ROBUST AUDIO ANOMALY DETECTION

Wo Jae Lee[*†]
Purdue University

Karim Helwani[*], **Srikanth Tenneti, & Arvindh Krishnaswamy**
Amazon Web Services
`{helwk}@amazon.com`

ABSTRACT

We propose an outlier robust multivariate time series model which can be used for detecting previously unseen anomalous sounds based on noisy training data. The presented approach doesn't assume the presence of labeled anomalies in the training dataset and uses a novel deep neural network architecture to learn the temporal dynamics of the multivariate time series at multiple resolutions while being robust to contaminations in the training dataset. The temporal dynamics are modeled using recurrent layers augmented with attention mechanism. These recurrent layers are built on top of convolutional layers allowing the network to extract features at multiple resolutions. The output of the network is an outlier robust probability density function modeling the conditional probability of future samples given the time series history. State-of-the-art approaches using other multiresolution architectures are contrasted with our proposed approach. We validate our solution using publicly available machine sound datasets. We demonstrate the effectiveness of our approach in anomaly detection by comparing against several state-of-the-art models.

1 INTRODUCTION

The task of anomaly detection has many natural applications and has been studied within diverse research areas and application domains such as monitoring financial indicators, machine health, cloud resources, etc. (Ayed et al., 2020). For example, anomaly detection methods are applied to maximize system uptime and to improve operational performance in manufacturing plants by detecting abnormal behaviors (e.g., breakdown) of manufacturing equipment. A concise overview of anomaly detection approaches has been presented in Chandola et al. (2009). A large class of anomaly detection methods assume the existence of only normal data for training. Hence, the main challenge in designing a reliable anomaly detector is to find the best representation of the data which neither generalizes to unseen anomalies resulting in false negatives, nor overfits on existing data and hence, recognize falsely unseen normal data points as anomalies which results in alerting the user too often.

In a realistic setting, it is often challenging to obtain a clean training dataset and it is common to encounter outliers in the training dataset. For example, time series data collected by vibration sensors or microphones in an industrial environment is typically noisy as a result of dynamic interaction of multiple factors. Therefore, a mixture of, generally, non-stationary signals from multiple unknown sources is added to the signal of interest. The limitation of current anomaly detection methods is their lack of ability to deal with contaminated training datasets while capturing temporal dependencies across multiple time steps. In Nalisnick et al. (2019), the limitations of generative models on the anomaly detection / density estimation are discussed, and the necessity of the study on the robust to out-of-distribution inputs are emphasized. The present approach targets this limitation explicitly as it develops a robust multivariate time series anomaly detection model, in which we evaluate the probability of current samples given the time series history using a conditional probability density function learned from the training data. Three key characteristics of our method are: 1) The ability to model temporal dependency in the multivariate time-series data, 2) The ability to track the signal dynamics at multiple resolutions, and 3) Robustness to outliers and noisy training datasets. Our approach is based on an attention mechanism augmented recurrent neural network (RNN) architecture

---

[*]Equally contributing authors.
[†]Work conducted while the author was at Amazon Web Services.

which captures the temporal dependencies in the multivariate time series. Further, we propose a hierarchical structure to allow the extraction of global features (multiscale resolution), and finally, our approach models the time series innovation with an outliers robust probability distributions.

In prior works, anomaly detection methods have been discussed in settings where it is assumed that the input data is either independent and identically distributed or in explicit time series settings, where the temporal dynamic of the time series is explicitly modeled, e.g., using an autoregressive model as it is often the case in traditional change point detection methods (Gustafsson, 2000). In both mentioned cases, approaches to solve the anomaly detection problem are broadly either *Generative* or *Discriminative* (Wu et al., 2019).

In the generative approaches, a model is trained to generate samples of the data based on a certain latent representation. If the actual incoming measurement samples and the generated samples differ beyond a tolerable threshold, an anomalous data point is identified. Generative models include the popular autoencoder approaches (Zhou & Paffenroth, 2017; Zong et al., 2018; Gong et al., 2019) and the methods based on the recent neural density estimation (Papamakarios et al., 2017; Germain et al., 2015; Rezende & Mohamed, 2015) such as Yamaguchi et al. (2019); Jadidi et al. (2013); Giri et al. (2020a). Shortcomings of the density estimation based anomaly detectors have been discussed empirically in Nalisnick et al. (2019). They argued that a well-calibrated generative model is not able to assign higher probability to the training data than to some other out-of-distribution data. These shortcomings make the potential power of discriminative approaches apparent.

In the lack of anomalous data, the development of discriminative approaches employs the synthetic generation of anomalous data as perturbations on the normal data, e.g., Giri et al. (2020c). Clearly, the generation of synthetic anomalous data is problem specific and being able to generate relevant anomalous data points is often not a given. In this study, we argue that with generative models state-of-the-art anomaly detection performance can be achieved, if the target distribution is chosen to be outlier robust. Intuitively, a well performing anomaly detector should excel at modelling the boundaries of the normal data distribution and hence, it models the probability distribution of the data "tightly" which can be achieved by outlier robust modeling.

A popular representative of time series methods is the autoregressive model. In this model, the next sample is predicted as a linear function of previous samples while modeling the uncertainty of the model output as allowable *innovation* of the system (Kailath, 1968). For example, in a multivariate linear prediction setting, often a convolutive model is used where the innovation is modeled as

$$\mathbf{e}(t) = \mathbf{x}(t+1) - \mathbf{W}^T(t)\mathbf{X}(t), \tag{1}$$

where $\mathbf{X}(t) = \left[\mathbf{x}_1^T(t), \mathbf{x}_2^T(t), \cdots, \mathbf{x}_P^T(t)\right]^T$, $\mathbf{x}_p(t) = \left[x_p(t), x_p(t-1), \cdots, x_p(t-L+1)\right]^T$, $\mathbf{x}(t) = \left[x_1(t), x_2(t), \cdots, x_P(t)\right]^T$, $\{\cdot\}^T$ denotes transpose, L is the prediction context, p is the feature index, and t is the time index. The estimation of the model parameters can be done using least squares approach which implies a Gaussian innovation model.

Time series anomaly detection using recurrent architecture has been proposed in Marchi et al. (2015) with application to audio novelty detection where the auditory spectral features are processed by an autoencoder. More recently, a U-Net based architecture with convolutional LSTM layers has been proposed in Zhang et al. (2019). This approach is also temporally multiresolutional as it transforms the input into correlation matrices of multiple time lags. Methods for recurrent density estimation have been proposed in Oliva et al. (2017), where the power of gated-RNNs has been highlighted for this task as such models were able to scan through previously seen dimensions remembering and forgetting information as needed for conditional densities without making any strong Markovian assumptions. Similar architecture for forecasting has been presented in Salinas et al. (2020).

2 PREDICTIVE ROBUST PARAMETRIC DENSITY MODEL

An overview of the model architecture we are proposing is shown in Fig 1. The model aims at predicting a probability density function (pdf) of a next point given its past using a recurrent architecture where a conditional distribution is modeled as a function of past points using hidden state information $\mathbf{h}_t$ as follows

$$P\left(\mathbf{x}_t \mid \mathbf{x}_{t-1,\ldots,0}\right) \approx P\left(\mathbf{x}_t \mid f\left(\mathbf{x}_{t-1,\ldots,0}\right)\right) =: P\left(\mathbf{x}_t \mid \mathbf{h}_t\right).$$

To be robust to outliers during training, we include a parametric estimation of the conditional probability using a robust probability density function. Hampel (1971) gave a general formulation of distributional robustness in terms of a stability principle. Thereby, the deviations of a certain test statistic should be bounded for bounded deviations of the stochastic model. Consequently, a general approach to obtain an optimum robust estimator is based on the minimax principle (Huber & Ronchetti, 1981). Generally, the minimax approach aims at the least favorable situation for which it suggests the best solution. Hence, in our context, the basic underlying idea for a robust pdf for the next sample is to find a worst case pdf within the given class of models with a certain ϵ-deviation. As a suitable definition of the ϵ-deviation, a typically introduced model is the ϵ-contamination model

$$p(\mathbf{y}) = (1 - \epsilon)p_0(\mathbf{y}) + \epsilon p_{\text{outliers}}(\mathbf{y}), \tag{2}$$

where $p_0(\mathbf{y})$ denotes the distribution of the clean data, $p_{\text{outliers}}(\mathbf{y})$ the distribution of the outliers in the training data, and ϵ is the contamination parameter.

As shown in Huber & Ronchetti (1981), considering the maximum asymptotic variance of the estimates for all possible densities within the assumed ϵ-deviation yields a powerful approach to optimum outlier robustness.

Student t distribution can be less sensitive to outliers than Gaussian distribution by controlling the degree of freedom parameter (Peel & McLachlan, 2000; Svensén & Bishop, 2005; Wong et al., 2009).

In this paper, we will show how the Student t mixture model can be seen as a special case of the robust distribution framework presented. The parametric model we assume is

$$P(\mathbf{x}_t|\mathbf{x}_{t-1:0}; \boldsymbol{\theta}) = \sum_{i=1}^{c} \alpha_i f\left(\mathbf{x}_t|\mathbf{x}_{t-1:0}; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i, \nu_i\right), \tag{3}$$

where α_i denoting the responsibility, c the number of components, and

Figure 1: Architecture of the proposed anomaly detection system.

$$f(\mathbf{y}; \boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu)) = \frac{\Gamma((\nu + P)/2)}{\Gamma(\nu/2)\nu^{P/2}\pi^{P/2}|\boldsymbol{\Sigma}|^{1/2}} \cdot \left[1 + \frac{1}{\nu}(\mathbf{y} - \boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{y} - \boldsymbol{\mu})\right]^{-(\nu+P)/2},$$

where $\boldsymbol{\mu}$ representing the mean, $\boldsymbol{\Sigma}$ the covariance matrix, ν the degree of freedom, P the dimension, and $\Gamma(\cdot)$ is the Gamma function. It is known that the Student t distribution can be expressed as a scale mixture model (Murphy, 2012)

$$\mathcal{T}\left(\mathbf{x}_i \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu\right) = \int \mathcal{N}\left(\mathbf{x}_i \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}/z_i\right) \mathrm{Ga}\left(z_i \mid \frac{\nu}{2}, \frac{\nu}{2}\right) dz_i,$$

with $\mathrm{Ga}\left(z_i \mid \frac{\nu}{2}, \frac{\nu}{2}\right)$ being the gamma distribution with the shape and rate parameters being equal $\frac{\nu}{2}$. Splitting up the integral around the $z = 1$ offers a representation similar to Eq. (2). For the estimation of the parameters of the assumed model, we use a network inspired by the so-called WaveRNN (Kalchbrenner et al., 2018) and augment this architecture with an attention mechanism as a temporal regularizer as we describe in Appendix A, and a multiresolutional architecture employing convolutional layers described in Appendix B. These two architectural features help the network finding relevant patterns in long sequential data and allow the extraction of global features. The basic core operations of the network at a given resolution are as follows

$$\check{\mathbf{x}}_t = [\mathbf{x}_{t-1:0}; \mathbf{c}_t], \quad \mathbf{u}_t = \sigma\left(\mathbf{W}_u\mathbf{h}_{t-1} + \mathbf{U}_u\check{\mathbf{x}}_t\right), \quad \mathbf{r}_t = \sigma\left(\mathbf{W}_r\mathbf{h}_{t-1} + \mathbf{U}_r\check{\mathbf{x}}_t\right),$$

$$\tilde{\mathbf{h}}_t = \tanh\left(\mathbf{r}_t \circ \left(\mathbf{W}_h\mathbf{h}_{t-1}\right) + \mathbf{U}_h\check{\mathbf{x}}_t\right), \quad \mathbf{h}_t = \mathbf{u}_t \circ \mathbf{h}_{t-1} + (1 - \mathbf{u}_t) \circ \tilde{\mathbf{h}}_t,$$

$$\underline{\boldsymbol{\mu}}_t = \mathbf{W}_\mu \, \mathrm{relu}\left(\mathbf{W}_1\mathbf{h}_t\right), \quad \underline{\nu}_t = \sigma^+\left(\mathbf{W}_\nu \, \mathrm{relu}\left(\mathbf{W}_1\mathbf{h}_t\right)\right), \quad \underline{\alpha}_t = \mathrm{softmax}\left(\mathbf{W}_\alpha \, \mathrm{relu}\left(\mathbf{W}_1\mathbf{h}_t\right)\right),$$

$$\underline{\boldsymbol{\Sigma}}_{\mathrm{diag},t} = \mathrm{softplus}\left(\mathbf{W}_{2,\Sigma} \, \mathrm{relu}\left(\mathbf{W}_1\mathbf{h}_t\right)\right), \quad \underline{\boldsymbol{\Sigma}}_{\mathrm{lower},t} = \mathbf{W}_{1,\Sigma} \, \mathrm{relu}\left(\mathbf{W}_1\mathbf{h}_t\right),$$

where the $\mathbf{W}_{(\cdot)}$ and $\mathbf{U}_{(\cdot)}$ matrices are the GRU weights, $\mathbf{c}_t$ are conditioning parameters, e.g., features extracted as multiple resolutions, $\mathbf{u}_t$ is the update gate vector, $\mathbf{r}_t$ is the reset gate vector, $\circ$ denotes the Hadamard product, the underlined quantities denote a concatenation of the mixture
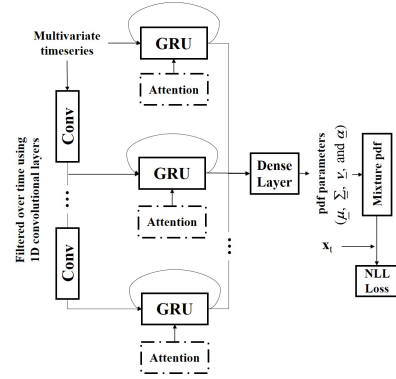
Table 1: Overall average performance of the proposed system (RSMM-MR) measured in AUC / pAUC and comparison to the baseline models.

METHOD		MIMII				ToyADMOS	
		FAN	PUMP	SLIDER	VALVE	TOY CONVEYOR	TOY CAR
SOLO MODELS	OC-SVM	0.6455 / 0.5824	0.6439 / 0.5679	0.5769 / 0.5029	0.4609 / 0.4970	0.5295 / 0.5087	0.6189 / 0.5318
	IF	0.6816 / 0.5374	0.6960 / 0.5623	0.5777 / 0.5029	0.4640 / 0.4952	0.5819 / 0.5125	0.6941 / 0.5429
	AE	0.7878 / 0.6466	0.7947 / 0.6280	0.6375 / 0.5190	0.5242 / 0.5067	0.8066 / 0.6432	0.7955 / 0.6204
	GMM	0.8257 / 0.6719	0.8684 / 0.6707	0.7533 / 0.5635	0.5678 / 0.5194	0.8243 / 0.6636	0.8017 / 0.6254
	MSCRED	0.5575 / 0.5367	0.4940 / 0.5148	0.4696 / 0.4901	0.4972 / 0.4984	0.4790 / 0.4845	0.4304 / 0.4984
	DAGMM	0.6438 / 0.5364	0.6996 / 0.5907	0.6492 / 0.5473	0.4729 / 0.5021	0.6223 / 0.5181	0.6730 / 0.5644
	DCASE BASELINE	0.8280 / 0.6579	0.8237 / 0.6411	0.7941 / 0.5906	0.5737 / 0.5079	0.8536 / 0.6495	0.8014 / 0.6617
	RGMM	0.8206 / 0.6504	0.8455 / 0.6590	0.9174 / 0.6909	0.8815 / 0.7263	0.8195 / 0.6497	0.8663 / 0.7277
	RGMM-MR	0.8264 / 0.6479	0.8702 / 0.6646	0.9370 / 0.7429	0.9572 / 0.8615	0.8660 / 0.6769	0.8812 / 0.7410
	RSMM	0.8560 / 0.6804	0.8571 / 0.6564	0.9358 / 0.7788	0.8672 / 0.7041	0.8713 / 0.6832	0.8805 / 0.7501
	RSMM-MR-W/O ATTENTION	0.8783 / 0.7013	0.8921 / 0.7082	0.9531 / 0.8546	0.9531 / 0.8056	0.9036 / 0.7168	0.8770 / 0.7518
	RSMM-MR	*0.8797 / 0.7054*	*0.9096 / 0.7353*	*0.9819 / 0.9052*	*0.9752 / 0.8873*	*0.9316 / 0.7875*	*0.9198 / 0.8112*
ENSEMBLE MODELS	DCASE WINNER	0.9454 / 0.8430	0.9365 / 0.8173	0.9763 / 0.8973	0.9613 / 0.9089	0.9119 / 0.7334	0.9434 / 0.8973
	RSMM-MR WITH DCASE WINNER	0.9891 / 0.9503	0.9590 / 0.8817	0.9979 / 0.9928	0.9849 / 0.9554	0.9316 / 0.7876	0.9583 / 0.9262

components parameters, $\mathbf{\Sigma}_{\text{lower}}$ denotes the components in the lower triangular part of the covariance matrix, $\mathbf{\Sigma}_{\text{diag}}$ stands for the diagonal components of the covariance matrix, σ is the sigmoid function, σ^+ is a scale sigmoid function, and the matrices $\mathbf{W}_{\nu,\{1,2\}\Sigma,\mu,\alpha}$ are matrices with block zero matrices selecting the respective output channels of the last layer. Here, we omit the biases for clarity. During the training phase, negative log likelihood (NLL) is adopted as a loss function, which can be written as follows

$$L(\mathbf{x}_t|\mathbf{x}_{t-1:0}) = -\log(P(\mathbf{x}_t|\mathbf{x}_{t-1:0};\boldsymbol{\theta})) = -\log\left(\sum_{i=1}^{c}\alpha_i f\left(\mathbf{x}_t|\mathbf{x}_{t-1:0};\boldsymbol{\mu}_i,\mathbf{\Sigma}_i,\nu_i\right)\right), \quad (4)$$

where $f(\cdot)$ is the assumed innovation distribution as discussed above.

3 EXPERIMENTAL RESULTS

The datasets used for evaluating the presented approach are the same used for DCASE challenge task 2 (DCASE, 2020). The desciptions of the dataset, data preprocessing, and evaluation metrics can be found in Appendix C. We compare the performance of the proposed model with eight baseline models including the DCASE winner (Giri et al., 2020c), the DCASE baseline (Koizumi et al., 2020), two state-of-the-arts anomaly detection models (MSCRED (Zhang et al., 2019) and DAGMM (Zong et al., 2018)), and four traditional anomaly detection models. The traditional models we chose are One-class Support Vector Machine (OC-SVM), Isolation Forest (IF), Autoencoder (AE), and Gaussian Mixture Model (GMM). Each of the baseline models is described in detail in Appendix D. We also consider the four variants of the proposed model for ablation study with respect to multi-resolution, attention mechanism, and Student t mixture model. The four variants are named as follow: (1) RGMM: we don't implement convolutional layers and hence, exclude the multi-resolutional architecture and replace a mixture of multivariate Student t with a mixture of multivariate Gaussian distribution, (2) RGMM-MR: we implement a multiresolutional architecture but replace a mixture of multivariate Student t with a mixture of multivariate Gaussian distribution, (3) RSMM: we don't implement the convolutional layers but use a mixture of multivariate Student t to parameterize the target pdf, and (4) RSMM-MR-W/o Attention: we don't implement the temporal attention mechanism in the proposed model.

The details of the model implementation is presented in Appendix E. The results of the experiment are summarized in Table 1. Additional descriptions including the results on each machine IDs are also reported in Appendix F. RSMM-MR outperforms most of the baseline models, and shows the comparable results to the DCASE winner although the DCASE winner is an ensemble of multiple anomaly detectors. We further show the results that can be obtained by ensembling the presented RSMM-MR with the system described in Giri et al. (2020c) (RSMM-MR with DCASE Winner in Table 1) and show the achieved superior performance of the obtained ensemble. Further, we conducted ablation test with respect to multi-resolution, attention mechanism, and Student t mixture model. The results of the four variants demonstrate the benefit of the individual components in the proposed system. Quantitative ablation analysis is given in Appendix G. Further, an analysis on the robustness to synthetically added Gaussian noise bursts to training data is provided in Appendix H.

REFERENCES

Fadhel Ayed, Lorenzo Stella, Tim Januschowski, and Jan Gasthaus. Anomaly detection at scale: The case for deep distributional time series models. *arXiv preprint arXiv:2007.15541*, 2020.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3):1–58, 2009.

DCASE. Task 2 challenge description web page - unsupervised detection of anomalous sounds for machine condition monitoring, 2020. URL http://dcase.community/challenge2020/task-unsupervised-detection-/-of-anomalous-sounds.

Mathieu Germain, Karol Gregor, Iain Murray, and Hugo Larochelle. Made: Masked autoencoder for distribution estimation. pp. 881–889, 2015.

Ritwik Giri, Fangzhou Cheng, Karim Helwani, Srikanth V. Tenneti, Umut Isik, and Arvindh Krishnaswamy. Unsupervised anomalous sound detection using self-supervised classification and group masked autoencoder based density estimation for audio anomaly detection. *DCASE2020 Challenge*, July 2020a.

Ritwik Giri, Fangzhou Cheng, Karim Helwani, Srikanth V. Tenneti, Umut Isik, and Arvindh Krishnaswamy. Group masked auto-encoder based density estimation for audio anomaly detection. In *sumbitted to DCASE 2020 Workshop*, 2020b.

Ritwik Giri, Srikanth V. Tenneti, Karim Helwani, Fangzhou Cheng, Umut Isik, and Arvindh Krishnaswamy. Unsupervised anomalous sound detection using self-supervised classification and group masked autoencoder for density estimation. *DCASE2020 Challenge*, July 2020c.

Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1705–1714, 2019.

Adam Goodge, Bryan Hooi, See Kiong Ng, and Wee Siong Ng. Robustness of autoencoders for anomaly detection under adversarial impact. In Christian Bessiere (ed.), *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pp. 1244–1250. International Joint Conferences on Artificial Intelligence Organization, 7 2020. doi: 10.24963/ijcai.2020/173. URL https://doi.org/10.24963/ijcai.2020/173. Main track.

Fredrik Gustafsson. *Adaptive filtering and change detection*, volume 1. Citeseer, 2000.

Frank R Hampel. A general qualitative definition of robustness. *The Annals of Mathematical Statistics*, pp. 1887–1896, 1971.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. URL http://arxiv.org/abs/1512.03385.

Peter J Huber and E Ronchetti. Robust statistics john wiley & sons. *New York*, 1(1), 1981.

Z. Jadidi, V. Muthukkumarasamy, E. Sithirasenan, and M. Sheikhan. Flow-based anomaly detection using neural network optimized with gsa algorithm. pp. 76–81, 2013.

Thomas Kailath. An innovations approach to least-squares estimation–part i: Linear filtering in additive white noise. *IEEE transactions on automatic control*, 13(6):646–655, 1968.

Nal Kalchbrenner, Erich Elsen, Karen Simonyan, Seb Noury, Norman Casagrande, Edward Lockhart, Florian Stimberg, Aaron van den Oord, Sander Dieleman, and Koray Kavukcuoglu. Efficient neural audio synthesis. *arXiv preprint arXiv:1802.08435*, 2018.

Yuma Koizumi, Shoichiro Saito, Hisashi Uematsu, Noboru Harada, and Keisuke Imoto. Toyadmos: A dataset of miniature-machine operating sounds for anomalous sound detection. In *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 313–317. IEEE, 2019.

Yuma Koizumi, Yohei Kawaguchi, Keisuke Imoto, Toshiki Nakamura, Yuki Nikaido, Ryo Tanabe, Harsh Purohit, Kaori Suefusa, Takashi Endo, Masahiro Yasuda, and Noboru Harada. Description and discussion on DCASE2020 challenge task2: Unsupervised anomalous sound detection for machine condition monitoring. In *arXiv e-prints: 2006.05822*, pp. 1–4, June 2020.

Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. ICDM '08, pp. 413–422, USA, 2008. IEEE Computer Society. ISBN 9780769535029. doi: 10.1109/ICDM.2008.17. URL https://doi.org/10.1109/ICDM.2008.17.

Larry M Manevitz and Malik Yousef. One-class svms for document classification. *Journal of machine Learning research*, 2(Dec):139–154, 2001.

Erik Marchi, Fabio Vesperini, Felix Weninger, Florian Eyben, Stefano Squartini, and Björn Schuller. Non-linear prediction with lstm recurrent neural networks for acoustic novelty detection. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7. IEEE, 2015.

Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, volume 8, 2015.

Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Gorur, and Balaji Lakshminarayanan. Do deep generative models know what they don't know? In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=H1xwNhCcYm.

Junier B Oliva, Kumar Avinava Dubey, Barnabás Póczos, Eric Xing, and Jeff Schneider. Recurrent estimation of distributions. *arXiv preprint arXiv:1705.10750*, 2017.

George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density estimation. In *Advances in Neural Information Processing Systems*, pp. 2338–2347, 2017.

David Peel and Geoffrey J McLachlan. Robust mixture modelling using the t distribution. *Statistics and computing*, 10(4):339–348, 2000.

Harsh Purohit, Ryo Tanabe, Takeshi Ichige, Takashi Endo, Yuki Nikaido, Kaori Suefusa, and Yohei Kawaguchi. MIMII Dataset: Sound dataset for malfunctioning industrial machine investigation and inspection. In *(DCASE2019)*, pp. 209–213, November 2019.

Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. *arXiv preprint arXiv:1505.05770*, 2015.

David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. Deepar: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, 36(3):1181–1191, 2020.

Mark Sandler, Andrew G. Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. *CoRR*, abs/1801.04381, 2018. URL http://arxiv.org/abs/1801.04381.

Markus Svensén and Christopher M Bishop. Robust bayesian mixture modelling. *Neurocomputing*, 64:235–252, 2005.

CS Wong, WS Chan, and PL Kam. A student t-mixture autoregressive model with applications to heavy-tailed financial data. *Biometrika*, 96(3):751–760, 2009.

Ying Nian Wu, Ruiqi Gao, Tian Han, and Song-Chun Zhu. A tale of three probabilistic families: Discriminative, descriptive, and generative models. *Quarterly of Applied Mathematics*, 77(2): 423–465, 2019.

Masataka Yamaguchi, Yuma Koizumi, and Noboru Harada. Adaflow: Domain-adaptive density estimator with application to anomaly detection and unpaired cross-domain translation. pp. 3647–3651. IEEE, 2019.

Chuxu Zhang, Dongjin Song, Yuncong Chen, Xinyang Feng, Cristian Lumezanu, Wei Cheng, Jingchao Ni, Bo Zong, Haifeng Chen, and Nitesh V Chawla. A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 1409–1416, 2019.

Chong Zhou and Randy C Paffenroth. Anomaly detection with robust deep autoencoders. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 665–674, 2017.

Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In *International Conference on Learning Representations*, 2018.

A TEMPORAL ATTENTION MECHANISM

To help the network finding relevant pattern in long sequential data, we augment the recurrent layers with the attention mechanism presented in Bahdanau et al. (2014). In this mechanism, while processing a sequence of length L the hidden states of the RNN, $\mathbf{h}_t$, are augmented with normalized weights a.k.a. attention vector $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_L]$ which is calculated as follows

$$\mathbf{q}_l = \tanh\left(\mathbf{W}_a^l \mathbf{h}_t\right), \text{ and } \beta_l = \mathrm{softmax}\left(\mathbf{W}_q^l \mathbf{q}_l\right), \tag{5}$$

where $\mathbf{W}_a^l$ and $\mathbf{W}_q^l$ are learnable weights, $\mathbf{q}_l$ is a learned embedding for each of the hidden states, and $\tanh$ is a hyperbolic tangent activation function. Finally, this attention vector is applied to the states in a sequence as following to obtain a new state vector as an average of the hidden states weighted by $\boldsymbol{\beta}$ or attention vector

$$\mathbf{h} = \sum_{l=1}^{L} \beta_l \mathbf{h}_l. \tag{6}$$

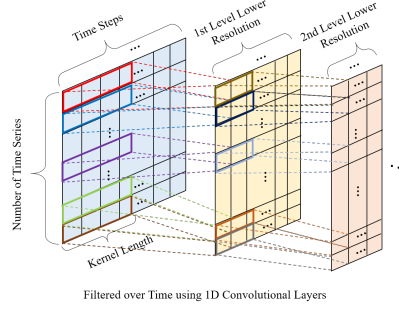## B MULTIRESOLUTIONAL ARCHITECTURE

Figure 2: Multiresolution features extraction over time

The network we propose in this paper is built in a hierarchical manner allowing the extraction of global features (Multiresolution). We achieve this by replicating the original network and feeding it with the inputs after passing them through a temporal filter. As shown in Fig. 2, 1D filters are applied to each time series to extract features over time. The extracted features are fed into GRU as the original network does. The final output layer is then conditioned on the output of these additional networks.

C DATASETS, DATA PREPROCESSING, AND EVALUATION METRICS

The dataset is organized in sound recordings where each recording is a single-channel 10-sec length audio which contains both a target machine's operating sound and environmental noise. The following six types of toy/real machines were used; Toy-car (`ToyADMOS`), Toy-conveyor (`ToyADMOS`), Valve (`MIMII`), Pump (`MIMII`), Fan (`MIMII`), Slide rail (`MIMII`) (DCASE, 2020; Koizumi et al., 2020; 2019; Purohit et al., 2019). There are three machine IDs in each machine type, so the total number of datasets used for training and evaluation is 18. The given task is to detect unknown anomalous sounds that are not observed during the training phase. This can be used to determine whether a condition of target machine is normal or anomalous (e.g., machine failure). For training, only recordings from machines operating under normal conditions are used. More detailed descriptions of the datasets (`ToyADMOS` and `MIMII`) and data reprocessing are as follow.

- `ToyADMOS` (Koizumi et al., 2019). The dataset is provided for audio based machine anomaly detection. The dataset consists of normal operating sounds and anomalous operating sounds. For training, only normal operating sounds are used. The anomalous machine operation sounds were collected by deliberately introducing a fault (e.g., bent shaft, deformation of gear, and excessive

tension) in a machine. To simulate actual factory environment, the sounds were randomly mixed with various kinds of environmental noise, which were collected from several places in a real factory. Multivariate time-series are extracted from the raw audio signal either as log Mel or MFCC features using the librosa python library package (McFee et al., 2015); log-Mel feature is used for Fan and Pump and MFCC feature is used for slider and valve. The number of Mel and MFCC is set to 90 (i.e., 90 dimensions), and the features are scaled to a range between -1 and 1.

- MIMII (Purohit et al., 2019). MIMII is the industrial machine sounds dataset that contains sounds of industrial equipment (e.g., valves, pumps, fans, and sliders). The sound of industrial equipment is recorded under normal and anomalous operating conditions. The malfunctions causing the anomalous sound included mechanical unbalanced and looseness in the industrial equipment. Various background noises, which were collected from multiple locations in a factory, were mixed with the recorded sounds to simulate a variety of noisy conditions. We extracted log Mel features from the raw audio signal, and the same preprocessing and scaling methods are used as in ToyADMOS.

For evaluation metrics, we use the same metrics as proposed by the organizing committee of the DCASE challenge (DCASE, 2020), which are Area Under the ROC Curve (AUC) and partial AUC (pAUC). The AUC and pAUC are defined as follows

$$\text{AUC} = \frac{1}{N_- N_+} \sum_{i=1}^{N_-} \sum_{j=1}^{N_+} \mathcal{H}\left(\mathcal{A}_\theta\left(x_j^+\right) - \mathcal{A}_\theta\left(x_i^-\right)\right)$$
$$\text{pAUC} = \frac{1}{\lfloor pN_- \rfloor N_+} \sum_{i=1}^{\lfloor pN_- \rfloor} \sum_{j=1}^{N_+} \mathcal{H}\left(\mathcal{A}_\theta\left(x_j^+\right) - \mathcal{A}_\theta\left(x_i^-\right)\right)$$

where $\lfloor \cdot \rfloor$ is the flooring function and $\mathcal{H}(x)$ returns 1 when $x > 0$ and 0 otherwise, and $\mathcal{A}_\theta$ denotes the model output given the parameters θ. Here, $\left\{x_i^-\right\}_{i=1}^{N_-}$ and $\left\{x_j^+\right\}_{j=1}^{N_+}$ are normal and anomalous test samples, respectively, and have been sorted so that their anomaly scores are in descending order. Here, N_- and N_+ are the number of normal and anomalous test samples, respectively. According to the above formulas, the anomaly scores of normal test samples are used as the threshold for a final decision whether a sample is anomalous or normal.

D BASELINE MODELS

OC-SVM. One-class support vector machine (OC-SVM) (Manevitz & Yousef, 2001) is an unsupervised algorithm learning a decision boundary for anomaly/novelty detection. In this study, we test several kernel types (e.g., linear, RBF, and polynomial), and we simply choose a kernel that renders the best performance. In our experiment, OC-SVM with RBF kernel shows the best performance.

IF. Isolation Forest (IF) (Liu et al., 2008) is a unsupervised anomaly detection algorithm, which explicitly isolates anomalies in the dataset using by means of decision trees. In the method, an ensemble of isolation trees is used to isolate anomalies. To train the model, several parameters (e.g., contamination, the number of trees in ensemble, and maximum number of samples to be drawn to train each base estimator) need to be set, so we simply conduct exhaustive search to find the optimal parameters, and following parameters are adopted: the number of trees = 150, max sample = 256, contamination = 0.05).

AE. Autoencoder (AE) learns a representation of the data, and it's trained to reconstruct an original input. After training, an error between input and output can used as an anomaly score. The method is based on the assumption that anomaly data cannot be effectively reconstructed. We used the same number of layers as done in the DCASE baseline in our AE model, and it is FC(90, 90, relu)-FC(90, 60, relu)-FC(60, 30, relu)-FC(30, 15, relu)-FC(15, 30, relu)-FC(30, 60, relu)-FC(60, 90, relu)-FC(90,90, none), where FC is a fully connected layer. In the training phase, following parameters are used: learning rate = 0.001, optimizer = adam, batch size = 64, regularization = L2, epoch = 50, and loss function = mean squared error (L2).

GMM. Gaussian Mixture Model (GMM) is a density estimation model to model a complex distribution using multiple Gaussian distributions. Two key parameters of GMM are the number of mixture components and covariance type (full, tied, diagonal, and spherical). We simply perform exhaustive search to find optimal parameters, and ten components with full covariance type are adopted in our study. An expectation–maximization (EM) algorithm is used to fit the model using the training dataset, and log-likelihood is used as an anomaly score during the inference.

MSCRED. (Zhang et al., 2019) Multi-Scale Convolutional Recurrent Encoder-Decoder (MSCRED) reconstructs multi-scale signature matrices, which encode the inter-variable correlations and the temporal dependency. A reconstruction error is used as an anomaly score. The publicly available code is used to implement the model. The size of signature matrices is modified according to the number of dimensions we analyzed. For training parameters, we borrow the wisdom from the original implementation.

DAGMM. (Zong et al., 2018) Deep Autoencoding Gaussian Mixture Model (DAGMM) is a density based model. Sample energy is used as an anomaly score. The publicly available code is used to implement the model. For training parameters, we borrow the wisdom from the original implementation.

DCASE winner. (Giri et al., 2020c) The system is an ensemble of self-supervised models using MobileNetV2, ResnetV2, and a multivariate density estimator GroupMADE. The model is trained for an auxiliary classification task which fits into the self-supervision framework. During the model training, several data augmentation techniques such as synthetic augmentation techniques and spectral warping augmentation were considered, more details can be found in the cited paper.

DCASE baseline. (Koizumi et al., 2020) The DCASE baseline is an autoencoder based model, which uses sample reconstruction error of the observed sound as an anomaly score. Details on the model can be found in the cited paper.

E   IMPLEMENTATION DETAILS

The architecture shown in Fig. 1 presents an overview of the proposed model. We implement the proposed model for the task of anomaly detection. We describe the implementation in four parts; (a) estimating the temporal dynamic of the time series, (b) multiresolutional feature extraction, (c) parameter estimation for a mixture of multivariate distributions, and (d) NLL loss calculation.

In (a), the multivariate time series, $\mathbf{x}_{t-1:0}$, are passed through the attention augmented recurrent layers. The recurrent architecture used in the model is based on gated recurrent unit (GRU), and the number of hidden layers, sequence length, and hidden size are set to 2, 70 (i.e., 70 frames), and 512. The attention vector is calculated as described in Appendix A.

(b) is a parallel process with (a). In (b), 1D-Convolutional layers (Conv) with kernel length = 10, stride = 3, and padding = 0 are used to extract lower resolution features as described in Appendix B. Each frequency bin (i.e., dimension of the time series is treated as univariate time series) is processed with a 1-D convolutional layer; as shown in Fig. 2, 1D filters are applied to each time series to extract features over time. The output of the convolutional layer is passed through GRU layers similar to (a). As in (a), the outputs of the GRU layer are augmented with an attention mechanism. Outputs from (a) and (b) are concatenated and fed into a fully connected layer for the pdf parameters estimation.

In (c), the parameters of a mixture of multivariate Student-t distribution, which are means vector, covariance matrices, degrees of freedom, and responsibilities, are estimated as described in Section 2. While the means vector and the lower triangular part of covariance matrices are simply estimated using fully connected layers, the specialized activation functions are additionally considered to impose constraints on the diagonal components of covariance matrices, the degrees of freedom, and the responsibilities. For example, the softplus activation function is placed after the fully connected layer to output only positive values for the diagonal components of the covariance matrix. Also, the scaled sigmoid activation function is used to set upper and lower bounds of the degree of freedom (or, to avoid an extremely large degrees of freedom, which would make Student t distribution similar to a Gaussian distribution). In this study, the lower and upper bounds of the degree of freedom are set between 1 and 10. The softmax activation is used to calculate the responsibilities of each mixture components, hence, these values are all positive and sum to 1. The output size of the network for each parameter is a function of the number of components in a mixture (c) and the dimensionality of the data (P). In our experiments, c is set to 3.

In (d), with the parameters estimated by the model in (c), a mixture model is obtained using Eq. (3), and an NLL loss is calculated for an observed value, $\boldsymbol{x}_t$, through Eq. (4).

The proposed model (RSMM-MR) and the four variants (RGMM, RGMM-MR, RSMM, RSMM-MR-W/o Attention) are trained with the training parameters of epoch = 30, batch size = 128, initialization = uniform with the magnitude of 0.1, optimizer = Adam, and learning rate = 0.00001. L2 regularization with weight decay parameter = 0.001 is chosen. The training and inference are done on a Tesla V100 GPU.

In the implementation of the ensemble of our model with the DCASE winner model (Giri et al., 2020c), we transform the anomaly scores of each model in the ensemble into a standardized scale, before combining them. The standardization transformation for any given model is applied in a per-machine manner where we calculate the mean and variance of its anomaly scores over the training data for that machine ID. The anomaly scores are then transformed to have zero mean and unit variance over the training data of that machine ID. Standardized anomaly scores across different models are then combined using mean or max ensembling.

F   DETAILED RESULTS OF ANOMALY DETECTION EXPERIMENT

For the anomaly detection task, we use the machine sounds datasets (`ToyADMOS` and `MIMII`). We present the average performance of each model in Table 1. Our proposed model, RSMM-MR, demonstrates the superior performance over the most of the baseline models, and shows the comparable results to the DCASE winner. The DCASE winner is an ensemble of MobileNetV2 (Sandler et al., 2018), ResnetV2 (He et al., 2015), and GroupMADE (Giri et al., 2020b), and also most top-ranked models in the challenge are ensembles of different anomaly detector. Among the traditional anomaly detection models, GMM, which is a density based model, performs the best in all settings in terms of AUC / pAUC, followed by AE, IF, and OC-SVM. While the GMM works well in many datasets, RSMM-MR outperforms in all datasets, especially improving AUC / pAUC by 0.2286 / 0.3417 and 0.4074 / 0.3679 in Slider and Valve, respectively. In our experiment, our approach outperforms MSCRED and DAGMM. In Zong et al. (2018), DAGMM was able to reduce the effect of contamination in trading data, but the anomaly detection accuracy gradually decreased with increasing contamination ratio. DAGMM performed similarly in benchmarks conducted in two recent anomaly detection papers (Goodge et al., 2020; Zhang et al., 2019). Also, ensembling the proposed model with the DCASE winner model further improves the results in all cases.

In Table 2, we report the results on each machine IDs (there are three machine IDs in each machine type). Results of the DCASE winner on each machine IDs were not reported by the authors, hence, we omit this model in Table 2. As it can be observed, our proposed model outperforms the baseline models consistently in most cases. Confirming that the model learns well the representation of the noisy training data (i.e., normal data) without either generalizing to unseen anomalies data nor overfitting on the training dataset.
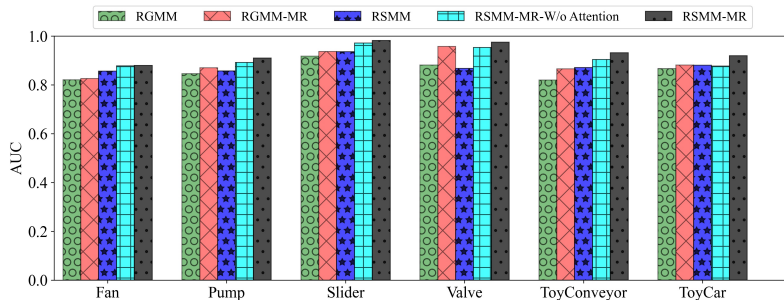
G   ABLATION ANALYSIS

Figure 3: Ablation analysis-AUC.

We conduct ablation analysis with respect to the different main features of our model to quantify the influence on the overall achieved model. To this end, four variations of the model are implemented,

Table 2: Performance of the proposed system on machine IDs measured in AUC / pAUC and comparison to the baseline models (bold numbers are the highest number on each ID)

| METHOD	ID	MIMII				ID	TOYADMOS TOY CONVEYOR	ID	TOYADMOS TOY CAR
		FAN	PUMP	SLIDER	VALVE				
OC_SVM	1	0.6087 / 0.5634	0.7824 / 0.6529	0.5969 / 0.5115	0.3957 / 0.4781	4	0.5192 / 0.4914	5	0.5894 / 0.5295
	3	0.6589 / 0.6253	0.6398 / 0.5021	0.5742 / 0.4920	0.4724 / 0.5132	5	0.5756 / 0.5307	6	0.6478 / 0.5318
	5	0.6688 / 0.5584	0.5094 / 0.5488	0.5597 / 0.5050	0.5147 / 0.4997	6	0.4937 / 0.5040	7	0.6195 / 0.5341
IF	1	0.6512 / 0.5032	0.7680 / 0.6284	0.6178 / 0.5044	0.4125 / 0.4886	4	0.6090 / 0.5188	5	0.6268 / 0.5257
	3	0.7220 / 0.5638	0.7028 / 0.4886	0.5825 / 0.4979	0.4812 / 0.5018	5	0.6138 / 0.5121	6	0.7232 / 0.5461
	5	0.6716 / 0.5453	0.6171 / 0.5700	0.5327 / 0.5062	0.4984 / 0.4951	6	0.5230 / 0.5066	7	0.7324 / 0.5568
AE	1	0.7167 / 0.5644	0.8147 / 0.6779	0.7129 / 0.5452	0.5391 / 0.5193	4	0.8832 / 0.7437	5	0.6744 / 0.5610
	3	0.7857 / 0.6263	0.7782 / 0.5813	0.6399 / 0.5169	0.5273 / 0.5127	5	0.8031 / 0.6000	6	0.8564 / 0.6513
	5	0.8611 / 0.7491	0.7911 / 0.6250	0.5596 / 0.4950	0.5063 / 0.4882	6	0.7335 / 0.5860	7	0.8556 / 0.6489
GMM	1	0.7524 / 0.5789	0.8868 / 0.7350	0.8519 / 0.6446	0.5673 / 0.5351	4	0.9088 / 0.7900	5	0.6873 / 0.5674
	3	0.8391 / 0.6498	0.8401 / 0.6297	0.7153 / 0.5293	0.6283 / 0.5298	5	0.8048 / 0.6141	6	0.8395 / 0.6255
	5	0.8857 / **0.7869**	0.8783 / 0.6475	0.6926 / 0.5166	0.5077 / 0.4934	6	0.7593 / 0.5868	7	0.8784 / 0.6833
MSCRED	1	0.5840 / 0.5260	0.5598 / 0.5544	0.4205 / 0.4867	0.4433 / 0.4750	4	0.4631 / 0.4761	5	0.4424 / 0.5001
	3	0.6159 / 0.5797	0.4461 / 0.5147	0.5556 / 0.4894	0.4673 / 0.5039	5	0.4905 / 0.4888	6	0.4160 / 0.4993
	5	0.4728 / 0.5045	0.4760 / 0.4752	0.4328 / 0.4944	0.5811 / 0.5163	6	0.4834 / 0.4885	7	0.4328 / 0.4959
DAGMM	1	0.5994 / 0.5111	0.7379 / 0.6252	0.8142 / 0.6212	0.4280 / 0.5009	4	0.5891 / 0.5143	5	0.5719 / 0.5280
	3	0.6406 / 0.5112	0.5858 / 0.5002	0.6394 / 0.5293	0.4964 / 0.5153	5	0.6045 / 0.5187	6	0.7111 / 0.5574
	5	0.6913 / 0.5870	0.7750 / 0.6466	0.4941 / 0.4914	0.4942 / 0.4900	6	0.6733 / 0.5212	7	0.7359 / 0.6076
DCASE BASELINE	1	0.7720 / 0.6127	0.8526 / 0.6953	0.9044 / 0.7202	0.5933 / 0.5175	4	0.9208 / 0.7226	5	0.7426 / 0.6316
	3	0.8561 / 0.6650	0.7944 / 0.6060	0.8196 / 0.5475	0.5657 / 0.5152	5	0.8335 / 0.6144	6	0.8338 / 0.6992
	5	0.8560 / 0.6961	0.8242 / 0.6220	0.6584 / 0.5040	0.5622 / 0.4911	6	0.8066 / 0.6115	7	0.8279 / 0.6543
RGMM	1	0.7813 / 0.5589	0.8722 / 0.7146	0.9392 / 0.8096	0.9978 / 0.9882	4	0.9197 / 0.7836	5	0.7518 / 0.6400
	3	0.8379 / 0.6560	0.8089 / 0.6060	0.9050 / 0.5769	0.8932 / 0.6605	5	0.7978 / 0.5889	6	0.9347 / 0.7914
	5	0.8425 / 0.7362	0.8554 / 0.6564	0.9079 / 0.6863	0.7536 / 0.5301	6	0.7409 / 0.5761	7	0.9123 / 0.7518
RGMM-MR	1	0.7721 / 0.5547	0.9011 / 0.7291	0.9571 / 0.8415	0.9998 / 0.9990	4	0.9245 / 0.7792	5	0.7685 / 0.6315
	3	0.8334 / 0.6376	0.8301 / 0.6134	0.9169 / 0.6443	0.9554 / 0.8425	5	0.8595 / 0.6139	6	0.9348 / 0.7869
	5	0.8738 / 0.7513	0.8794 / 0.6511	0.9370 / 0.7429	0.9165 / 0.7428	6	0.8141 / 0.6374	7	0.9402 / 0.8048
RSMM	1	0.7872 / 0.5659	0.8795 / 0.7064	0.9613 / 0.8746	0.9988 / 0.9934	4	0.9361 / 0.7954	5	0.7682 / 0.6459
	3	0.8755 / 0.6898	0.8011 / 0.6190	0.9573 / 0.8115	0.8659 / 0.6123	5	0.8799 / 0.6532	6	0.9318 / 0.7866
	5	**0.9053** / 0.7856	0.8909 / 0.6439	0.8888 / 0.6502	0.7370 / 0.5067	6	0.7978 / 0.6009	7	0.9414 / 0.8177
RSMM-MR-W/O ATTENTION	1	0.8294 / 0.6320	**0.9420 / 0.8412**	0.9881 / 0.9520	0.9997 / 0.9983	4	0.9185 / 0.7517	5	0.7732 / 0.6630
	3	0.9044 / **0.7456**	0.8632 / 0.6460	0.9718 / 0.8545	0.9505 / 0.7798	5	0.9148 / 0.7193	6	0.9361 / 0.7991
	5	0.9012 / 0.7264	0.8711 / 0.6373	0.9534 / 0.7572	0.9090 / 0.6387	6	0.8775 / 0.6793	7	0.9216 / 0.7934
RSMM-MR	1	**0.8387 / 0.6418**	0.9319 / 0.8221	**0.9928 / 0.9619**	**0.9998 / 0.9991**	4	**0.9595 / 0.8630**	5	**0.8336 / 0.6955**
	3	**0.9090 / 0.7378**	**0.8810 / 0.6814**	**0.9824 / 0.9092**	**0.9703 / 0.8640**	5	**0.9337 / 0.7520**	6	**0.9659 / 0.8774**
	5	0.8913 / 0.7365	**0.9160 / 0.7025**	**0.9704 / 0.8444**	**0.9555 / 0.7988**	6	**0.9017 / 0.7477**	7	**0.9598 / 0.8608**
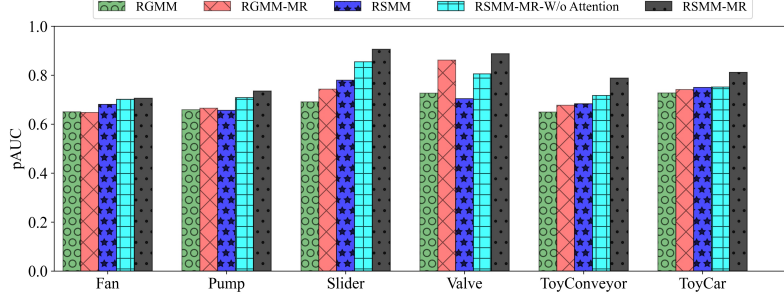
Figure 4: Ablation analysis-pAUC.

these are; RGMM (Recurrent GMM - standing for the model presented in this paper using a non-robust GMM model and without using CNN layers for the multiresolutional feature extraction), RGMM-MR (Recurrent GMM with multiresolutional feature extraction - standing for the model presented in this paper using a non-robust GMM model and using CNN layers for multiresolutional feature extraction), RSMM (recurrent Student-t mixture model), and RSMM-MR-W/o Attention. The performance metrics, AUC and pAUC, on each machine types are visualized in Figs. 3 and 4.

G.1 EFFECT OF CHOOSING A ROBUST PROBABILITY DENSITY DISTRIBUTION

To isolate the effect of choosing an outlier robust pdf in our modelling to the innovation process of the time series the RSMM and RSMM-MR, in which the multivariate Student t mixture model is used, are compared with RGMM and RGMM-MR, in which the multivariate Gaussian mixture model are employed.

With the same number of mixture components modelling the innovation process, the average of AUC and pAUC observed in RSMM is higher in all cases than those observed in RGMM, except

for Valve. The biggest improvement, 0.0184 (AUC) / 0.0879 (pAUC), is observed in Slider. When incorporating the multiresolutional architecture in the models, i.e., RSMM-MR and RGMM-MR, RSMM-MR shows the better performance than RGMM-MR in all cases. Especially, we observe a big improvement in pAUC; the average pAUC of all machine types is higher around 0.0839 in RSMM-MR than RGMM-MR. This empirical results demonstrate the effectiveness of the Student t mixture model in modelling robust parametric distribution.

To further highlight the effectiveness of the choice of outliers robust distribution to deal with noisy training data, we conduct an additional experiment on ToyCar (`ToyADMOS`), Slider (`MIMII`), and Valve (`MIMII`) where we synthetically introduce noise to the training data. The results are discussed in Appendix H.

G.2 EFFECT OF A MULTIRESOLUTIONAL ARCHITECTURE

We also consider the model with and without multiresolutional architecture described in Section A to demonstrate the effectiveness of extracting global features in the anomaly detection task. Enabling multiresolutional architecture in the model makes the final output conditioned on the output of multiresolutional time-series features. In all cases, the comparisons between RGMM and RGMM-MR or RSMM and RSMM-MR demonstrate that the multiresolutional architecture indeed helps to improve the performance. The averages of AUC and pAUC increase by 0.0348 and 0.0757 after incorporating multiresolutional architecture in RGMM and RSMM, respectively.

G.3 EFFECT OF A TEMPORAL ATTENTION MECHANISM

Lastly, we investigate the effect of the temporal attention mechanism described in Section A. As shown in Table 2 and Figs. 3 and 4, the results of RSMM-MR without the attention mechanism are consistently lower than RSMM-MR. This demonstrates that the recurrent layer augmented with attention mechanism is more effective to capture temporal dynamic in time series data.

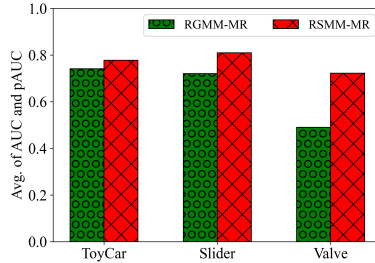## H ROBUSTNESS OF THE MODEL TO NOISY TRAINING DATA

Figure 5: Results of introducing Gaussian noise bursts to training data

To further investigate the robustness of the proposed model, we conduct one more experiment using ToyCar (`ToyADMOS`), Slider (`MIMII`), and Valve (`MIMII`). The effectiveness of an outliers robust distribution modeling the innovation process is highlighted by randomly introducing bursts of zero-mean Gaussian noise with variance $\sigma^2 = 5$ to 10% of the overall processing frames in spectrogram. The result is summarized in Fig. 5, and it shows the average values of AUC and pAUC of three IDs. As expected, the synthetically induced outliers negatively affect the detection accuracy for both models. However, the performance of our proposed robust model (RSMM-MR) is consistently higher than the non-robust model (RGMM-MR); the average values of AUC and pAUC are higher for Toycar, Slider, and Valve by 0.0366, 0.0986, and 0.2247.