13 – 15 April, 2022

Robust Variational Autoencoders and Normalizing Flows for Unsupervised Network Anomaly Detection

Authors

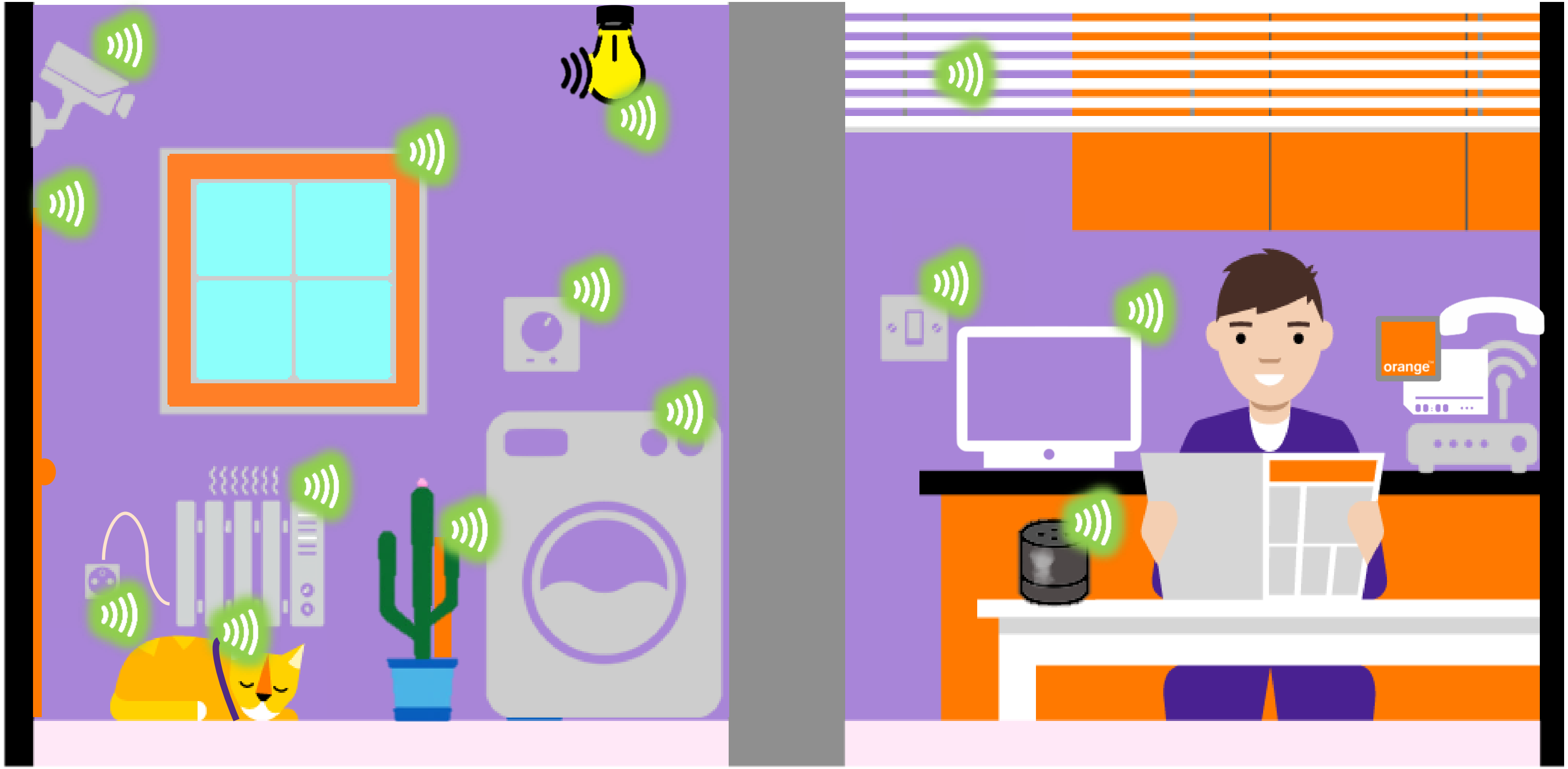
Naji NAJARI[1,2], Samuel BERLEMONT[1], Grégoire LEFEBVRE[1]

Stefan DUFFNER[2,3], Christophe GARCIA[2,3]

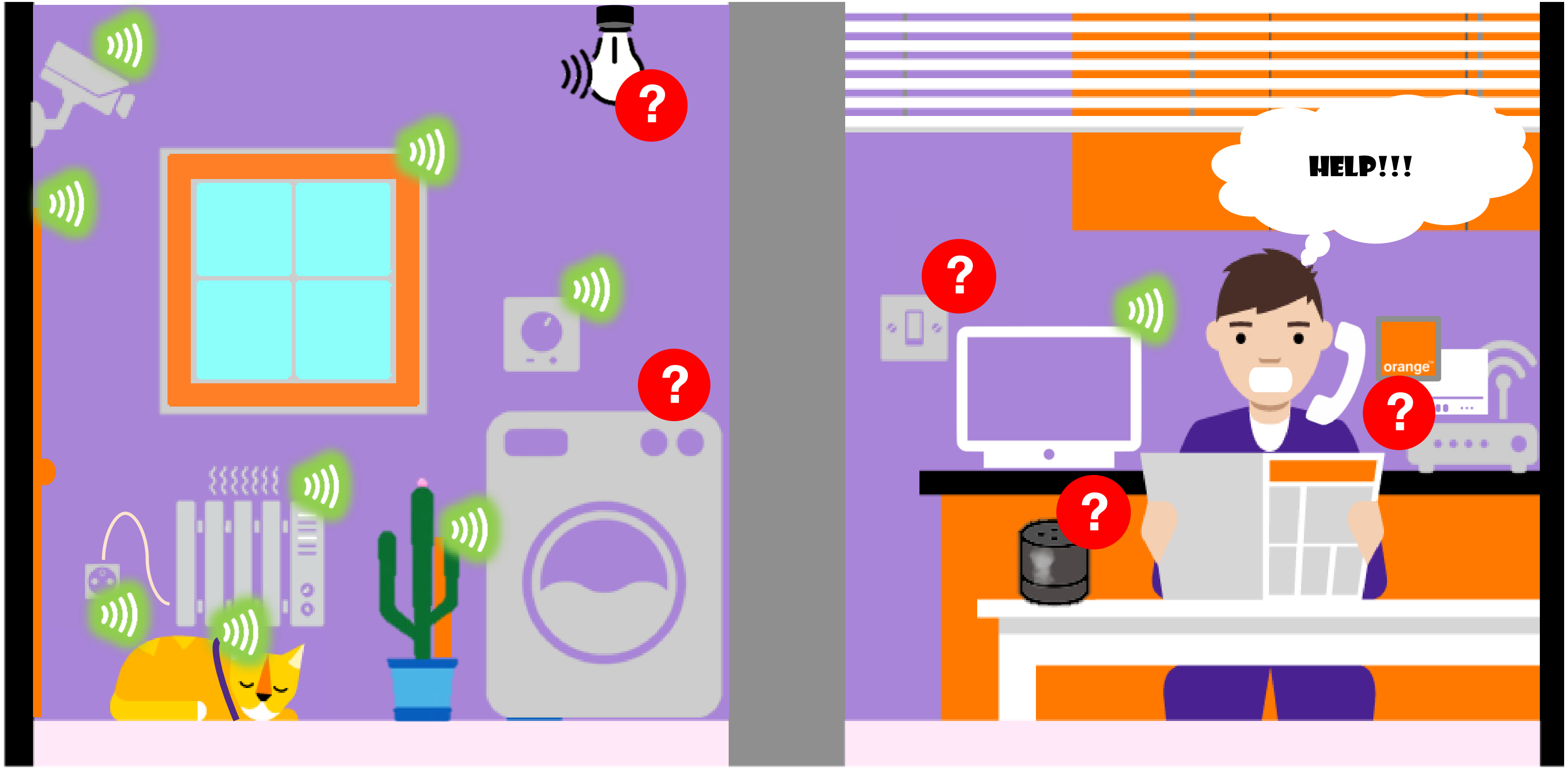
1 2 3

Outline

1. Context and Objective
Smart Home Device Management



1. Context and Objective
Smart Home Device Management



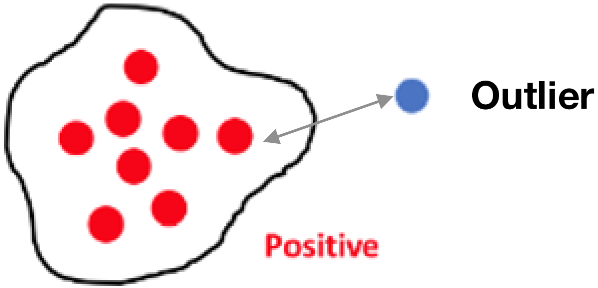
2. Related Work

Definition:

– Anomalies are patterns in data that do not conform to a well-defined notion of normal behavior [1]

Classical anomaly detectors [2]**: 2 steps**

1. Models the normal expected network behavior
2. Anomalies are deviations of the current behavior from the previously built model



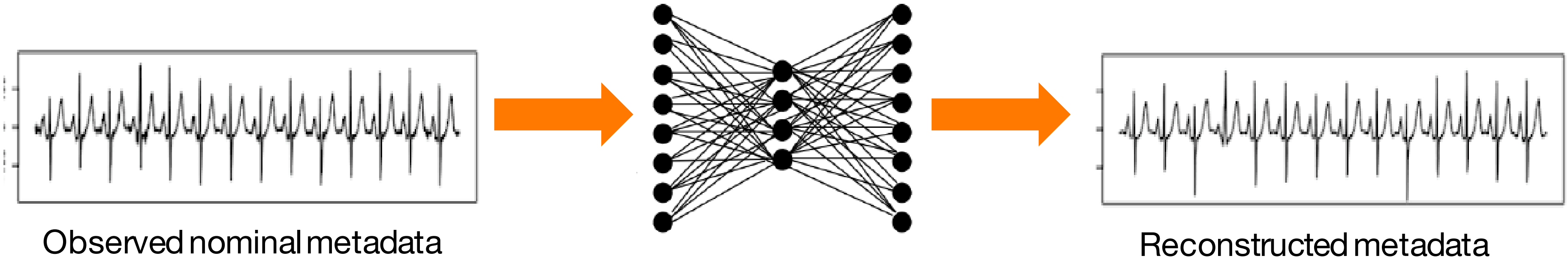
[1] Chandola, V., Banerjee, A. & Kumar, V., Anomaly Detection: A Survey. ACM Computing Surveys, 2009.

[2] Bulusu, Saikiran, Bhavya Kailkhura, Bo Li, Pramod K. Varshney and Dawn Xiaodong Song. “Anomalous Example Detection in Deep Learning: A Survey.” IEEE Access, 2020.

2. Related Work

Autoencoder-based anomaly detection:

– Training : train an autoencoder to reconstruct normal data [3]



– Minimize the energy function = maximize the log likelihood

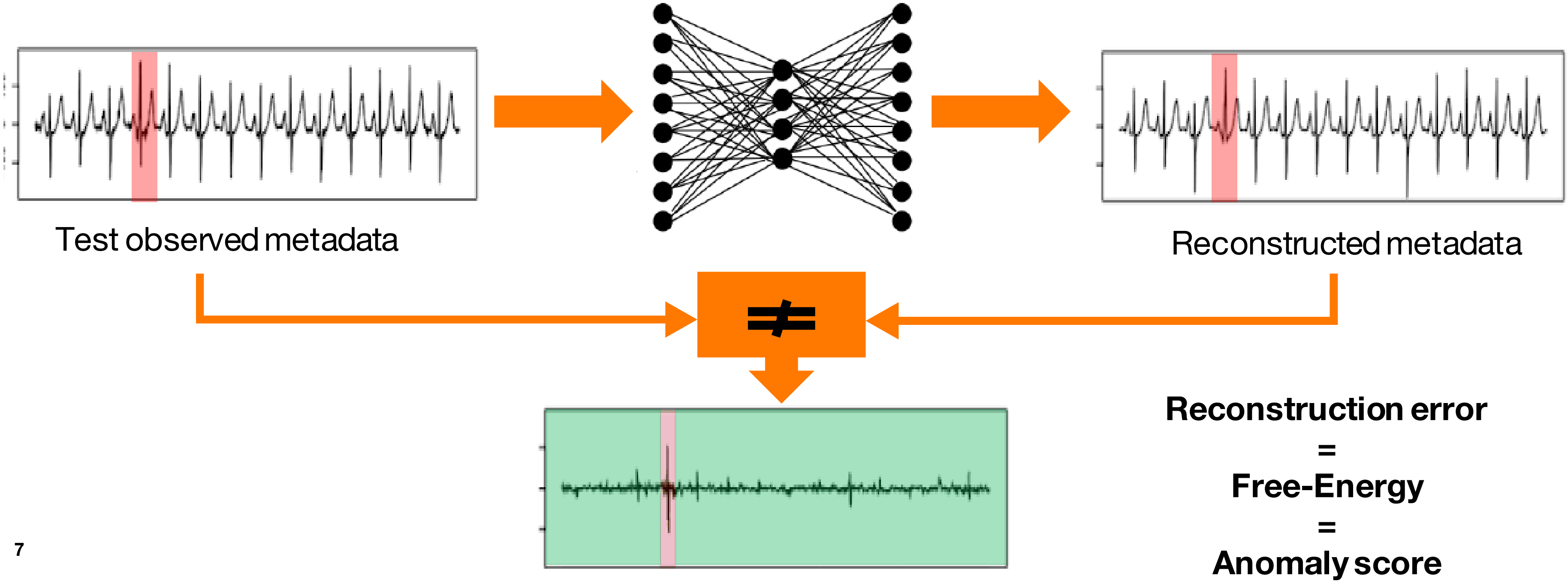
$$\log p_\theta(x) \geq \mathbb{E}_q[\log p_\theta(x|z)] - \mathbb{D}_{KL}[q_\phi(z|x)||p(z)] = -\mathcal{F}(x)$$

[3] Pol, Adrian Alan, Victor Berger, Cécile Germain, Gianluca Cerminara and Maurizio Pierini. "Anomaly Detection with Conditional Variational Autoencoders." 18th IEEE International Conference On Machine Learning And Applications (ICMLA), 2019.

2. Related Work

Autoencoder-based anomaly detection:

- Training : train an autoencoder to reconstruct normal data
- Testing : use the trained autoencoder to detect anomalies



Reconstruction error

=

Free-Energy

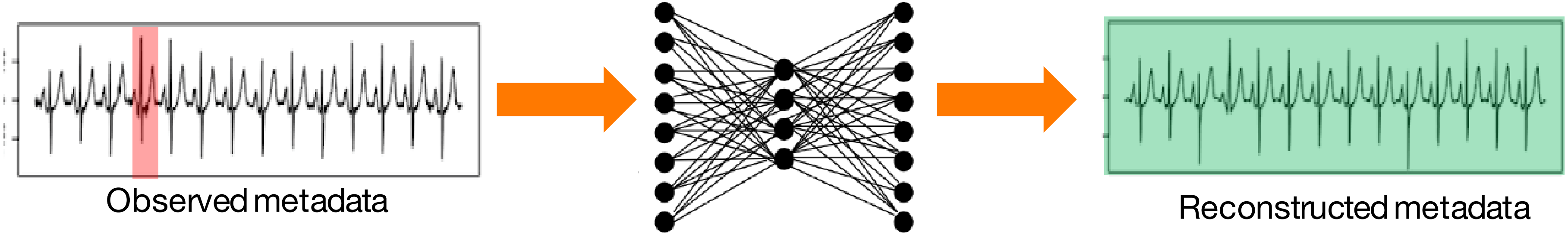
=

Anomaly score

2. Related Work

Limitations of existing approaches

- Strong assumption:
  - Training data are anomaly-free, **impossible in an IoT context [3]**
- Local training: data collection in the LAN
  - Anomalies may contaminate the training data
    - Data poisoning
    - Operational events: configuration errors, hardware failure, traffic congestion



[3] Wright, J., Ganesh, A., Rao, S., Peng, Y., & Ma, Y. Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization. In Advances in neural information processing systems, 2009.

3. Proposed Approach

Problem statement:

- Robust unsupervised anomaly detection [4]
  - The unlabeled training data contain both inliers and outliers (contaminants)
  - The **majority** of the training instances are nominal
  - The ratio of outliers is unknown in advance

Contribution:

- GRAnD, a **G**enerative **R**obust **An**omaly **D**etector that alternates between
  1. Filtering training anomalies
     - Extreme Value Theory (EVT)-based rejection strategy
  2. Learn a robust representation using a generative autoencoder

[4] Zhou, Chong and Randy Clinton Paffenroth. "Anomaly Detection with Robust Deep Autoencoders." Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017.

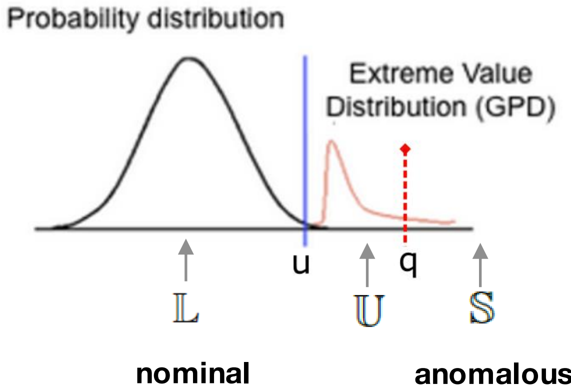
3. Proposed Approach
EVT-based rejection strategy

Problem statement:

– Early in the training phase, contaminants have larger free energy compared to inliers
  – Isolate these extreme values with the Peaks-Over-Threshold (POT) [5] approach
  – 2 hyperparameters to define: the initial threshold u, and the risk parameter q.

$$u = Q_3(F) + \alpha * IQR(F)$$
$$q = 10^{-3}$$

  – where,
    – Q_3: the third quartile
    – F: the free energy of training instances
    – IQR: the Inter-Quartile Range: $Q_3 - Q_1$
    – $\alpha = 1.5$
– We perform a sensitivity analysis w.r.t. hyperparameters



[5] Siffer, A., Fouque, P.A., Termier, A., Largouet, C.: Anomaly Detection in Streams with Extreme Value Theory. In: ACM SIGKDD, 2017.

3. Proposed Approach
Training loss

3 losses to optimize:

- 3 losses to optimize:

$$\mathcal{L}(x) = \mathbb{E}_{x\sim D_L}[\mathcal{F}_{\mathcal{L}}(x)] + |m - \mathbb{E}_{x\sim D_S}[\mathcal{F}_{\mathcal{S}}(x)]| + eCDF_m(\mathcal{F}_{\mathcal{U}}(x))\,|m - \mathbb{E}_{x\sim D_U}[\mathcal{F}_{\mathcal{U}}(x)]|$$

3. Proposed Approach
Training loss

3 losses to optimize:

- 3 losses to optimize:

$$\mathcal{L}(x) = \boxed{\mathbb{E}_{x\sim D_L}[\mathcal{F}_{\mathcal{L}}(x)]} + |m - \mathbb{E}_{x\sim D_S}[\mathcal{F}_{\mathcal{S}}(x)]| + eCDF_m(\mathcal{F}_{\mathcal{U}}(x))\,|m - \mathbb{E}_{x\sim D_U}[\mathcal{F}_{\mathcal{U}}(x)]|$$

Minimize the free energy function of L samples

$$\log p_\theta(x) \geq \mathbb{E}_q[\log p_\theta(x|z)] - \mathbb{D}_{KL}[q_\phi(z|x)||p(z)] = -\mathcal{F}(x)$$

3. Proposed Approach
Training loss

3 losses to optimize:

- 3 losses to optimize:

$$\mathcal{L}(x) = \boxed{\mathbb{E}_{x\sim D_L}[\mathcal{F}_{\mathcal{L}}(x)]} + \boxed{|m - \mathbb{E}_{x\sim D_S}[\mathcal{F}_{\mathcal{S}}(x)]|} + eCDF_m(\mathcal{F}_{\mathcal{U}}(x))\,|m - \mathbb{E}_{x\sim D_U}[\mathcal{F}_{\mathcal{U}}(x)]|$$

Minimize the free energy function of L samples

Maximize the free energy function of S samples

- |.| is the absolute distance and m is a margin
- we propose to fix an upper bound m, to prevent the loss from diverging

3. Proposed Approach
Training loss

3 losses to optimize:

– 3 losses to optimize:

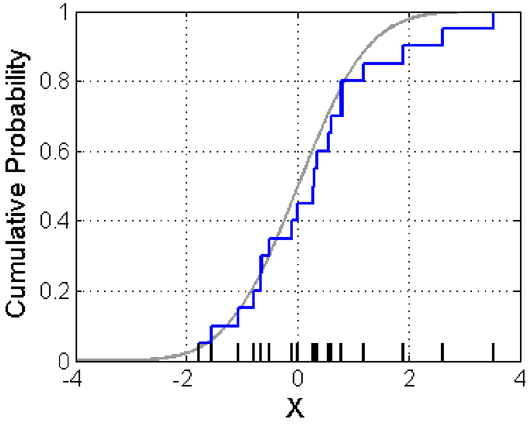
$$\mathcal{L}(x) = \boxed{\mathbb{E}_{x\sim D_L}[\mathcal{F}_{\mathcal{L}}(x)]} + \boxed{|m - \mathbb{E}_{x\sim D_S}[\mathcal{F}_{\mathcal{S}}(x)]|} + \boxed{eCDF_m(\mathcal{F}_{\mathcal{U}}(x))\,|m - \mathbb{E}_{x\sim D_U}[\mathcal{F}_{\mathcal{U}}(x)]|}$$

Minimize the free energy function of L samples

Maximize the free energy function of S samples

Maximize the free energy function of U samples

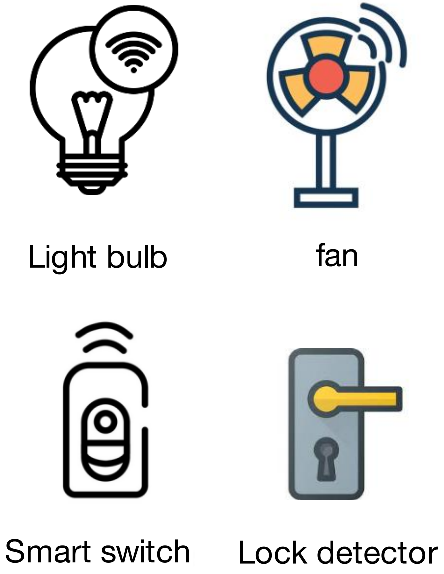
– Weighted with their anomalousness probability
  – to account for the uncertainty of these instances.
– computed with the empirical Cumulative Distribution Function (eCDF)



4. Experimental Results

Dataset: MedBIoT [3]

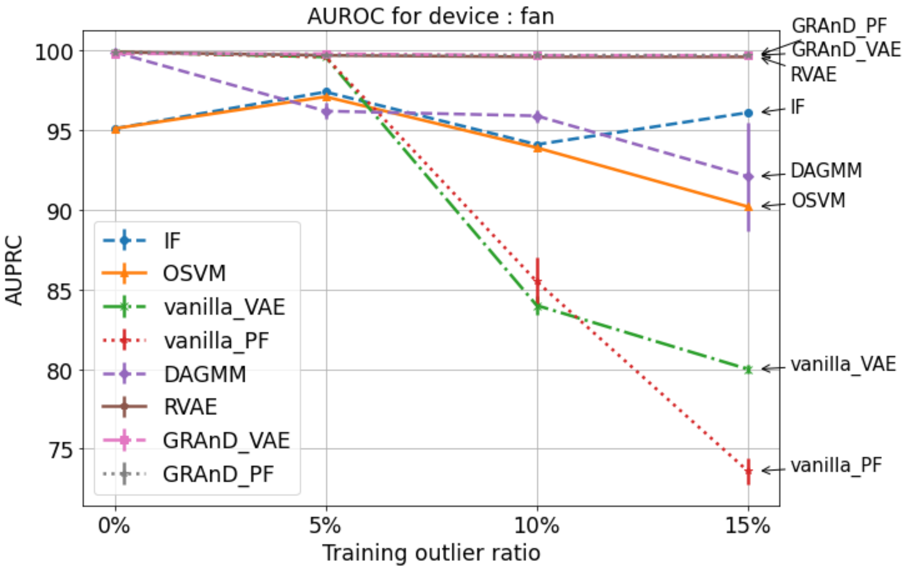
- 83 IoT devices
  - 4 families: fans, light bulbs, switches, lock detectors
- Three malwares: Mirai, Bashlite, Torii
  - ~17 million packets : 70% nominal and 30% anomalous
  - 61 metadata-based features
- Training
  - we vary the training anomaly percentage : 0%, 5%, 10%, 15%
  - Outliers are selected randomly from all training outliers
  - We train one model for each device family

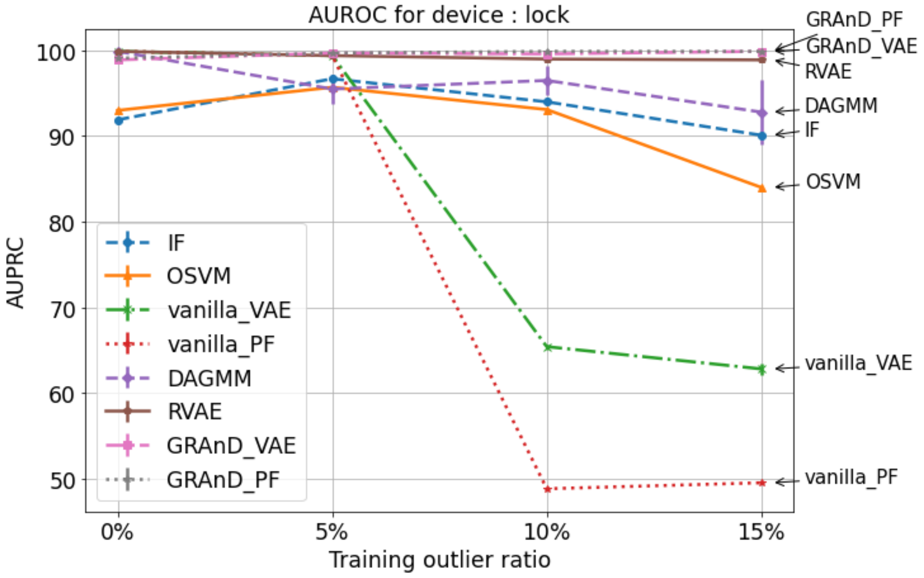


[6] Guerra-Manzanares, Alejandro, Jorge Medina-Galindo, Hayretdin Bahsi and Sven Nomm. “MedBIoT: Generation of an IoT Botnet Dataset in a Medium-sized IoT Network.” ICISSP, 2020.

4. Experimental Results

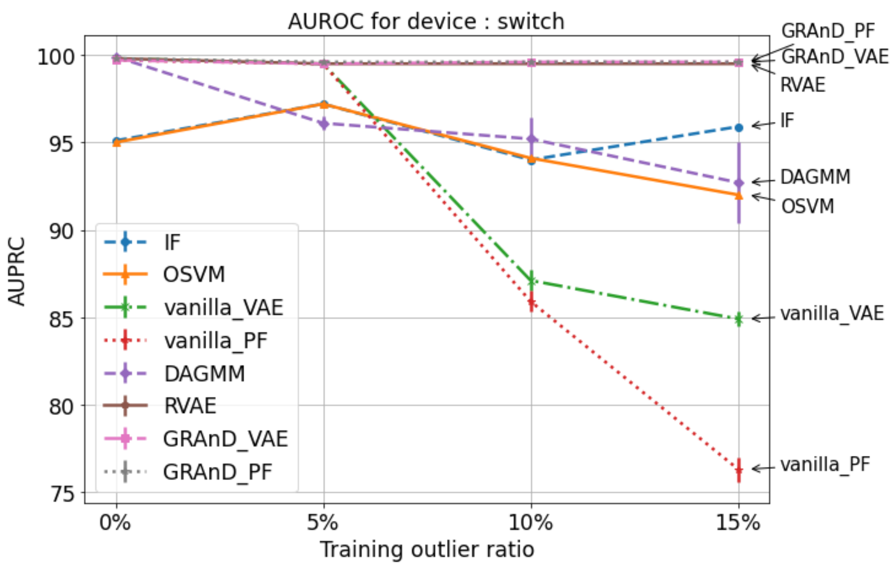
Results

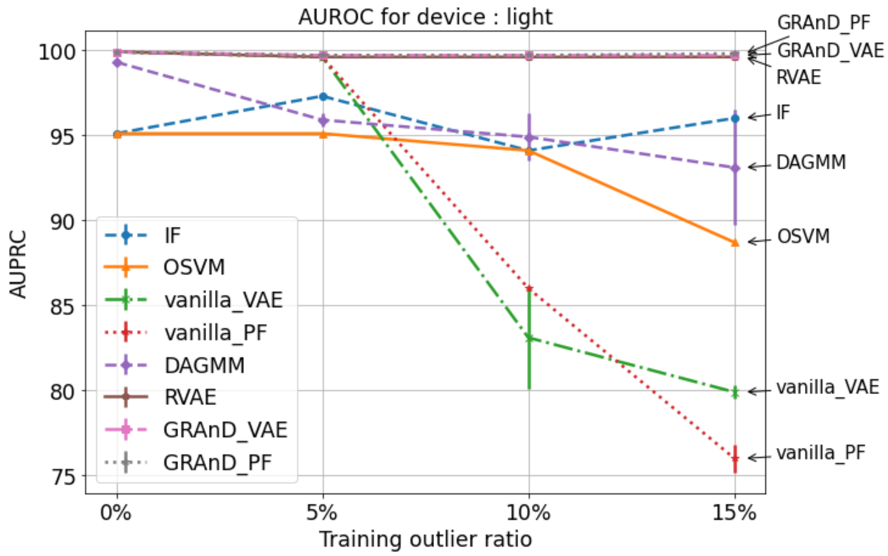




4. Experimental Results

Results



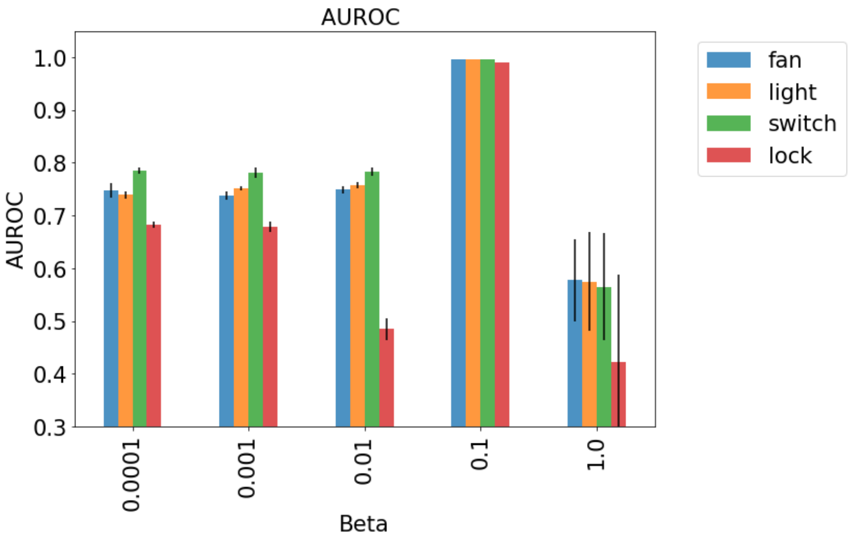


4. Experimental Results

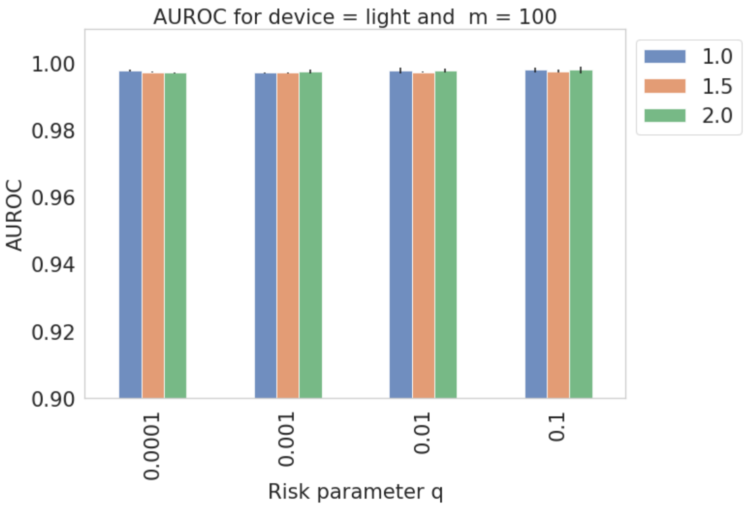
Sensitivity analysis with respect to hyperparameters

– Comparison between RVAE and GRAnD

RVAE



GRAnD



4. Conclusion And Future Work

Conclusion:

- GRAnD, a Generative and Robust Anomaly Detector
  - Rejection strategy : filters out outliers contaminating the data,
  - Joint training: learns a robust representation,
    - Inliers can be accurately reconstructed, while outlier reconstructions are corrupted.

Future work:

- Extend this approach to anomaly detection in time-series data
- Detect contextual and collective anomalies

Thank you

Naji NAJARI	naji.najari@orange.com
Samuel BERLEMONT	samuel.berlemont@orange.com
Grégoire LEFEBVRE	gregoire.lefebvre@orange.com
Stefan DUFFNER	stefan.duffner@liris.cnrs.fr
Christophe GARCIA	christophe.Garcia@liris.cnrs.fr

TABLE I: Extracted flow features using NFStream. See [35] for detailed feature descriptions.

Features			Abbreviations
src_port	src2dst_stdev_piat_ms	src2dst_duration_ms	src : source (e.g., src_port means the source port of the packet)
dst_port	src2dst_max_piat_ms	src2dst_packets	dst : destination
protocol	dst2src_min_piat_ms	src2dst_bytes	src2dst : traffic from source to destination
ip_version	dst2src_mean_piat_ms	bidirectional_min_piat_ms	piat : packet inter arrival time.
dst2src_stdev_piat_ms	dst2src_max_piat_ms	bidirectional_mean_piat_ms	stdev : standard deviation
bidirectional_duration_ms	bidirectional_syn_packets	bidirectional_stdev_piat_ms	ps : packet size
src2dst_mean_piat_ms	bidirectional_max_piat_ms	bidirectional_packets	
bidirectional_cwr_packets	bidirectional_bytes	bidirectional_ece_packets	
bidirectional_urg_packets	bidirectional_ack_packets	src2dst_syn_packets	
bidirectional_psh_packets	bidirectional_rst_packets	bidirectional_fin_packets	
dst2src_mean_ps	dst2src_stdev_ps	dst2src_max_ps	
src2dst_cwr_packets	dst2src_duration_ms	src2dst_ece_packets	
bidirectional_max_ps	src2dst_min_ps	src2dst_mean_ps	
dst2src_cwr_packets	dst2src_ece_packets	dst2src_urg_packets	
dst2src_syn_packets	src2dst_max_ps	dst2src_ack_packets	
dst2src_min_ps	dst2src_psh_packets	src2dst_stdev_ps	
dst2src_rst_packets	dst2src_fin_packets	src2dst_min_piat_ms	
dst2src_packets	src2dst_urg_packets	dst2src_bytes	
src2dst_ack_packets	bidirectional_min_ps	src2dst_psh_packets	
bidirectional_mean_ps	src2dst_rst_packets	bidirectional_stdev_ps	
src2dst_fin_packets			

4. Experimental Results

Dataset : NSL-KDD

- A benchmark dataset used to assess the performance Intrusion Detection Systems (IDS)
- Each instance of this dataset contains 41 features extracted from the network traffic
  - e.g., protocol type, TCP flags
  - One-hot encoding of categorical features + standardization of all features

- This dataset encompasses 39 types of attacks, with 17 not present in the training set.
- Assessment of the ratio of outliers in the training set to test robustness
  - we vary the training anomaly percentage : 0%, 5%, 10%, 15%
  - Outliers are selected randomly from all training outliers

- Architecture of the model
  - Symmetric autoencoder (Encoder layer size : 122, 8)

4. Experimental Results

Results