Deep Contextual Novelty Detection with Context Prediction

Ellen Rushe ELLEN.RUSHE@UCD.IE and **Brian Mac Namee** BRIAN.MACNAMEEE@UCD.IE
School of Computer Science
University College Dublin,
Belfield
Dublin 4

Editor: Nuno Moniz, Paula Branco, Luís Torgo, Nathalie Japkowicz, Michał Woźniak and Shuo Wang.

Abstract

Contextual novelty detection models detect novelties with respect to a given context. This is crucial in streaming scenarios where the definition of both normal and novel evolve over time. Such models however require contextual labels not only for training but also for detection during deployment. This creates an often unreasonable burden for additional contextual labels during the deployment of these models. In order to eliminate the need for these labels, we propose to predict this contextual information using an auxiliary prediction strategy which takes advantage of the rarity of novel examples, allowing these labels to instead be inferred. The inferred labels are then used as a conditioning criterion for deep autoencoders. We evaluate our approach on a large, public industrial machine sound dataset and show that our approach can successfully recognise context and use this to effectively condition novelty detection models, allowing them to outperform their unconditioned counterparts.

Keywords: novelty detection, anomaly detection, semi-supervised learning, deep learning, audio.

1. Introduction

A distinct limitation of conditioning methods for contextual novelty detection is the need for contextual feature labels during both training and detection. This becomes challenging when the contextual label is a complex feature which may, itself, require a prediction model in order to ascertain its value. For example, in novelty detection for autonomous driving, accounting for complex multi-facited environmental changes (road type, road conditions, etc.) may require a user to describe the driving environment using a contextual label manually, which is a burden on a preoccupied user. A more realistic approach would be to infer complex conditions such as these. In this example, a scene classifier could be used to determine whether the road is a crowded city street or suburban road, for instance. To give another example, in a home monitoring system, moving a device from room to room means having to input the details of the new setting. Here, the room could be inferred when the device's physical location is changed, with the room being the context. If contextual labels are available during training time but not easily available during deployment, there is potential to build a predictive model that can infer these labels during testing. Such a predictive model presents its own challenges, however. While the task of building a context predictor on training data can be achieved by simply using a supervised classification model

on contextual labels, the challenge lies in integrating these predictions into a semi-supervised novelty detection framework when context labels are not available during deployment. This is mainly due to the fact that, although contextual predictions are likely to be correct when presented with normal instances, novel examples in isolation are more likely to be misclassified due to their novel characteristics. This is especially the case where a normal example in one context is completely abnormal in another. Take for instance the reaction a driver might have to a cyclist on a city street versus a cyclist on a highway.

To build an effective context predictor that minimises the misclassification of context and to integrate this into a semi-supervised framework for realistic novelty detection scenarios, it is necessary to make two primary assumptions. It is first assumed that, at testing time, neighbouring examples in a sequence of data relate to each other so their contexts are likely to be the same. This is an especially realistic assumption where data points have a temporal relation to one another. It is also assumed that novelties occur rarely, meaning that the majority of data will be of the normal class. In this paper, we describe the Context-Aware Novelty Detection autoEncoder with Context Prediction (CANDE-CP). This method uses the underlying assumptions outlined above to reduce the effects of context misclassifications caused by novel examples by smoothing the context predictions through the use of the contribution of a window of context predictions preceding a given datapoint.

The remainder of the paper is organised as follows: Section 2 discusses recent advances in both deep novelty detection and deep contextual novelty detection; Section 3 defines the proposed approach for context prediction. The experimental structure and dataset used for evaluation are described in Section 4; Results of the experiments are discussed in Section 5. Finally, Section 6 concludes the work by summarises our main findings.

2. Related Work

Contextual novelty detection assumes the presence of both *contextual attributes*, which depend on some contextual information within the data, and *behavioural attributes*, which are said to be non-contextual (Chandola et al., 2009). The nature of such contextual attributes in anomaly detection techniques often rely on some sort of domain knowledge, meaning that application specific methods are common. Note that we use the terms "novelty" and "anomaly" interchangeably for the purposes of this section, though it should be noted that they do not always amount to the same task outside of the context of this discussion. In acoustic event detection, scene dependent anomaly detection was achieved by Komatsu et al. (2019) using a Wavenet model conditioned on *I-vectors*. *I-vectors* act as an embedding that encodes the degree to which an example deviates from a Universal Background Model. Araya et al. (2016) use historic sensor data along with contextual features with autoencoders for anomaly detection for smart buildings.

In terms of non-domain specific methods, Shulman (2019) modelled contextual attributes and behavioural attributes separately using variational autoenocoders. A strategy using deep autoencoders conditioned using Feature-wise Linear Modulation (FiLM) Perez et al. (2018) in order to adapt to contextual information is proposed by Rushe and Mac Namee (2020a). One of the drawbacks of these methods, however, is that contextual feature values are needed during testing in order to condition networks. In this work we remove the need for these labels at test time using context prediction to infer these values. It is

also noteworthy that adaptation to specific contexts bares some relationship to multi-domain classification (Rebuffi et al., 2017), where a model can shift between domain-specific representations. In contrast, our approach attempts to achieve adaptation in a semi-supervised setting where the criteria used for adaptation uses preceding examples within a sequence. In our work, we extend the the semi-supervised novelty detection framework proposed by Rushe and Mac Namee (2020a) to work with context prediction.

3. Methodology

In this section, we will briefly explain CANDE (Rushe and Mac Namee, 2020a), which is the base of our proposed algorithm. We then go on to explain our proposed context prediction and aggregation methodology which removes the need for contextual labels during detection.

3.1. Context-Aware Novely Detection autoEncoder (CANDE)

CANDE (Rushe and Mac Namee, 2020b) conditions deep autoencoders (AE) on contextual information in a layer-wise fashion for novelty detection. This is done using a conditioning strategy originally introduced by Perez et al. (2018) known as Feature-wise Linear Modulation (FiLM). FiLM consists of a layer-wise affine transformation applied to network activations. In this way, a single network can be used to detect novelties in many different contexts. This vector can represent a one-hot encoded label or a more complex representation of context such as an embedding. The activations of each layer are modulated using a different affine transformation.

For the embedding, Rushe and Mac Namee (2020a) introduce a *context discriminator* which is trained to recognise the contexts that an input example belongs to in order to generate embeddings. The context disciminator, D, is a fully connected neural network:

$$\hat{y} = D(\boldsymbol{x}, \theta^c) \tag{1}$$

where $\hat{y}$ defines the predicted context class for input $\boldsymbol{x}$ and network parameters θ^c. The activations of the penultimate layer of this model are used to generate the embedding vector. This embedding vector is created by taking the mean of the penultimate layer activations for each example in the training set belonging to a particular context.

3.2. Context-Aware Novely Detection autoEncoder with Context Prediction (CANDE-CP)

Contextual novelty detection requires that models adapt to contextual information. Specifically, in the case of deep autoencoders, this means that it is necessary to modulate the output of the network to reflect the shift in context.

3.2.1. Predicting Context

One of the difficulties of many contextual novelty detection approaches is that contextual labels are needed in both the training and detection phases. However, these labels may not be available at detection time. The context discriminator, which can classify the context of a given example, can be used to infer the most appropriate context on which to condition

the novelty detection network. Essentially, here the accuracy of the discriminative model used is relied on when embedding examples. However, this raises another issue. Given the inherent unusual nature of novel examples, these examples may be prone to context misclassification due to their unusual, novel nature. If we assume that examples from the same context tend to arrive together in a streaming scenario, however, we can derive the context from a window of examples preceding a given input query. This requires the use of a number of context predictions from a window of previous examples which will be discussed in detail in the next section.

3.2.2. Aggregating Context Prediction

To start the context prediction process, the context discriminator is used to classify the context based on a window containing the current timestamp and some past examples. This window, which will be denoted as Q, operates in a first-in-first-out fashion with the context predictions for the oldest examples in the queue being removed first. At the beginning of the sequence, the context prediction is done with fewer previous past examples as Q will not yet be full. A naive approach to ascertaining the context from this window would be to simply use the most common context predicted within the window. Practically, this means that for each context $c \in C$ (the set of all possible contexts) the frequency of that context is obtained in the window Q. The scoring function is defined as follows:

$$s(c) = \sum_{t=1}^{|Q|} \delta(Q_t, c) \tag{2}$$

where $|Q|$ is the number of elements in the window Q, Q_t represents the predicted context at timestamp t in the window and

$$\delta(c_i, c_j) = \begin{cases} 1 & \text{if } c_i = c_j \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where c_i and c_j represent context labels. We can then simply calculate the aggregated predicted context $\hat{c}$ in set C with the highest score.

$$\hat{c} = \underset{c \in C}{\operatorname{argmax}} \; s(c) \tag{4}$$

This would alleviate the effect of novel examples, given that it is assumed they are rare. This approach, however, leads to problems immediately after a context changes in a streaming scenario. This is because the most common context, for a period, will inevitably be the context that precedes the new context. To prevent this issue, an exponentially decaying weight is applied to each context prediction in the window, giving a higher weight to more recent examples. The decay factor λ, is a hyperparameter chosen depending on the degree of importance that the most recent predictions in the window hold. The following new scoring function is defined:

$$s(c, \lambda) = \sum_{t=1}^{|Q|} (1 - \lambda)^{|Q|-t} \delta(Q_t, c) \tag{5}$$

where $t = 1$ represents the oldest timestep and $t = |Q|$ represents the most recent timestep.

For recent examples, t approaches $|Q|$ and therefore $|Q| - t$ is close to zero. This leads to a value for more recent timesteps approaching 1 for $(1 - \lambda)^{|Q|-t}$ given the range $0 \leq \lambda < 1$. Each predicted context in the window is allocated a weight depending on the position in the window at which it occurs. For less recent examples, as t approaches 1, $|Q| - t$ will be closer to $|Q|$, meaning $(1 - \lambda)^{|Q|-t}$ overall will have a smaller value for older examples. A λ value of 1 will remove the effect of past examples while, as the value of λ gets closer to zero, the weight of past predictions will be higher. After obtaining a score for each context, Equation 4 is applied to obtain the highest scoring context. The procedure for generating reconstruction error using this approach to aggregate context predictions for conditioning is given in Algorithm 1.

Algorithm 1: Aggregated Predicted Context Conditioning for CANDE-CP

Input: $\boldsymbol{x} \in \mathbb{R}^d$: Input vector, w: window size, Q: Queue, λ: decay factor, $f(., \theta^e)$: AE encoder with parameters θ^e, $g(., \theta^d)$: AE decoder with parameters θ^d, $D(., \theta^c)$: context discriminator, $\phi : C \mapsto V$: mapping from context to embedding, $c \in C$: context labels, $\boldsymbol{v}_c \in V$: embedding for context c

Output: $\boldsymbol{x}' \in \mathbb{R}^d$: Reconstruction of $\boldsymbol{x}$

$\hat{y} \leftarrow D(\boldsymbol{x}, \theta^c)$

if $|Q| = w$ **then** // We remove the oldest class in the window.

  Q.dequeue()

end

Q.enqueue($\hat{y}$)

$\hat{c} \leftarrow \underset{c \in C}{\text{argmax}} \sum_{t=1}^{|Q|} (1 - \lambda)^{|Q|-t} \delta(Q_t, c)$

if ϕ **then** // If using embedding method, i.e. ϕ exists.

  $\mathbf{v}_{\hat{c}} \leftarrow \phi(\hat{c})$

  $\boldsymbol{x}' \leftarrow g(f(x, \theta^e, \mathbf{v}_{\hat{c}})\theta^d, \mathbf{v}_{\hat{c}})$

else

  $\boldsymbol{x}' \leftarrow g(f(x, \theta^e, \hat{c})\theta^d, \hat{c})$

end

A straightforward extension of this method is to use the embedding corresponding to the predicted label $\hat{c}$ as the context vector. Rushe and Mac Namee (2020b) found context embeddings to outperform one-hot-encoded labels of context for CANDE. This only requires a slight modification to Algorithm 1. As described in Algorithm 1, the aggregated context prediction $\hat{c}$ is obtained in the same way as before, however this label is not used directly as the context vector. Instead $\hat{c}$ is mapped to its corresponding embedding vector $\boldsymbol{v}_c$. This is done using the mapping ϕ, which maps context labels to the embeddings created at training time as described in Section 3.1. It is expected that these embeddings contain richer contextual information than the one-hot-encoded labels alone.

The windowing method in Algorithm 1 has the capacity to be used with any context prediction classifier. This may be desirable where computational expense of deep algorithms is an issue. Similarly, if the context classifier discussed in the preceding sections is used, Algorithm 1 can be used with any contextual novelty detection algorithm.

4. Experimental Set-up

This section discusses the dataset used for evaluating the proposed approaches, the network architectures of these approaches, and those of the baselines used in the evaluation. It also describes the evaluation procedure used to assess the performance of the proposed models.

4.1. MIMII Dataset

In our experiments we use the MIMII dataset (Purohit et al., 2019), an industrial machine sound dataset containing machine sounds both in normal and anomalous states. The overall dataset is imbalanced, with a larger number of normal examples than abnormal examples. In the public version of the MIMII dataset four different types of machine were recorded: 'valve', 'pump', 'fan' and 'slide rail'. Recordings from four different models of each machine are included. There are 16 individual machine models, each of which we consider as defining a context. Each file in this dataset contains a 10-second clip of audio mixed with background noise at varying signal-to-noise ratios (specifically -6dB, 0dB and 6dB) to provide realistic environmental noise. Audio is sampled at 16KHz and contains eight 16-bit channels. Following previous work on this dataset (Purohit et al., 2019), we reduce the number of channels to one and compute 64 log-mel spectrogram filters with frame size 1,024 and hop-length 512 over a period of 5 frames.

The overall aim of the proposed approach is to detect novelties even in the presence of contextual shifts during deployment. The overall desired characteristics for the evaluation set are therefore that contextual shifts should occur within a stream of data with sporadic novelties preceded by a large amount of normal data. This is in line with the assumption in novelty detection that the novel class is rare (Chandola et al., 2009). The MIMII dataset provides a number of normal files along with abnormal files. In the original setup of the MIMII baseline experiments (Purohit et al., 2019), the number of normal files was set to be equal the number of abnormal files in the test set. This balanced normal and novel classes, which is unrealistic given that it is assumed that novelties are rare. It was therefore necessary to increase the number of normal examples for each machine ID (i.e., each context) in the testing data so novel examples were more likely to be preceded by normal examples in a given window. This was achieved by adding 50% of the overall normal files for each machine ID from the dataset to the test data, leaving the other 50% for training. Because there were more normal examples in the dataset overall, this led to a varying prevalence of novelties within each context depending on the number of novel files for each machine ID. A portion of normal examples from the test set was used for validation. This portion's size was equal to 10% of the number of files in the training data (which are all normal files). Please note that these normal validation examples were also used in the evaluations for all models including baselines. During evaluation, audio files were then streamed in, one machine ID (i.e. context) at a time, with novel files from the same machine ID being randomly placed throughout the stream. Given that the evaluation stream moves from one context to the

next, the algorithm has to adapt to each oncoming context in order to correctly identify novelties.

4.2. Models

In order to evaluate the efficacy of the proposed prediction strategy, conditioned models are evaluated against two unconditioned models using the same design as Rushe and Mac Namee (2020a). All models were trained and evaluated over three random weight initialisations. The optimal training epoch is determined using the validation performance accuracy for all models. All code related to the implementation, experimental setup and data preprocessing is available in this work's Github repository [1]

Context Discriminator The context discriminator used is a seven layer fully connected neural network using ReLU activations (He et al., 2015) throughout. There are 16 output classes corresponding to the 16 contextual classes in the dataset, namely the number of machine IDs. The network is optimised using cross entropy loss with an Adam optimiser (Kingma and Ba, 2014), a batch size of 512, a maximum number of epochs of 50 and a learning rate of 10^{-4}.

Individual Models For each context, a single autoencoder is trained on only data from that context and evaluated on only test data from that context. This leads to 16 individual models in total. Training individual models on each context separately means that these models are not being biased by data from any other context. This is therefore treated as a type of "optimal" result and measures how closely the conditioned models can match the performance of these oracle individual models.

Unconditioned Single Model As a baseline, we would also like to see whether the proposed models are actually effective against their unconditioned counterparts – recall that the overall aim is to create a system where fast modulation of deep architectures can occur. Combining data from all contexts without conditioning is expected to degrade results significantly compared to individual unconditioned models. Autoencoders are fully connected and contain three layers in the encoder with three layers in the decoder.

Conditioned Models As was discussed in Section 3 two different methods to condition models at test time are introduced using both predicted labels and corresponding embeddings. For consistency between experiments the window size for the past audio files used for prediction is fixed to 50 and an exponential decay factor, λ, of 0.1 is used for all conditioning approaches where windowing is used. For models conditioned on labels, a 16 dimensional one-hot-encoded vector is passed to the FiLM operation. The optimal discriminative architecture determined by Rushe and Mac Namee (2020a) was used, with a penultimate layer size of 64, leading to an embedding of the same dimension. The same architecture is used for conditioned models as for unconditioned models in order to accurately measure the performance of these models relative to their unconditioned counterparts.

Furthermore, to evaluate the effectiveness of the windowed context aggregation method outlined in Section 3.2.2, the performance of CANDE-CP with "raw", un-windowed predictions, without context aggregation is also evaluated. The accuracy of the predictions with

1. https://github.com/EllenRushe/CANDE-CP

versus without context aggregation is also evaluated to show the degradation in context prediction accuracy when using raw predictions alone. In a similar manner to the individual models, a second oracle model is compared, which, instead of using predictions, utilises ground-truth context labels. This experiment helps us explore how closely CANDE-CP can approximate the performance of a conditioned model using the true label.

For all autoencoder models, ReLU activations (He et al., 2015) were used throughout and Mean Squared Error (MSE) was used as the loss function and optimised using Adam (Kingma and Ba, 2014). Futhermore, the maximum number of epochs was set to 100, the batch-size was 256 and the learning rate was 10^{-4}.

4.3. Evaluation

In line with the dataset baselines in (Purohit et al., 2019), for each example in each 10 second audio file, MSE is calculated. The mean of these errors is then taken in order to obtain an overall reconstruction error for each file. It is assumed that files exhibiting a high reconstruction error indicate novelties, while those with low reconstruction error contain a normal event, therefore this error can be used as a novelty score. The ratio between normal and novel examples in the test set is imbalanced in this case with the novel examples being in the minority class. Due to the fact that ROC-AUC can sometimes overestimate performance in this scenario, to get a more complete picture of performance, Area Under the Precision-Recall Curve (PR-AUC) is measured (Japkowicz and Shah, 2011).

5. Results

This section discusses the results of the evaluation. First, unconditioned models are compared with different flavours of CANDE-CP. Next, the best performing model from this section is compared and contrasted with oracle models in order to evaluate how closely CANDE-CP can match models with ground-truth information.

5.1. Conditioned vs. Unconditioned

Table 1 compares the single unconditioned model (*AE no Cond*), CANDE-CP with one hot encoding (*CANDE-CP one hot*), and CANDE-CP with embeddings (*CANDE-CP embed*). For the best performing flavour of CANDE-CP in terms of average rank, *CANDE-CP embed*, the same model without context aggregation (*CANDE-CP embed no CA*) is also included in these tables to show the effectiveness of the proposed context aggregation strategy. The performance of each modelling approach is ranked for each dataset, and to summarise results, the average rank is calculated. Performance is measured using Area Under the Precision-Recall curve (PR-AUC) $\pm$ 95% confidence intervals (t-distribution) with $n = 3$, where n is the number of random initialisations. Note that the confidence intervals are calculated based on the mean AUC to show the range in performance between random initialisations of the model. Algorithms are ranked from 1 to 4 across each row, the lower the rank, the better. The average rank reported is computed for each algorithm on the column level.

The results of the evaluation show a clear advantage is to be gained from conditioning, as all conditioned models (i.e., flavours of CANDE-CP) show a higher rank than a single

Table 1: An overview of the results from all experiments using Area Under the Precision-Recall curve (PR-AUC) ± 95% confidence intervals with the average ranking.

Machine Name	ID	SNR	AE no cond.	CANDE-CP one hot	CANDE-CP embed no CA	CANDE-CP embed
fan	00	0dB	0.521±0.051	0.523±0.057	0.552±0.037	0.541±0.037
		6dB	0.644±0.047	0.719±0.083	0.745±0.027	0.733±0.027
		-6dB	0.518±0.036	0.510±0.035	0.518±0.023	0.519±0.026
	02	0dB	0.670±0.135	0.848±0.024	0.814±0.064	0.856±0.015
		6dB	0.913±0.030	0.947±0.038	0.939±0.022	0.935±0.033
		-6dB	0.485±0.026	0.622±0.031	0.542±0.023	0.628±0.034
	04	0dB	0.614±0.014	0.663±0.017	0.677±0.036	0.662±0.061
		6dB	0.847±0.034	0.858±0.028	0.890±0.010	0.846±0.034
		-6dB	0.431±0.011	0.445±0.014	0.461±0.023	0.457±0.034
	06	0dB	0.704±0.196	0.892±0.030	0.825±0.019	0.949±0.078
		6dB	0.913±0.024	0.934±0.003	0.964±0.016	0.961±0.052
		-6dB	0.464±0.059	0.659±0.075	0.556±0.020	0.768±0.194
pump	00	0dB	0.273±0.055	0.378±0.102	0.438±0.203	0.458±0.187
		6dB	0.295±0.119	0.467±0.129	0.562±0.163	0.526±0.172
		-6dB	0.305±0.040	0.349±0.094	0.398±0.141	0.409±0.117
	02	0dB	0.266±0.085	0.261±0.048	0.281±0.111	0.279±0.109
		6dB	0.280±0.074	0.280±0.062	0.285±0.171	0.286±0.149
		-6dB	0.239±0.058	0.232±0.053	0.239±0.083	0.239±0.080
	04	0dB	0.482±0.224	0.878±0.047	0.574±0.049	0.869±0.044
		6dB	0.610±0.160	0.934±0.072	0.681±0.117	0.916±0.038
		-6dB	0.414±0.181	0.793±0.055	0.495±0.047	0.784±0.029
	06	0dB	0.161±0.005	0.200±0.046	0.193±0.072	0.227±0.105
		6dB	0.153±0.004	0.201±0.085	0.193±0.121	0.240±0.169
		-6dB	0.176±0.011	0.195±0.036	0.194±0.063	0.207±0.055
slider	00	0dB	0.919±0.019	0.947±0.014	0.933±0.024	0.930±0.027
		6dB	0.981±0.023	0.992±0.003	0.986±0.007	0.986±0.011
		-6dB	0.857±0.025	0.858±0.052	0.851±0.042	0.833±0.043
	02	0dB	0.505±0.103	0.687±0.108	0.629±0.174	0.643±0.180
		6dB	0.369±0.099	0.700±0.234	0.606±0.411	0.622±0.435
		-6dB	0.399±0.047	0.496±0.059	0.476±0.091	0.482±0.097
	04	0dB	0.706±0.020	0.749±0.032	0.857±0.038	0.799±0.078
		6dB	0.778±0.043	0.818±0.036	0.922±0.044	0.841±0.051
		-6dB	0.568±0.006	0.606±0.040	0.698±0.049	0.634±0.063
	06	0dB	0.334±0.061	0.283±0.022	0.291±0.043	0.294±0.041
		6dB	0.519±0.076	0.391±0.012	0.422±0.134	0.407±0.119
		-6dB	0.269±0.032	0.253±0.016	0.250±0.007	0.254±0.004
valve	00	0dB	0.154±0.003	0.186±0.033	0.187±0.014	0.183±0.015
		6dB	0.141±0.024	0.177±0.019	0.185±0.060	0.178±0.060
		-6dB	0.171±0.007	0.204±0.026	0.187±0.020	0.198±0.024
	02	0dB	0.370±0.045	0.350±0.011	0.355±0.021	0.349±0.024
		6dB	0.439±0.039	0.429±0.053	0.383±0.046	0.422±0.061
		-6dB	0.283±0.010	0.272±0.017	0.275±0.008	0.269±0.002
	04	0dB	0.191±0.033	0.244±0.021	0.242±0.051	0.238±0.049
		6dB	0.176±0.018	0.243±0.002	0.237±0.036	0.232±0.033
		-6dB	0.169±0.017	0.194±0.020	0.197±0.038	0.189±0.038
	06	0dB	0.207±0.008	0.217±0.004	0.198±0.017	0.224±0.025
		6dB	0.269±0.041	0.268±0.004	0.226±0.019	0.282±0.037
		-6dB	0.195±0.002	0.189±0.001	0.189±0.005	0.188±0.004
Average rank			3.3542	2.2917	2.2083	2.1458

unconditioned model with the same base architecture. This indicates that the context prediction strategies improved performance and helped the network perform in a context sensitive way. Conditioning using a one-hot-encoding of the context label also performs well, though in general not as well as most models utilising embeddings.

Table 2: Accuracy of context predictor with windowed predictions as discussed in Section 3 (*CANDE-CP embed*) and context predictor without windowing (*CANDE-CP no CA embed*) ± 95 % confidence intervals (binomial).

Model	% Accuracy ± 95% ci.
CANDE-CP embed	88.987 ± 0.343
CANDE-CP no CA embed	76.851 ± 0.462

As discussed above, CANDE-CP without context aggregation is compared against CANDE-CP with context aggregation in terms of the accuracy of the context prediction strategy, and the resulting effect of this accuracy on novelty detection performance. We can see in Table 2 that context aggregation clearly improves the performance of context recognition, with roughly a 12 percentage point increase in accuracy. This is also reflected in the novelty detection performance in Table 1 with CANDE-CP with context aggregation (CANDE-CP embed) achieving a higher rank.

5.2. Oracle comparison

Unconditioned individually trained autoencoders, *AE indv.*, provide an oracle reference to which a conditioned models can be compared. It is expected that the individually trained models will outperform conditioned models as these have been specifically trained on each context individually and only evaluated on that context. A two-sided Wilcoxon-signed rank test (Wilcoxon, 1992) was performed to evaluate whether there is a statistical difference between the performance of the individual models and CANDE-CP with embeddings in terms of PR-AUC. The test statistics and p-values for PR-AUC are reported in Table 3. The null hypothesis here states that these pairs of results are from the same distribution. It was found that, with a significance level of .05, that the null hypothesis cannot be rejected and that there is no statistical difference between the performance of the individually trained models and CANDE-CP with embeddings. This shows that the proposed model is capable of matching the performance of individually trained models.

This then leads to the question of how much better a model trained with ground-truth contexts labels would be. A two-sided Wilcoxon-signed rank test was again performed between the PR-AUC scores of *CANDE-CP embed* and of this oracle model, with test statistics and p-values again being reported in Table 3. With a significance level of .05, that the null hypothesis of the paired results being from the same distribution can be rejected. This means that there was a statistically significant difference found between *CANDE embed oracle* and *CANDE-CP embed*. This indicates that, given a more accurate context predictor, CANDE-CP could exceed the performance of individually trained models, leaving room for further performance improvements in future. We also noticed that *CANDE embed oracle*

and, in some contexts, even *CANDE-CP embed*, outperform models individually trained on each context separately, this may be because the conditioned models have access to more data with which to learn non-contextual features while still maintaining the advantage over their unconditioned counterparts due to their ability to bias the network to a specific contexts dynamically.

Table 3: The test statistic (W) and p-value for a two-sided Wilcoxon-signed rank test between *CANDE-CP embed* and *AE indiv.*, and between *CANDE-CP embed* and *CANDE embed oracle* in terms of PR-AUC (See Table **??** in the Appendix for PR-AUC scores).

Oracle model	W-statistic	p-value
AE indiv.	585	0.98
CANDE embed oracle	307	0.007

6. Conclusion

CANDE-CP adds context classification to the CANDE model for deep contextual novelty detection to eliminate the need for ground-truth contextual labels at test time. The need for contextual labels at test time can be eliminated by using the underlying assumptions inherent to novelty detection to predict such contexts to efficiently infer context when the model is deployed. Two strategies for context prediction were proposed, one using predicted labels from a discriminative model, and another using embeddings of these labels. This means that this strategy can be used both when a deep network has been used for context prediction, or with another form of context prediction. Furthermore, the context aggregation method proposed allows for accurate predictions of reoccurring contexts even in the presence of sporadic novelties. The results not only show that conditioned variants of CANDE-CP can outperform unconditioned models using the same base architecture, but also show that conditioned models using embeddings can outperform some models trained on individual contexts. This makes a powerful case for the use of conditioned autoencoders for novelty detection, and demonstrates that these can even be used when context labels are not available at test time.

Acknowledgments

We would like to warmly thank Thomas Laurent for his suggestions and proofreading of this paper. This work has been supported by a research grant by Science Foundation Ireland under grant number SFI/15/CDA/3520.

References

Daniel B Araya, Katarina Grolinger, Hany F ElYamany, Miriam AM Capretz, and G Bitsuamlak. Collective contextual anomaly detection framework for smart buildings. In *2016*

International Joint Conference on Neural Networks (IJCNN), pages 511–518. IEEE, 2016.

Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Computing Surveys (CSUR)*, 41(3):15, 2009.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

Nathalie Japkowicz and Mohak Shah. *Evaluating Learning Algorithms: A Classification Perspective*. Cambridge University Press, 2011.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Tatsuya Komatsu, Tomoki Hayashiy, Reishi Kondo, Tomoki Todaz, and Kazuya Takeday. Scene-dependent anomalous acoustic-event detection based on conditional wavenet and i-vector. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 870–874. IEEE, 2019.

Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

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. *arXiv preprint arXiv:1909.09347*, 2019.

Sylvestre-Alvise Rebuffi, Hakan Bilen, and Andrea Vedaldi. Learning multiple visual domains with residual adapters. In *Advances in Neural Information Processing Systems*, pages 506–516, 2017.

Ellen Rushe and Brian Mac Namee. Deep context-aware novelty detection. *arXiv preprint arXiv:2006.01168*, 2020a.

Ellen Rushe and Brian Mac Namee. Deep context-aware novelty detection. *1st NeurIPS workshop on Interpretable Inductive Biases and Physically Structured Learning, 2020.*, 2020b.

Yaniv Shulman. Unsupervised contextual anomaly detection using joint deep variational generative models. *arXiv preprint arXiv:1904.00548*, 2019.

Frank Wilcoxon. Individual comparisons by ranking methods. In *Breakthroughs in statistics*, pages 196–202. Springer, 1992.