National Research University
Higher School of Economics
Faculty of Computer Sciences

Final 2nd year research project

Structural breaks identification in sound signals

made by
Gladilin Egor Vyacheslavovich

supervised by
Lukyanchenko Pyotr Pavlovich

June 15, 2023

Contents

1 Introduction

Structural breaks identification is not a new challenge, which relates to many spheres: finance, econometrics, physics, medicine, climate forecasting. In statistics and econometrics structural breaks are known as unexpected changes over time in the parameters of regression models. There were invented many methods to identify them like Chow test or CUSUM test.
Within the epoch of AI development this challenge went further and now it is addressed to finding anomalies in videos, photos, text information, time series using neural networks and has other applications. With the usage of deep learning methods the were introduced multiple interesting solutions, which helps to overcome: unknown new data (as anomalies are distinct from the majority of data, model has to generalise known data efficiently and identify whether the difference exists), imbalance of classes (as anomalies occurs rarely), heterogeneity of different classes of objects (depending on the class the difference between anomaly and normal object can distinguish). Solving these issues, programmers faces low accuracy scores classifying data as abnormal/normal, lack of marked-up data, dealing with high-dimensional data, anomaly detection of a particular abnormal group, difficulty of results interpretation.
Sound processing is considered as a modern field in deep learning, however many approaches from CV and NLP are well adopted and widely used there. Hence, they are used to detect anomalies in weather changes (rain, wind, storm, hail), malfunctions in the operation of mechanisms, unusual road traffic, out-of-tune musical instrument, echolocation of ships and submarines, anomalies of air flows in air conditioning systems.
In this study anomaly detection in sound signals is explored. However, solutions are highly dependent on the given dataset, so it was decided to choose a particular dataset and explore models that were constructed specially for it. In the article I discuss the dataset, study the models and give the comparative analysis based on the metric's scores.

2 Dataset

First of all, before going to the models, one has to choose and describe the properties of the dataset. For this study, it was decided to unite 2 datasets: ToyADMOS [8] and MIMII [11]. Both dataset contains audio recordings of mechanisms' performance stored in "wav" files. There are 6 types of mechanisms: Toy-car (ToyADMOS), Toy-conveyor (ToyADMOS), Valve (MIMII), Pump (MIMII), Fan (MIMII), Slider (MIMII). For each type of mechanisms there were chosen several distinct machines, which id's are labeled on Figure 1. The united dataset consists of development dataset, additional trainig dataset and evaluation dataset.
Development dataset contains nearly 1000 normal sound recordings of each machine for training and around 200 recordings of each normal and abnormal to test.
Additional training dataset contains around 1000 normal recordings for each machine form evaluation dataset.
In evaluation dataset Machine IDs coincide with machine IDs from additional training dataset. This one is used only for estimating models and contains around 400 unlabeled recording for each machine.

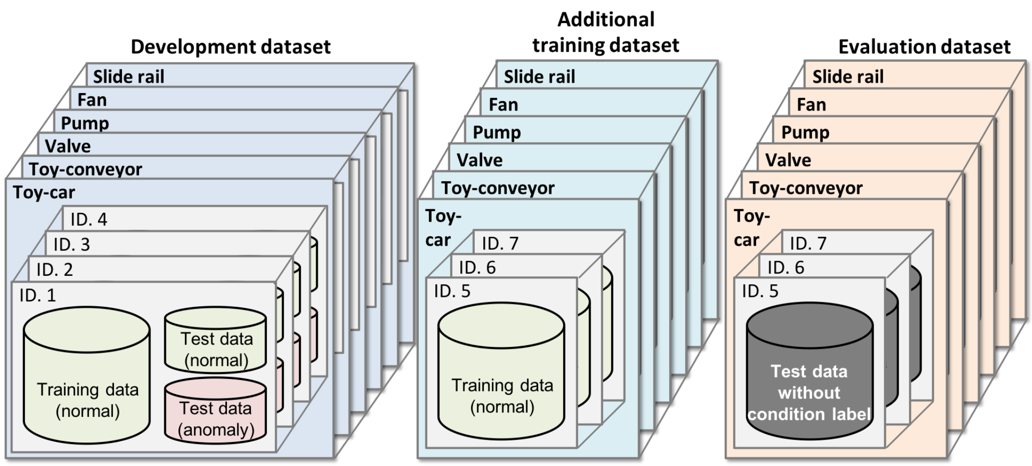


Figure 1: The dataset visualisation

3 Models

There already exists models that well-enough deal with anomaly detection for the dataset. In this section properties of the models and their metric's scores are discussed. The metrics for evaluating models' performance are Area Under the ROC Curve (AUC) and partial AUC (pAUC).

3.1 Group-MADE + MobileNetV2 ensemble

The model architecture and usage called Group-MADE was described in [4] and [3].

3.1.1 Data preprocessing

The authors propose to use log-Mel spectrograms as inputs construcred in the following way: each recording is split in 64ms frames having 32ms hop length between frames, then for each the log-Mel spectrogram is computed with 1024-FFT and 128 Mel bins and finally each 5 frames are concatenated resulting in $5 \times 128 = 640$ dimensional input.

3.1.2 How Group-MADE works

The core idea of MADE is to mask the connections between the layers of a vanilla autoencoder. Thus, the probability of a subsequent prediction depends on the previous ones and the result of the autoencoder is considered autoregressive for a specific sequence.

Thus, having autoregressive autoencoder, the probability density, $\mathbf{x}$ is calculated in the following way:

$$p(\mathbf{x}) = \prod_{d=1}^{D} p(x_d | x_{<d}) \tag{1}$$

Where $x_{<d} = [x_1, ..., x_{d-1}]^T$.

Defining $p(x_d = 1|x_{<d}) = \hat{x}_d$ and $p(x_d = 0|x_{<d}) = 1 - \hat{x}_d$, the loss-function becomes a negative log-likelihood:

$$\begin{aligned} -\log p(\mathbf{x}) &= \sum_{d=1}^{D} -\log p(x_d|x_{<d}) && \text{(2a)} \\ &= \sum_{d=1}^{D} -x_d \log p(x_d = 1|x_{<d}) - p(x_d = 0|x_{<d}) && \text{(2b)} \\ &= \ell(x) && \text{(2c)} \end{aligned}$$

Thus $\hat{x}_d$ is a function that takes $x_{<d}$ as an input and outputs $p(x_d)$. This allows us to refer to the autoregression, as each next probability depends only on previous object.
The authors constructed Group-MADE in such way that each further prediction depends on previous objects in a frame, which are inputs, instead of the whole probability distribution for each dimension.
Having an input $t = [t_{i+1}, t_{i+2}, t_{i+3}, t_{i+4}, t_{i+5}]$, where i^{th} frame is t_i, the probability of t_i is computed as following:

$$p(\mathbf{t}) = \prod_{i=1}^{5} p(t_i|t_{<i}) = \prod_{i=1}^{5} \prod_{j=128}^{5} p(t_{ij}|t_{<i}) \tag{3}$$

So, each Mel bin is dependent on the previous ones, but is not dependent on the other ones.
The autoencoder is constructed with fully connected layers with the following features: $[128, 128, 128, 128, 32, 128, 128, 128, 128]$
Negative log-likelihood was used as loss function and for estimating the normality for test samples. There is used Adam optimizer with learning rate set at 0.001 to train. As for training data, there were used all mechanism types without regard the machine IDs.

3.1.3 Classification

For classification task the authors decided to use self-supervised classification strategy. For each mechanism type, it was chosen to train MobileNetV2 [13] on normal data do not considering machine IDs to identify machine's ID and to distinguish real sample from synthetically generated versions.
Finally, there was added a softmax layer and negative softmax score on the test sample is considered to be the anomaly score.
The final solution represents the ensemble of Group-MADE and 2 MobileNetV2 with appended softmax layer.

3.2 Ensemble of Auto-Encoder Based and WaveNeT like Systems

The solution proposed by the authors is the ensemble consisting of several methods, whose weighted anomaly score is aimed to maximize AUC and pAUC metrics. These combined 3 methods are separately explained below.

3.2.1 Heteroskedastic Variational Auto-Encoder (HVAE)

For each mechanism type was constructed its own HVAE [1] , so HVAE is either trained on OpenL3 embeddings or on log-Mel spectrograms.
The key modification proposed by the authors is changing the distribution. In vanilla VAE [7] the loss function is

$$D_{KL}(\mathcal{N}(\mu, \sigma^2)||\mathcal{N}(0,1)) + MSE(X, \hat{X}) \tag{4}$$

Where D_{KL} stands for kl-divergence, $(\mu, \log(\sigma)) = E(X)$, E(X) - encoder's output, $\hat{X}$ - decoder's output (reconstruction).
So the probability distribution $p(x)$ f the feature space is approximated by maximizing:

$$\log(p(x)) >= E_{q(Z|X)} + D_{KL}(q(Z|X)||p(Z)) \tag{5}$$

$q(Z|X)$ is normal distribution with parameters set by encoder's output, $p(Z)$ is $\mathcal{N}(0,1)$ and $p(X|Z)$ is normal distribution with $(\mu, \sigma) = (\hat{X}, 1)$. So the modification considers changing $p(X|Z)$, instead of setting 1 for σ, authors propose $\hat{\sigma} = D(Z)$. And with this change, not only reconstruction is estimated, but also its precision. Thus the new loss function:

$$D_{KL}(\mathcal{N}(\mu, \sigma^2)||\mathcal{N}(0,1)) + \beta(wMSE(X, \hat{X}, \hat{\sigma}) - \sum_{i=1}^{n} \log(\hat{\sigma}_i)) \tag{6}$$

So the authors constructed several HVAE for each mechanism type, each HVAE has the same number of hidden layers. Estimation of $\hat{\sigma}$ is realised either by doubling an output layer, like in encoder, or by building separate decoder for variance, this approach is called Big Precision (BP). The precise architectures are described in the table in [1].

3.2.2 ID Conditioned Auto-Encoder (IDCAE)

The model uses basic principles of encoding and decoding previously described in article devoted to C2AE [10], however the input for decoder is changed. The authors propose firstly propose to encode machine IDs using one-hot approach, after that use some functions $H_\gamma, H_\beta : \mathbb{Y} \to \mathbb{Z}$, which take one-hot encoded label *ell* and map it to latent vector Z from $\mathbb{Z}$. Thus the decoder looks so: $D(H_\gamma(\ell) \times Z(E(X)) + H_\beta(\ell))$.
In IDCAE architecture DenseBlock implies combination of Dense, batchnomralization and relu. Having sound signals, they are constructed to Short Time Fourier Transform (STFT) with 1024-window and 512 hop length, then STFT is transfromed to power mel-spectrogram with M mels and take log

Encoder(E)	H_γ, H_β	Decoder(E)
Input(F,M)	Input one-hot IDs	Input 16
Flatten	Dense 16	DenseBlock 128
DenseBlock 128	Sigmoid	DenseBlock 128
DenseBlock 64	Dense 16	DenseBlock 128
DenseBlock 32		DenseBlock 128
DenseBlock 16		Dense F $\cdot$ M
		Reshape(F, M)

Table 1: IDCAE architecture

with base 10 and multiply by 10. The input shape is (F, M) where, F is frame size and M is number of mels. For training Adam optimizer with basic, parameters, exponential learning rate and 100 epochs are used.

3.2.3 FREAK

The Freak model contains 3 ResidualBlocks that were taken from WaveNet [14] and One CausalConv1D. One ResidualBlock contains one causal convolution, 2 independent convolutions, then results of these convolutions are multiplied and go to one more convolution layer.After each convolution there is normalization and activation function, except for the last layer. Causal convolution is implemented by doing element-wise multiplication with the convolution kernel. For 1D outputs are realised just by shifting forward convolutions by a few elements. The input is represented by computed STFT with 2048-window and 512 hop length, then it is transformed to mel-spectrograms with either 64 or 128 bins. Afterward logarithm is applied to each mel-spectrogram. Finally, mean and varianace based on the data are computed and standartization is applied. Adam is set to be optimizer with $\alpha = 0.001$, $\beta_1 = 0.85$, $\beta_2 = 0.999$, the batch size is equal to 32.

layer	channels	dilation	kernel	group
ResidualBlock	m*bins	1	3	4
ResidualBlock	m*bins	2	3	4
ResidualBlock	m*bins	4	3	4
CausalConv1D	bins	8	3	4

Table 2: FREAK architecture

3.3 Outlier-exposed classification

3.3.1 Data preprocessing

This method is based on the exposure of abnormal class. As training data contains only normal data, for classification we are lack of abnormal. The solution is to take into consideration that are neither belong to normal nor to abnormal classes - outliers. This allows to treat the problem as classification task and label outliers as abnormal objects.
After that the data transformation takes place. Firstly, audios are normalized to zero mean and unit variance. Then we calculate mel-spectrograms with sampling rate = 16000Hz, 1024 window, 512 hop length, 128 filters. Finally, by taking logarithm log-mel spectrograms are obtained.

3.3.2 ResNet for anomaly detection

The chosen model for classification is ResNet [5] which is widely used for classification tasks. Loss-function is Binary Cross Entropy (BCE), optimizer - Adam, having $\beta_1 = 0.9$, $\beta_2 = 0.999$, the batch size is equal to 64.

Layers	channels	Kernel size	ResidualBlock (RB)	Kernel size
Conv2d	5	5	Conv2d	ks1
BN			BN	
RB	5	(ks1=3, ks2=3)	Conv2d	ks2
MaxPool	5	2	BN	
RB	5	(ks1=3, ks2=3)	result + input	
MaxPool	5	2		
RB	5	(ks1=3, ks2=3)		
RB	5	(ks1=3, ks2=3)		
MaxPool	5	2		
RB	5	(ks1=1, ks2=1)		
RB	5	(ks1=1, ks2=1)		
RB	5	(ks1=1, ks2=1)		
Conv2d	1	1		
BN				
AvgPool				

Table 3: ResNet architecture

3.4 Ensemble of statistical methods

The authors [15] suggest to apply model based on geometric transformations for anomaly detection, however facing obstacles they add 2 statistical methods: scaling method [9] and probability aggregation method [2]. Authors assumes that the anomaly score of sub-model i denoted $s_i(x)$ is distributed by Gamma-distribution $s_i(x) \sim \Gamma(\alpha_i, \beta_i)$. For obtaining scaled scores one can use CDF $F_i(x; \alpha_i, \beta_i)$ over the training dataset distribution. [h!]

$$\hat{s}_i(x) = F_i(x; \alpha_i, \beta_i) \tag{7a}$$

$$= \frac{\gamma_i(\alpha_i, \beta_i, s_i(x))}{\Gamma(\alpha_i)} \tag{7b}$$

Ultimately, the anomaly score is obtained by calculating:

$$[h!]\hat{s}_e(x) = 1 - \prod_{i=1}^{n}(1 - \hat{(s)}_i(x)) \tag{8}$$

Where $\hat{s}_e(x)$ is the anomaly score and $\hat{s}_e(x)$ are scaled scores calculated in (7a), (7b).

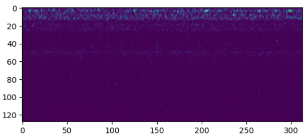
3.5 Analysis of existing solutions

Looking through approaches from above one can notice that ,due to complex data transformation the models are able to generalise data in efficient way, this increase robustness of the models when the inputs are normal and vise versa in case id data is abnormal.
On the other side authors tried to improve models and maximize score due to complex models and ensembles. Thus, we see that [1], besides augmentation, uses an ensemble of several auto-encoder based models and diverse classification models, [4] used an ensemble of auto-encoder, that takes into account sequences and IDs, and classification model. [15] used a statistical methods to solve the tasks, nevertheless authors applied an ensemble and succeeded as well.
Thus, our solution is based on complex data preprocessing to derive all valuable information from audio files and we dicede to use auto-encoder based model as previous results were quite successful.

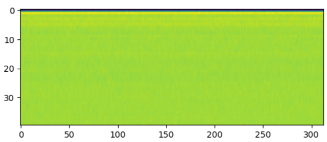
4 Proposed approach

4.1 Data preprocessing

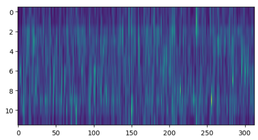
Analyzing previously proposed solutions, It found out that data preprocessing impacts significantly on the success. Log-mel specrograms are widely used as features, however, it is important to preserve all properties of a recording in its final representation, that is why it was decided to apply several sound representaions: mel-spectrogram, mel-frequency spectrum (MFCC), short-time Fourier transformation (STFT), spectral contrast and tonnetz.



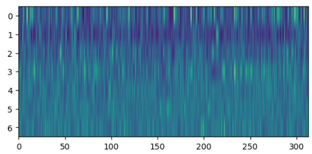
(a) Mel-spectrogram



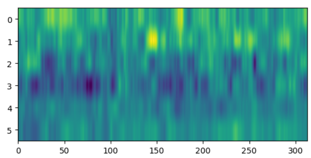
(b) MFCC



(c) Tasty coffee



(d) Spectral contrast



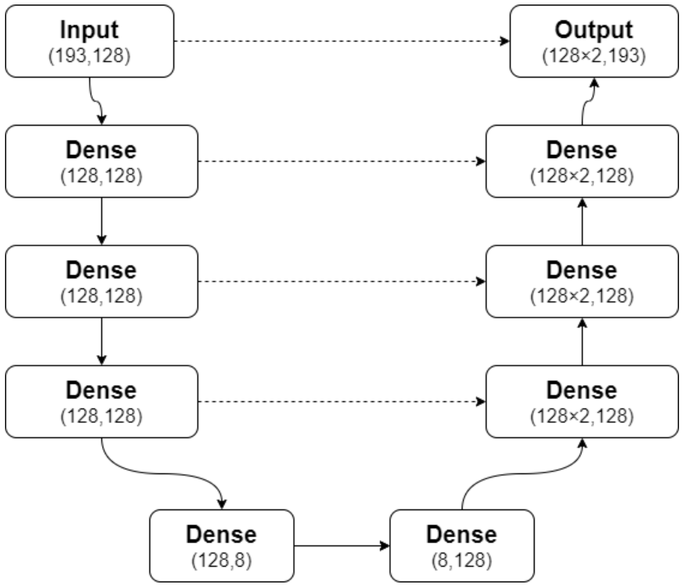
(e) Tonnetz

Figure 2: Sound representations used.

Hence, mel-spectrogram allows us to understand the volume of the signal or the amplitude when the signal changes over time at different frequencies. MFCC describes the timbral features of the audio and helps to identify the unique features of sound. STFT decomposes the sound into sinusoidal waves of different amplitude and frequency. Spectral contrast "represents the relative spectral distribution instead of average spectral envelope. Spectral Contrast deals with the strength of spectral peaks, valleys, and their difference separately in each sub-band, and represents the relative spectral characteristics. Octave-based Spectral Contrast feature has a better discrimination among different music types than MFCC" [6]. Ultimately, tonnetz is a 2-dimensional mesh which maps the tonal landscape of sound. Thus, we compute Mel-spetrogram with 128 mel bins, MFCC with 40 coefficients, STFT with frame with 12, spectral contrast with number of frequency bands is equal to 7 and numer of chroma bins in tonnetz is equal to 6.
Following that we compute final feature by taking an average for each sound representation and concatenating it into a $(1, 193)$-dimensional vector.

4.2 FC UNet

Having seen that auto-encoder performs efficiently resolving this task, it was decided to apply auto-encoder based model. In particular, we choose UNet architecture [12], however as we work with vectors of (1,193) dimensions it is reasonable to change convolutional layers to linear ones. UNet is a good choice, despite that it was initially used for segmentation problems, it has nice reconstruction properties that are valuable for our solution. I applied



batch normalization and Relu as activation fanfiction for each layer, except for the first and last ones. Also there is a concatenation (dotted lines) of features of encoder layers and previous decoder layers. The reason to use UNet architecture is well known generalisation property. I used Adam optimizer with lr = 10e-3, MSE as loss function and Reduce Lr on Plateau as lr scheduler with minimum lr = 10e-4, factor = 0.5, patience = 30 . I assumed that after training on normal data, calculated MSE for abnormal and reconstructed features should be rather bigger than MSE for normal and reconstructed. I trained this model for each machine.

After training the model we have to somehow classify anomalous data. We didnt use a specific model for this and assumed the following: after training model on normal data, the MSE loss between normal features and reconstructed ones has to be rather smaller than MSE between abnormal features and reconstructed ones, thus we consider anomaly score to be MSE loss between input and reconstruction. After computation MSE for each input, we find the threshold and if it is exceeded we classify data as abnormal and normal otherwise.

4.3 Results

IDs	Fan	Pump	Slider	Valve	ToyCar	Conveyor
1	0.6888	0.9545	0.9705	0.8253	0.9511	0.9213
2	0.9301	0.99	0.9721	0.805	0.9772	0.8074
3	0.931	1	0.9243	0.684	0.8913	0.9679
4	0.9794	0.8872	0.7869	0.685	0.9801	

Table 4: AUC score for each machine

Here are presented the results of our approach. We see that fully-connected UNet is specially efficient for pump toycar and toyconveyor, we see that for 16 out of 23 machines AUC score is higher than 0.88 and for 14 out of 23 machines AUC score is higher than 0.92. However, model demonstrated unexpectedly low scores for valve, this might happen because, the amount of train data for valve is significantly less than for other types of mechanisms. Overall, we received average AUC score over all types is equal to **0.8921**.

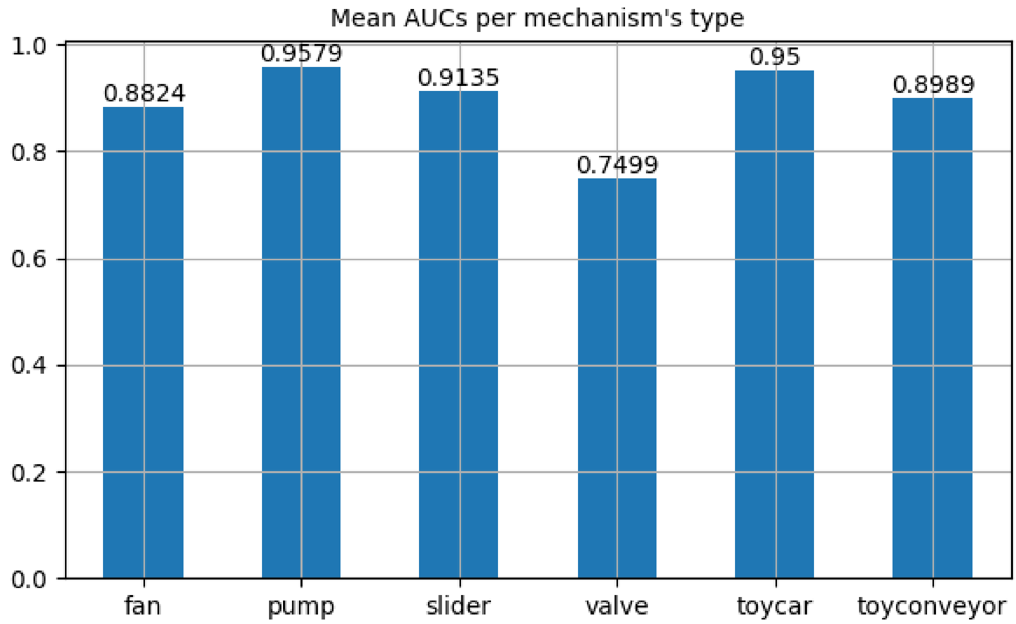


Figure 3: The dataset visualisation

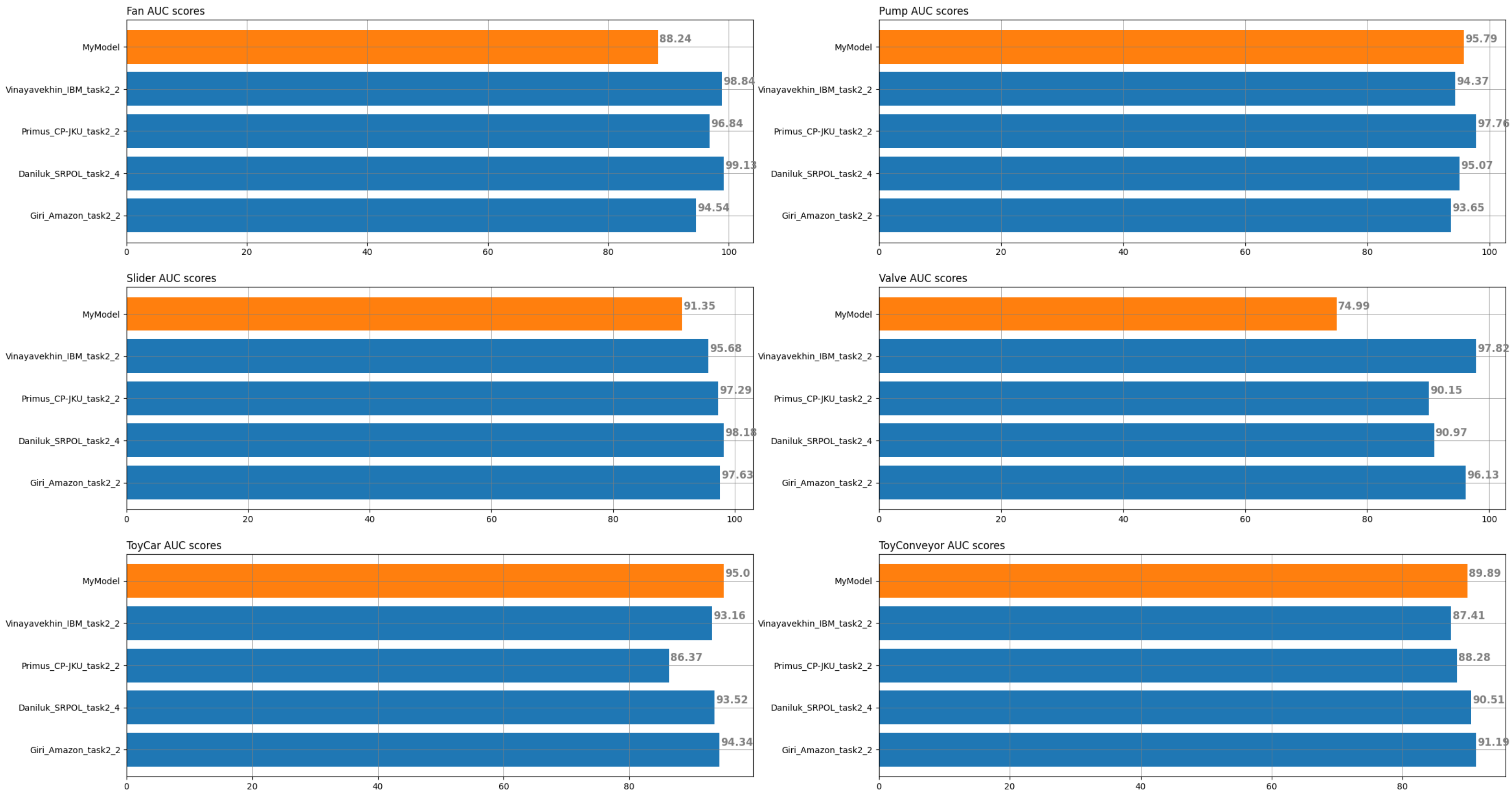


Figure 4: The dataset visualisation

5 Conclusion

To sum up the results of our work, we did detailed review of exiting solutions for solving task of sound anomaly detection in mechanisms, invented our own solution and demonstrated it. Actually, it performs efficiently and outperforms existing ones in several cases as it is seen on the bar charts above.
The future improvements might relate to both data and model. We can use more complex model for better finding general patterns of normal data, hence, better distinguishing normal and abnormal data. In particular one can apply variational modifications to UNet or use another architectures for encoder and decoder. As for data we can apply data augmentation to increase amount of training data notably amount for those types of mechanisms which demonstrated the lowest AUC score, which also might impact final results.
We will keep developing sound anomaly detection systems, improve algorithms for more precise detection and bigger number of mechanisms and apply our results in real life.

References

[1] Pawel Daniluk, Marcin Gozdziewski, Slawomir Kapka, and Michal Kosmider. Ensemble of auto-encoder based systems for anomaly detection. Technical report, DCASE2020 Challenge, July 2020.

[2] Jing Gao and Pang-ning Tan. Converting output scores from outlier detection algorithms into probability estimates. In *Sixth International Conference on Data Mining (ICDM'06)*, pages 212–221, 2006.

[3] Mathieu Germain, Karol Gregor, Iain Murray, and Hugo Larochelle. MADE: Masked Autoencoder for Distribution Estimation. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *JMLR Proceedings*, pages 881–889. JMLR.org, 2015.

[4] 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. Technical report, DCASE2020 Challenge, July 2020.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[6] Dan-Ning Jiang, Lie Lu, Hong-Jiang Zhang, Jian-Hua Tao, and Lian-Hong Cai. Music type classification by spectral contrast feature. In *Proceedings. IEEE International Conference on Multimedia and Expo*, volume 1, pages 113–116 vol.1, 2002.

[7] Diederik P. Kingma and Max Welling. An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, 12(4):307–392, 2019.

[8] 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)*, pages 313–317, 2019.

[9] Hans-Peter Kriegel, Peer Kroger, Erich Schubert, and Arthur Zimek. *Interpreting and Unifying Outlier Scores*, pages 13–24.

[10] Poojan Oza and Vishal M. Patel. C2ae: Class conditioned auto-encoder for open-set recognition. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2302–2311, 2019.

[11] Harsh Purohit, Ryo Tanabe, Kenji Ichige, Takashi Endo, Yuki Nikaido, Kaori Suefusa, and Yohei Kawaguchi. MIMII dataset: Sound dataset for malfunctioning industrial machine investigation and inspection. *CoRR*, abs/1909.09347, 2019.

[12] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

[13] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks, 2019.

[14] Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W. Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *CoRR*, abs/1609.03499, 2016.

[15] Phongtharin Vinayavekhin, Tadanobu Inoue, Shu Morikuni, Shiqiang Wang, Tuan Hoang Trong, David Wood, Michiaki Tatsubori, and Ryuki Tachibana. Detection of anomalous sounds for machine condition monitoring using classification confidence. Technical report, DCASE2020 Challenge, July 2020.