CONTINUAL LEARNING FOR ANOMALY DETECTION WITH VARIATIONAL AUTOENCODER

Universität Stuttgart

Felix Wiewel and Bin Yang

Institute of Signal Processing and System Theory, University of Stuttgart, Germany

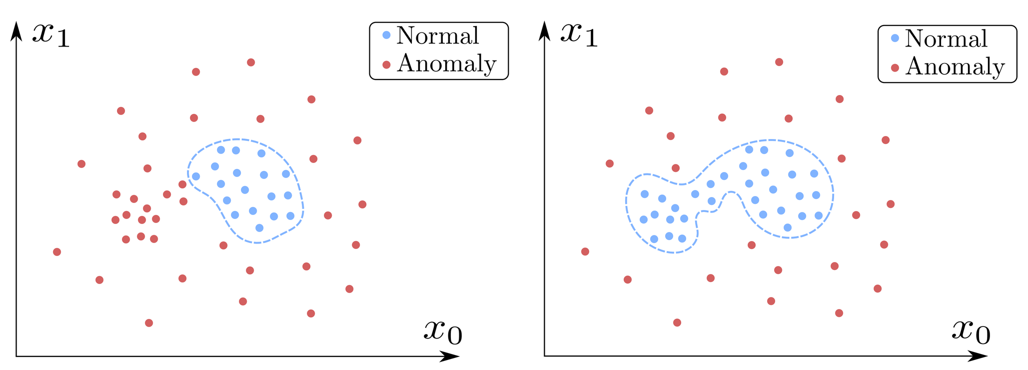
1. INTRODUCTION

Motivation

- Artificial neural networks suffer from catastrophic forgetting without protection mechanisms
- Methods proposed in literature mostly cover supervised and reinforcement learning problems
- Anomaly detection can also benefit from continual learning
- Using a Variational Autoencoder (VAE) for anomaly detection is a common method
- Generative capabilities of VAE are unused during anomaly detection

Problem Formulation

- **Given:** Sequence of data sets $\mathcal{D}^1, \dots, \mathcal{D}^N$ of normal data, where $\mathcal{D}^i$ represents the ith task
- **Goal:** Train a VAE for anomaly detection continually on all tasks
- **Restriction:** While training of task i only data set $\mathcal{D}^i$ is available



Timeinstance T_0 Timeinstance T_1

Contribution

- We propose an effective method for continual learning in anomaly detection with VAE
- Evaluation of proposed method on common data sets
- Study of degeneration effects

2. ANOMALY DETECTION USING VAE

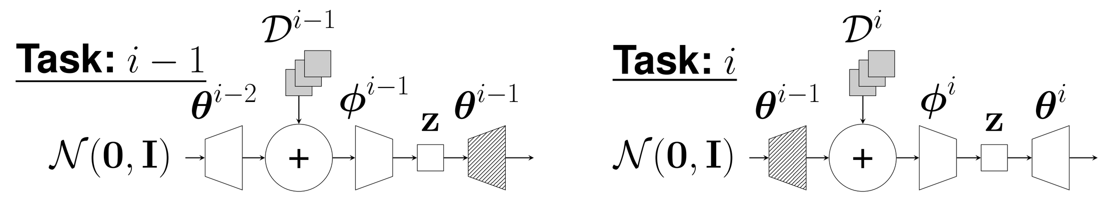
- Given dataset $\mathcal{D} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ of i.i.d. normal data
- VAE learns $\ln p(\mathbf{x}_1, \dots, \mathbf{x}_N) = \sum_{i=1}^{N} \ln p(\mathbf{x}_i)$ by maximizing Evidence Lower Bound (ELBO)

$$\mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta}; \mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\ln p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})] - D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z}))$$

- Distributions $q_\phi(\mathbf{z}|\mathbf{x})$ and $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$ are parameterized by encoder and decoder neural networks and $p(\mathbf{z})$ is a simple prior distribution
- After training an anomaly score is defined as $AI(\mathbf{x}) = \mathcal{L}(\boldsymbol{\phi}^\star, \boldsymbol{\theta}^\star; \mathbf{x})$
- A threshold based detector $AI(\mathbf{x}) < \gamma$ is used to detect anomalous samples

3. PROPOSED METHOD

- Key observations
  - After training on a data set $\mathcal{D}$ we can use $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$ and the prior $p(\mathbf{z})$ to generate samples
  - When using the VAE for anomaly detection, $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$ is only used to compute anomaly score
- We propose to use $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$ and the prior $p(\mathbf{z})$ to efficiently implement deep generative replay
  - We start with training a VAE on $\mathcal{D}^1$
  - For the following data sets we use $p_{\boldsymbol{\theta}^i}(\mathbf{x}|\mathbf{z})$ and the prior $p(\mathbf{z})$ to generate replay data for every batch of training samples and concatenate both
  - The amount of replay data in a training batch controls retention of previous tasks



Training process

```
Require: Sequence of datasets D^1, ..., D^N
  while Task t ≠ N do
    if Task t = 1 then
      while Not converged do
        Sample a batch B from D^1
        Update VAE on B
      end while
    else
      while Not converged do
        Sample a batch B from D^t
        Generate a batch B_GR of replay data
        Update VAE on concatenation of B and B_GR
      end while
    end if
    Copy weights of VAE decoder
    t = t + 1
  end while
```

4. EXPERIMENTS

Data sets

- KDD Cup 1999
  - Intrusion detection, 22 attacks, one normal class
  - ~ 4.9 million raw TCP dumps with 41 features
  - Categorical values are mapped to interval $[0, 1]$
- MNIST
  - Handwritten digit classification
  - 60000 training and 10000 test images with dimension $28 \times 28 \times 1$

Continual learning tasks

- KDD CUP 1999
  - Start with normal class from data set
  - Each task expands this definition by one attack
- MNIST
  - Start with digit 0 as normal class
  - Each task expands this definition by the next higher digit
- The proposed method, using Generative Replay (GR), is compared with Elastic Weight Consolidation (EWC)
- Upper Bound (UB) is given by joint training on all previous data sets $\mathcal{D}^j$ with $j \leq i$
- Lower Bound (LB) is given by a original VAE

Study of degeneration effects

- Capacity of the VAE is limited, i.e. replayed data is not perfect
- At first the VAE is trained on the latest task of each data set, performance on this task is considered a baseline (KDD Cup 1999: KB, MNIST: MB)
- The VAE is trained with repeated generative replay on the same task, which leads to a degradation of performance (KDD Cup 1999: KDG, MNIST: MDG)

VAE ARCHITECTURE

- Fully connected
- Symmetric structure

Network architecture

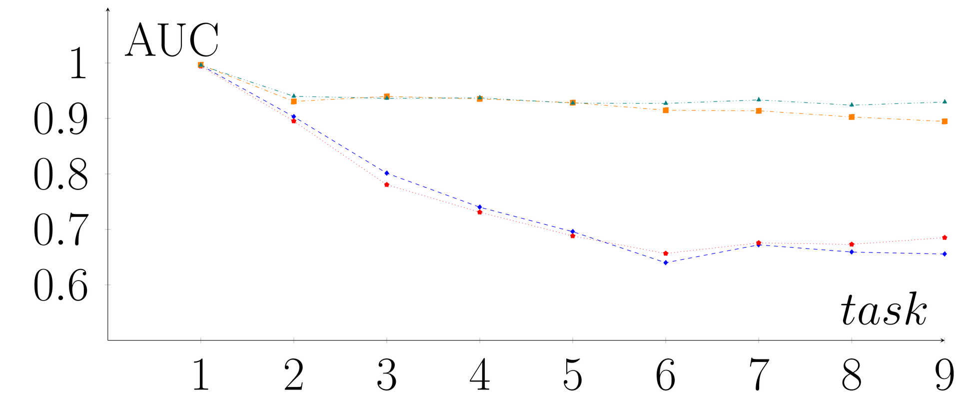
Layer	Neurons	Activation function
Input	784	-
Enc0	400	ReLU
Enc1	300	ReLU
Enc2	200	ReLU
Enc3	100	ReLU
Latent	50/50	-/Softplus
Dec0	100	ReLU
Dec1	200	ReLU
Dec2	300	ReLU
Dec3	400	ReLU
Output	784	Sigmoid

5. RESULTS

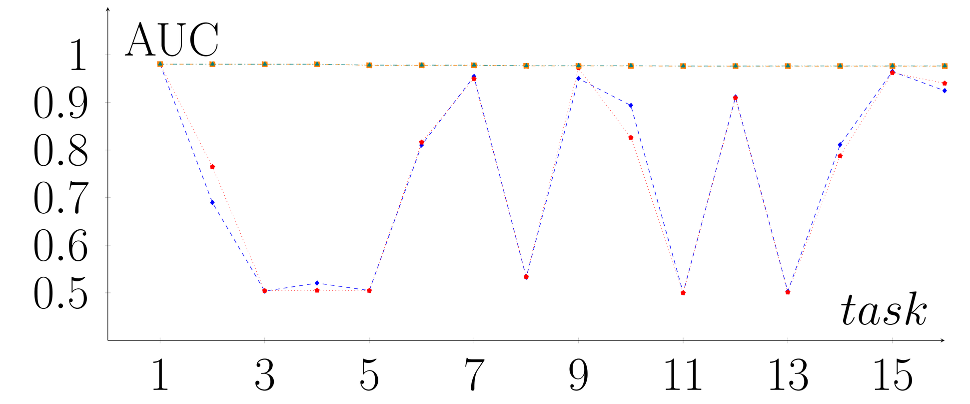
- Results are averaged over 10 runs
- Metric: Area under precision-recall curve (AUC)

Continual learning tasks

MNIST



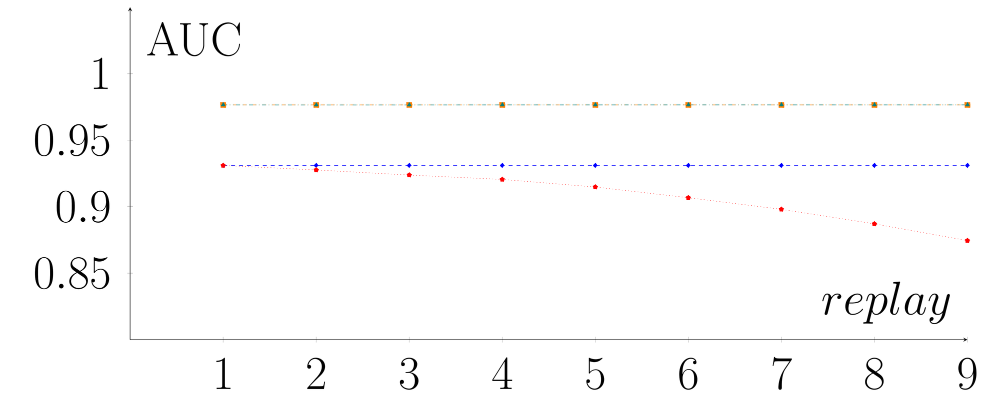
KDD Cup 99



LB · EWC · GR · UB

Study of degeneration effects

Degeneration



MB · MDG · KB · KDG

6. CONCLUSION

- VAE suffers from catastrophic forgetting
- We propose an effective method for continual learning of anomaly detection with VAE
- Evaluations indicate, that catastrophic forgetting can be mitigated
- Due to limited capacity of the VAE a degradation of performance can be observed during continued generative replay

ICASSP 2019, May 12-17, Brighton, UK