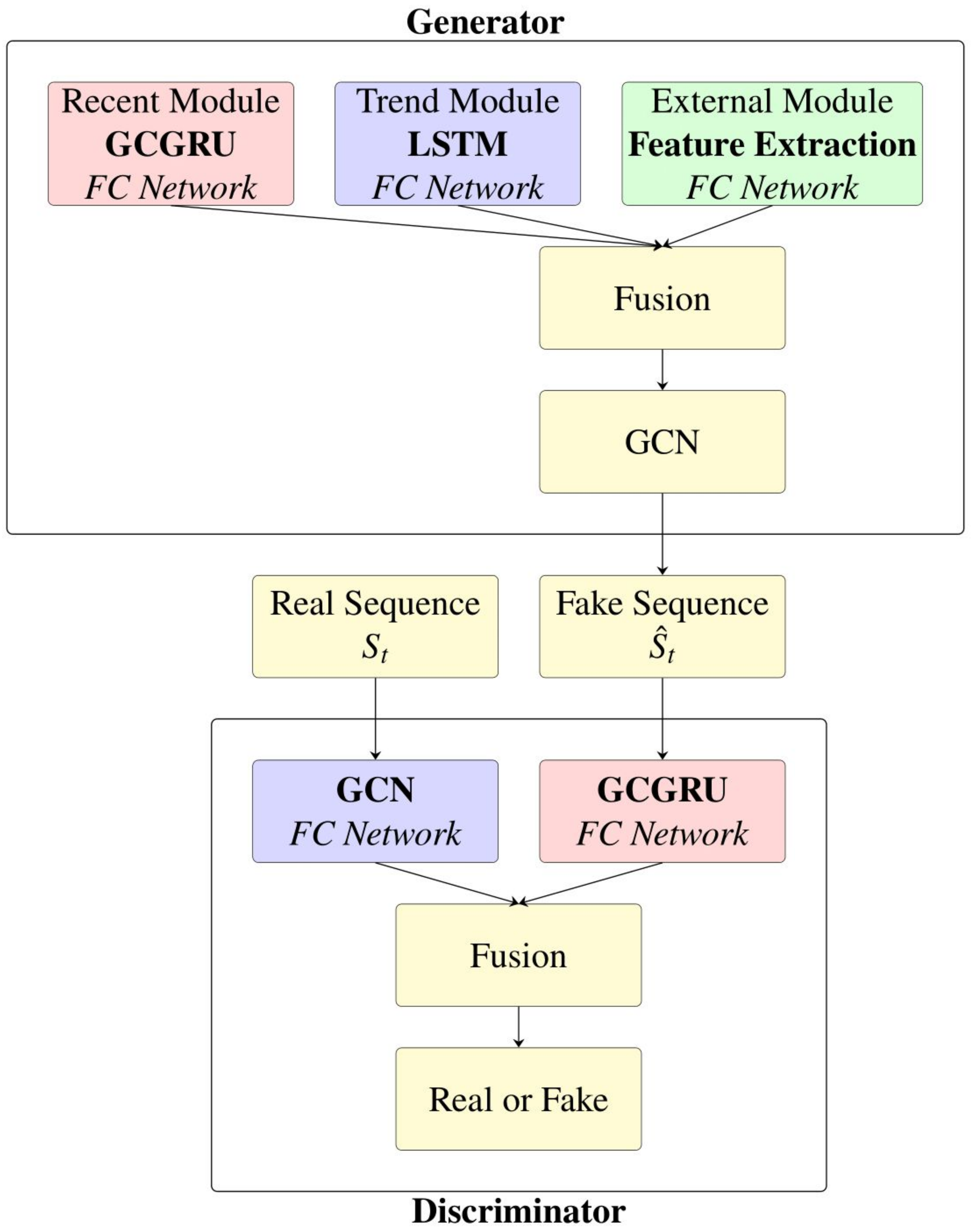
10th IEEE International Conference on Signal and Image Processing
Wuxi, China, July 2025

arXiv:2502.01391

github.com/fgias/traffic-anomaly-detection

Spatiotemporal Graph Convolutional Adversarial Network (STGAN)

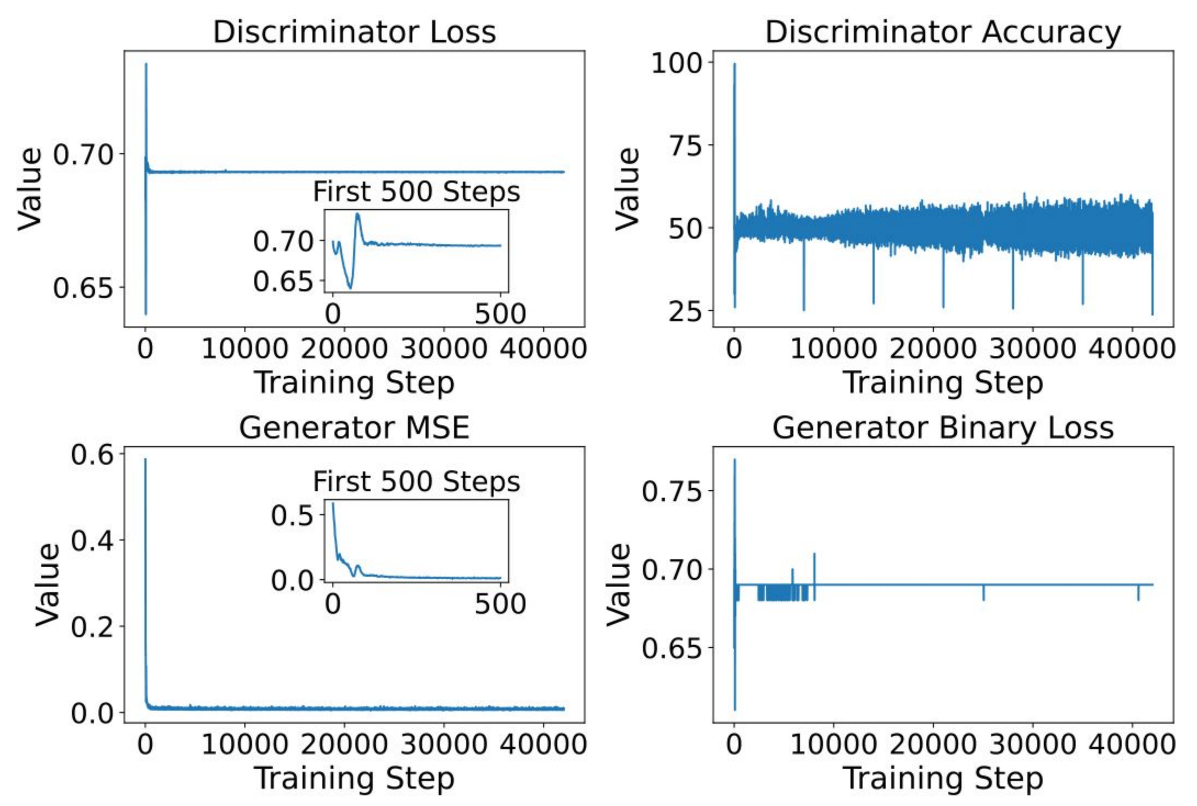
- Accurate detection of traffic anomalies is crucial for effective urban traffic management and congestion mitigation.
- STGAN framework combines Graph Neural Networks and Long Short-Term Memory networks to capture complex spatial and temporal dependencies in traffic data
- Real-time, minute-by-minute observations from 42 traffic cameras across Gothenburg, Sweden
- Preprocessing → Flow metric representing vehicle density: number of detected vehicles / maximum vehicle capacity
- Training: April to November 2020
- Validation: November 14–23, 2020
- Our results demonstrate that the model effectively detects traffic anomalies with high precision and low false positive rates
- The detected anomalies include camera signal interruptions, visual artifacts, and extreme weather conditions affecting traffic flow
- **Spatiotemporal Generator**: Generates predicted sequences of traffic data
- **Spatiotemporal Discriminator**: Distinguishes between real and generated sequences
- Recent Module: Captures short-term spatiotemporal dependencies using a Graph Convolutional Gated Recurrent Unit (GCGRU)
- Trend Module: Learns long-term temporal patterns using an LSTM network
- External Module: Incorporates external factors (e.g., time of day, day of the week) using a fully connected layer
- These inputs are fused and passed through a GCN to model spatial dependencies, generating a fake sequence
- The discriminator, using GCGRU and GCN modules, evaluates the generated sequence against real sequences
- Anomaly score: combine discriminator error and generator error



Algorithm 1 STGAN Training Procedure

Require: Training data $\{\mathbf{S}_{v,t}\}$, initial parameters $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$, learning rates η_G, η_D

while not converged **do**
    Generator Update:
    Generate sequences $\hat{\mathbf{S}}_{v,t} = G_\theta(v,t)$
    Compute generator loss $\mathcal{L}_G(\boldsymbol{\theta})$ using Eq. (B.6)
    Update generator parameters: $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta_G \nabla_{\boldsymbol{\theta}} \mathcal{L}_G(\boldsymbol{\theta})$
    Discriminator Update:
    Evaluate discriminator on real data: $D_\phi(\mathbf{S}_{v,t})$
    Evaluate discriminator on generated data: $D_\phi(\hat{\mathbf{S}}_{v,t})$
    Compute discriminator loss $\mathcal{L}_D(\boldsymbol{\phi})$ using Eq. (B.7)
    Update discriminator parameters: $\boldsymbol{\phi} \leftarrow \boldsymbol{\phi} - \eta_D \nabla_{\boldsymbol{\phi}} \mathcal{L}_D(\boldsymbol{\phi})$
end while



Results and Evaluation

- Calculate the anomaly scores of all the data in the test set
- Label top K% anomaly scores as anomalies
- Camera Signal Cut/Restart → Signals due to problems with the functioning of the camera: Nov. 16, 21 and 23, 2020
- Visual Artifacts → Anomalies triggered due to the visual quality of the input
- Extreme Weather Conditions → Anomalies due to developments in the weather, producing disturbances in the traffic flow: **Nov. 19, 2020**
- Anomaly at Nov. 19, 2020: At the anomaly time, heavy snowfall started, which in turn disturbed the normal traffic flow, eventually triggering an anomaly

Anomaly Type	Number of Anomalies
Camera Signal Cut/Restart	71
Visual Artifacts	2
Extreme Weather Conditions	2
True Positives	75
False Positives	6
Total	81