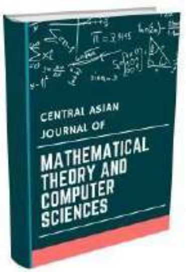

CENTRAL ASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES
https://cajmtcs.centralasianstudies.org/index.php/CAJMTCS
Volume: 06 Issue: 01 | January 2025 ISSN: 2660-5309

.

Article

ECG Anomaly Detection using Time Series Deep Learning Algorithms

Mohammed Ayden Omar*[1], Layth Hammood[2]

1. Ministry of Planning, Kirkuk Planning Director, Kirkuk, Iraq
2. Presidency of University of Kirkuk, University of Kirkuk, Kirkuk, Iraq

* Correspondence: moayden@yahoo.com

Abstract: This paper implements a version of a VAE for anomaly detection known as VELC. VELC uses a re-encoder at the end of the decoder. The results are evaluated with several versions of vanilla VAEs containing only an encoder and decoder. Initially, we built models that yield reliable results for the ECG dataset, and thereafter we used those models with the regular passenger cars dataset provided by the instructors of the course. We measured the performance of the models using the AUC. Thresholding different values of the mean squared error score between the actual and reconstructed time series calculated the AUC curve. Next, the best results were obtained with a bidirectional LSTM with a re-encoder for a value of 0.997. The regular passenger cars data yielded poor results for this method of classification, which motivated a different approach thresholding the Lilliefors test statistic of the residuals. This provided better results, with the best AUC value of 0.754 for a bidirectional GRU without re-encoding.

Keywords: Timeseries, Variational Inference, VAE, Anomaly Detection

1. Introduction

Denmark spends approximately 5 million DKK per year on assessing road conditions and optimizing maintenance strategies with a focus on safety, comfort, durability, etc. Apart from the high cost, limitations such as weather and road geometry make the monitoring and maintenance of the road a cumbersome process [1]. As an alternative way to monitor, maintain and manage the roads, the Danish Innovation Fund has invested in the Live Road Assessment project (LiRA). The overall goal of the LiRA project is to collect data from sensors installed in green mobility cars (GM) [2] and develop associated models to assess road conditions. Road wear can thus be detected and improved much earlier, enabling the roads to be maintained more efficiently and for fewer resources [3].

The present study aims to analyse part of the data collected from those sensors and develop an anomaly detection model able to spot roads that requires maintenance. Throughout the literature, several ways have been used to detect anomalies, from traditional machine learning (k-means clustering, Gaussian mixture models, PCA) to deep neural networks, with the latter performing significantly better as the number of data is growing [4]. Recent studies from [5], [6], [7], [9] have applied variational auto-encoders (VAE) in different variations in order to detect anomalies in various datasets (time series, images etc.). The results of complex VAEs compared to conventional machine learning

Citation: Omar M. A., Hammood L. ECG Anomaly Detection using Time Series Deep Learning Algorithms. Central Asian Journal of Mathematical Theory and Computer Sciences 2025, 6(1), 152-164.

Received: 11th Feb 2025
Revised: 20th Feb 2025
Accepted: 28th Feb 2025
Published: 4th Mar 2025

algorithms, as well as vanilla VAEs, have shown that they have great potential to detect anomalies much better. In the present study, we have decided to reproduce the VAE model with a constraint network and re-encoder, as described in [10]. This so-called VELC model has an interesting and different architecture compared to the other VAEs in the literature. The VELC model shows extremely good performance compared to the other state of the art models for anomaly detection.

Additionally, the model has been tested on sequential datasets similar to those that come from the car sensors. Therefore, it constitutes an appealing approach for our case study. In our implementation, though, we will exclude the constraint network in the latent space due to discrepancies found in the literature regarding its application.

1.1 Datasets

This paper uses two datasets. The ECG dataset contains the heartbeats of 9499 samples over 140 time points. The dataset contains 5546 normal samples and 3953 anomaly samples. The models are trained on the normal data and tested on the anomaly data. One can see from Figure 1 that there is a clear distinction between the normal and anomaly data.

The GM dataset provided by the course instructors contains 5031 sample points and the features contain the vehicle acceleration for 369 time steps. The target columns contain the index of road.

roughness. We consider the anomaly data the ones with an International Road Roughness Index (IRI) value two and normal data the ones with an IRI value < 2. Additionally, solely the perpendicular.

direction "z" is considered for the modeling as it is the most informative one. Due to the noisy nature of the data, we do not plot them all together but we can see how the series look like in Figure 1.

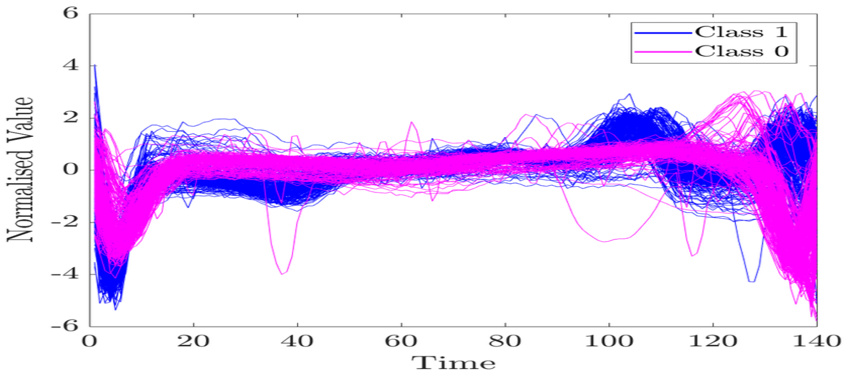

Figure 1. Class 0 represents the anomaly heartbeat, and class 1 the normal heartbeat for the ECG dataset.

2. Materials and Methods

2.1 Variational AutoEncoder (VAE)

The Variational AutoEncoder (VAE) is a generative model that combines Bayesian inference with the autoencoder framework, consisting of an encoder and a decoder trained to minimize reconstruction error. In VAE, (x) represents the data to be modeled, and (z) is the latent variable. The probability distributions are denoted as (p(x)) for data, (p(z)) for the latent variable, and (p(x|z)) for the conditional distribution of generating data given the latent variable. The generative process involves encoding input data (x) into a latent representation (z), sampling (z) from (p(z|x)), typically a Gaussian distribution, and decoding (z) back to (x) to match the original input closely. Mathematically, it is expressed as (p(x|z) = int p(x|z)p(z)dz). The VAE maximizes the evidence lower bound (ELBO), balancing reconstruction loss and regularization loss, enabling the generation of new data samples similar to the original data:

$$P(X) = \int P(X,Z))dz = \int P(X|Z)P(Z)dz \quad (1)$$

This is analytically impossible to compute since the search space for z is continuous and computationally large. The trick is to approximate the posterior with $q(z|x)$ in the encoder network, which is more feasible to compute compared to our unknown true posterior, $p(z|x)$. The log-likelihood $p(x|z)$ is learned from the decoder network. After a series of statistical derivations, the marginal log-likelihood of x can be written as follows:

$$\log p(x) = KL[q(Z|X) \parallel P(Z|X)] + \mathcal{L}(X) \quad (2)$$

Where

$$L(x) = Eq(z|x) [\log p(x, z)] - Eq(z|x) [\log q(z \mid x)] \quad (3)$$

It is called variational lower bounds or evidence lower bound objective (ELBO). Kullback-Leibler divergence, KL $[q(z \mid x) \parallel p(z \mid x)]$ measured the similarity of the two distributions $p(z \mid x)$ and $q(z \mid x)$. Hence, log $p(x)$ is constant. To minimize the KL divergence, we have to maximize the variational lower bound, L (x). With further statistical derivation, we can rewrite the variational lower bound as follows:

$$L(x) = -KL [q(z \mid x) \parallel p(z)] + Eq(z|x) [\log p(x \mid z)] \quad (4)$$

2.2 Long short-term memory (LSTM)

Long short-term memory (LSTM) is an advanced architecture of artificial recurrent neural networks (RNN) that features feedback connections, enabling it to process not only single data points but also entire sequences of data. An LSTM unit comprises a cell, an input gate, an output gate, and a forget gate. The cell is responsible for retaining values over arbitrary time intervals, which allows the network to maintain and recall information over long periods. The input gate controls the extent to which new information flows into the cell, the forget gate determines what portion of the information stored in the cell should be discarded, and the output gate regulates the information to be sent out of the cell. This structure allows LSTM networks to effectively manage the long-term dependencies in sequential data, making them particularly useful for tasks like time series prediction, natural language processing, and speech recognition. Figure 2 illustrates the detailed architecture of an LSTM unit, highlighting how these components interact to maintain and update the cell state.

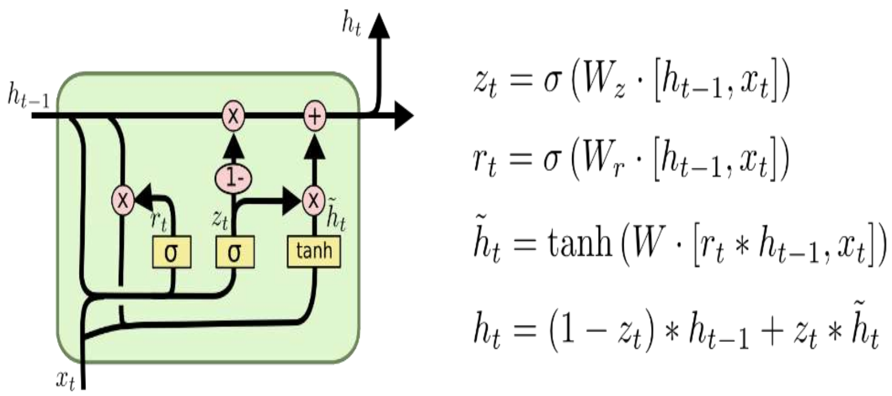

Figure 2. LSTM unit architecture.

2.3 Gated Recurrent Unit (GRU)

The Gated Recurrent Unit (GRU) architecture presents a streamlined alternative to the Long Short-Term Memory (LSTM) network by combining the forget and input gates into a single gate known as the update gate and merging the cell state and hidden state into one. This simplification reduces the number of parameters in the model, resulting in faster training and convergence without compromising performance. The update gate in a GRU regulates the information that is added to or removed from the hidden state, effectively controlling the flow of information and maintaining long-term dependencies in the data. Additionally, the GRU includes a reset gate that determines the amount of past information to forget. This architecture not only leads to more efficient computation but also generalizes well across different tasks, making it a preferred choice in many applications where LSTMs were traditionally used. Figure 3 illustrates the GRU architecture, highlighting its key components and how they interact to manage the information flow.

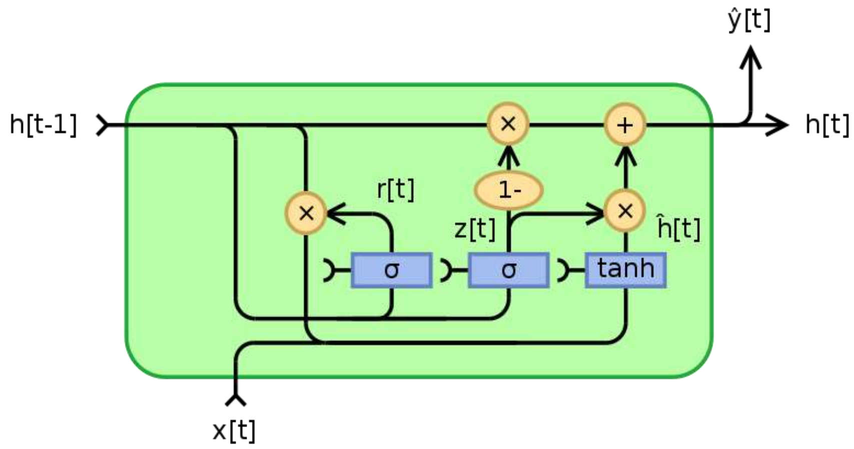

Figure 3. GRU unit architecture.

2.4 Methodology

The structure of the model has been inspired by Zhang and is illustrated in Figure 4. The LSTM architecture refers to bi-directional LSTM, which is two LSTM networks stacked on top of each other. The model has also been tested with GRU architecture.

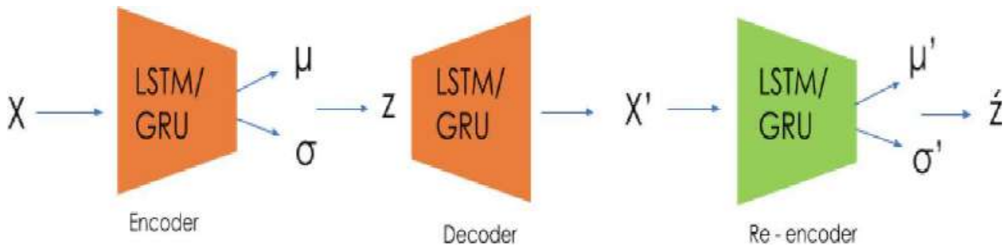

Figure 4. The model architecture.

Following the approach outlined in, our model aims to detect anomalies in sequential data (time series) by leveraging the reconstruction error of a generative model. This model is specifically trained on normal, non-anomalous data, allowing it to capture the underlying patterns and characteristics of such data accurately. During the training phase, the model learns to reconstruct normal data with high precision, resulting in relatively small reconstruction errors. When the model encounters abnormal or anomalous data during testing, it struggles to reconstruct it accurately due to the deviations from the learned normal patterns, leading to significantly larger reconstruction errors. By setting a threshold for reconstruction error, the model can effectively distinguish between normal and abnormal data, flagging instances with high reconstruction errors as potential anomalies. This method ensures a robust detection mechanism, relying on the generative model's inherent ability to differentiate between learned normal sequences and unforeseen anomalous patterns.

As discussed in Section 1, the first part of the model consists of the VAE framework with an additional re-encoder network. The re-encoder will add complexity and more parameters to the network, meaning that the whole model can extract more features from original and latent space which will yield (according to Zhang) better performance of the model. The loss function consisted of three parts, as described in. The first part is the loss function from VAE, meaning the reconstruction loss $Lrec\ x$ and the KL loss $LKL1$.

$$L_{rec_x} = \parallel X - X' \parallel_2^2 \tag{5}$$

$$L_{KL_1} = \frac{1}{2}\sum_{i=1}^{z}[(\mu_i^2 + \sigma_i^2) - 1 - log(\sigma_1^2)] \tag{6}$$

The second part refers to the re-encode's loss function, which is the same as the KL loss $LKL1$ from the VAE framework.

$$L_{KL_1} = \frac{1}{2}\sum_{i=1}^{z'}[(\mu'^2_i + \sigma'^2_i) - 1 - log(\sigma'^2_1)] \tag{7}$$

Lastly, the third part is the error of the latent vector.

$$L_{latent} = \parallel Z - Z' \parallel_2^2 \tag{8}$$

The loss function of the entire model is calculated as:

$$L_{model} = L_{rec_x} + L_{KL_1} + L_{Latent} + L_{KL_2} \tag{9}$$

Here, we use the area under curve (AUC) as a metric of the overall performance of the model. In order to validate the performance of the architecture presented in Figure 1, we compare it against common VAE with LSTM and GRU architecture. Additionally, we visualize the latent space of the re-encoder for the ECG and GM data using the PCA method to check the separability of the classes in the latent space. The latent space consists of 10 vectors in both the encoder and re-encoder. The number of vectors was tested as a hyperparameter along with values such as 15 and 20.

3. Results

3.1 Time Series Models Results

The project employs an Autoencoder LSTM network architecture realized through the PyTorch framework. This architecture is chosen for its ability to learn complex temporal patterns in ECG data. The LSTM autoencoder comprises an encoding stage to compress the input data into a condensed representation and a decoding stage to reconstruct the data. The model is trained exclusively on datasets of normal ECG readings. This approach enables the model to learn a baseline of what constitutes a normal heart rhythm. The trained model continuously analyzes incoming ECG data [11].

The focus is on the reconstruction error: significant deviations from the learned normal patterns are flagged as anomalies. When anomalies are detected, the system immediately triggers alerts to the patient and healthcare providers, enabling swift medical response. This proactive, real-time monitoring and alert system represents a significant leap in patient-centered cardiac care, potentially reducing the risk of late detection in cardiac events. Figures 5–14 illustrate the results of different time series models used for comparison with the proposed model for ECG Anomaly Detection [12].

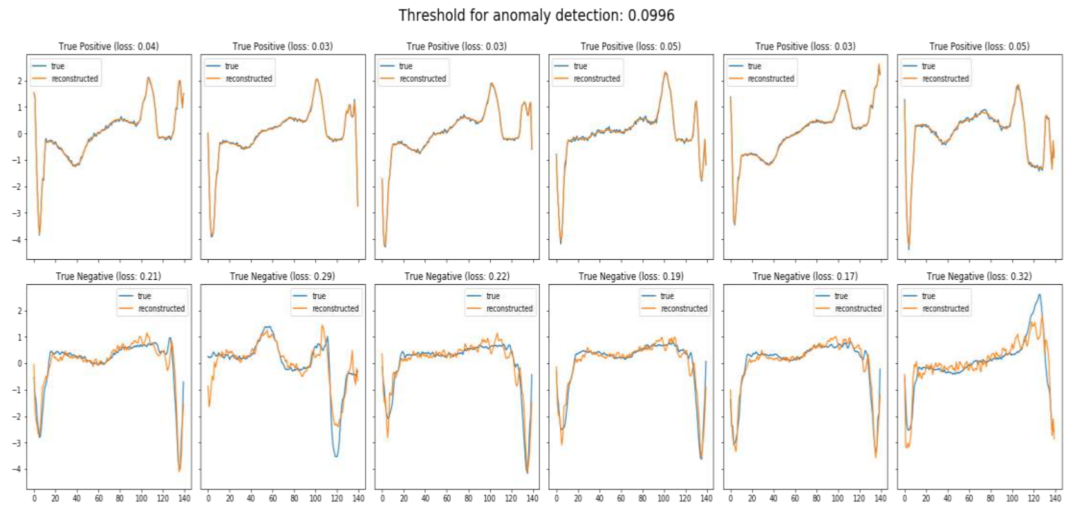

Figure 5. Reconstructed sequences for correctly predicted class for VAE LSTM.

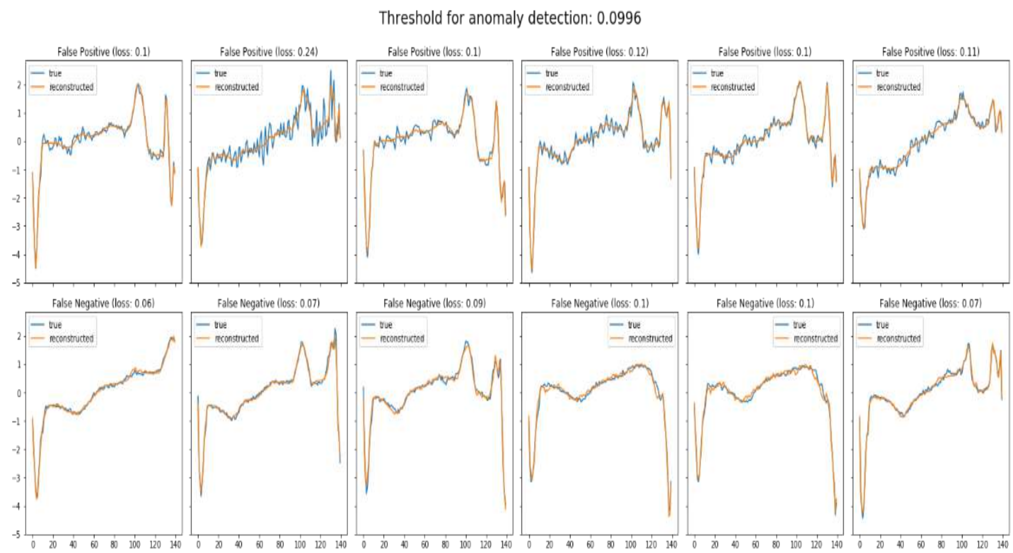

Figure 6. Reconstructed sequences for incorrectly predicted class for VAE LSTM.

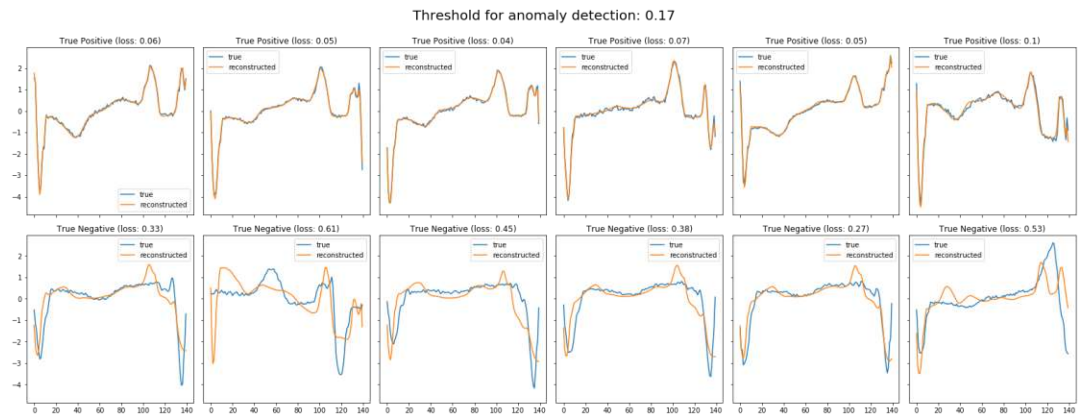

Figure 7. Reconstructed sequences for correctly predicted class for VAE Bi-LSTM.

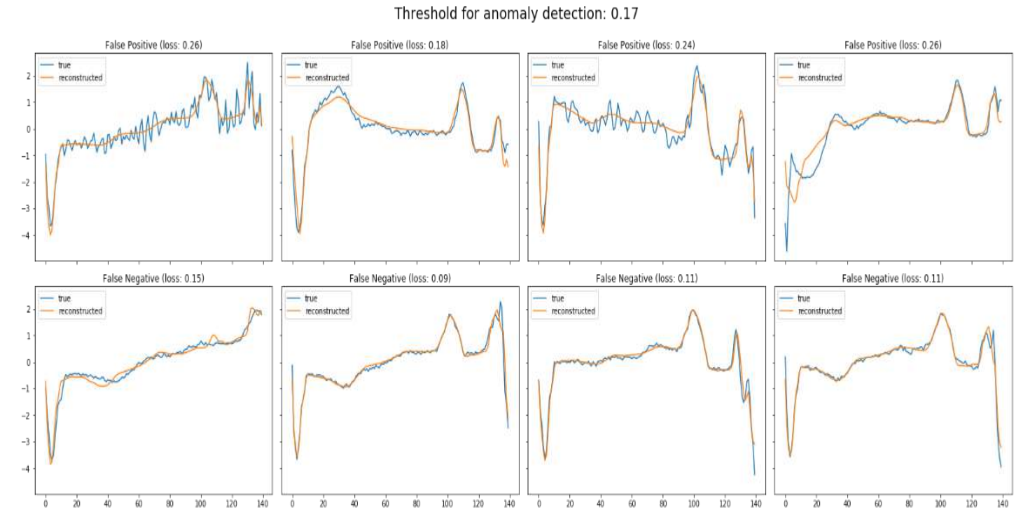

Figure 8. Reconstructed sequences for incorrectly predicted class for VAE Bi-LSTM.

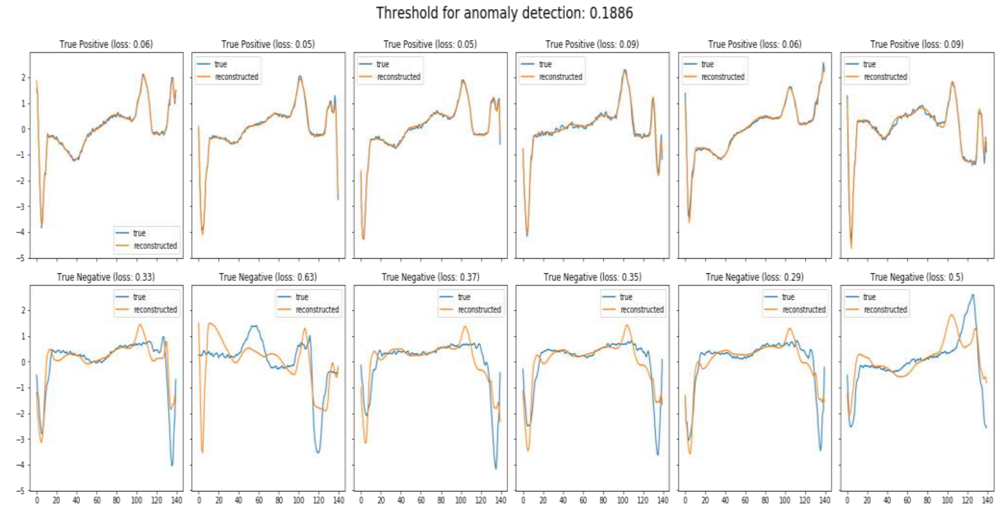

Figure 9. Reconstructed sequences for correctly predicted class for VAE Bi-GRU.

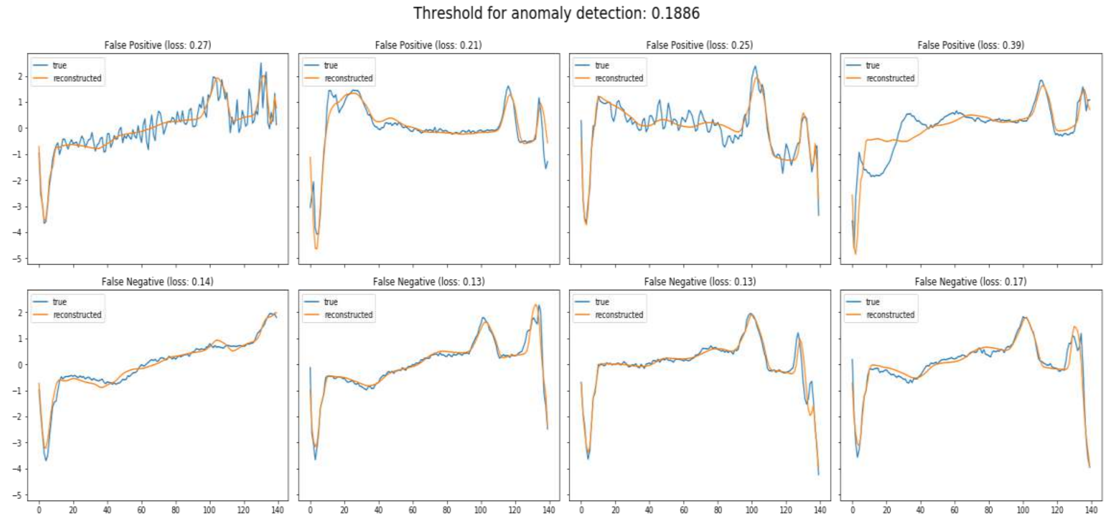

Figure 10. Reconstructed sequences for incorrectly predicted class for VAE Bi-GRU.

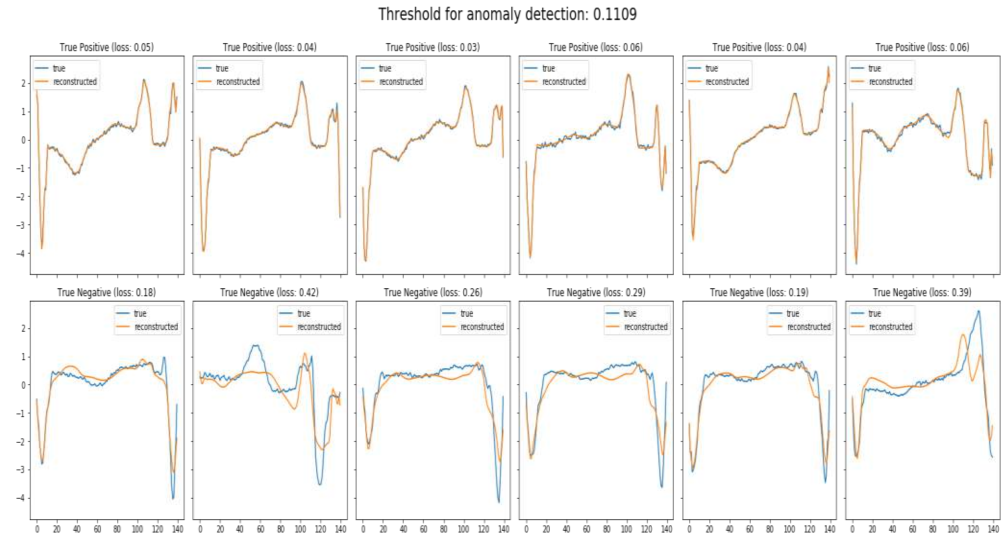

Figure 11. Reconstructed sequences for correctly predicted class for VrAE LSTM .

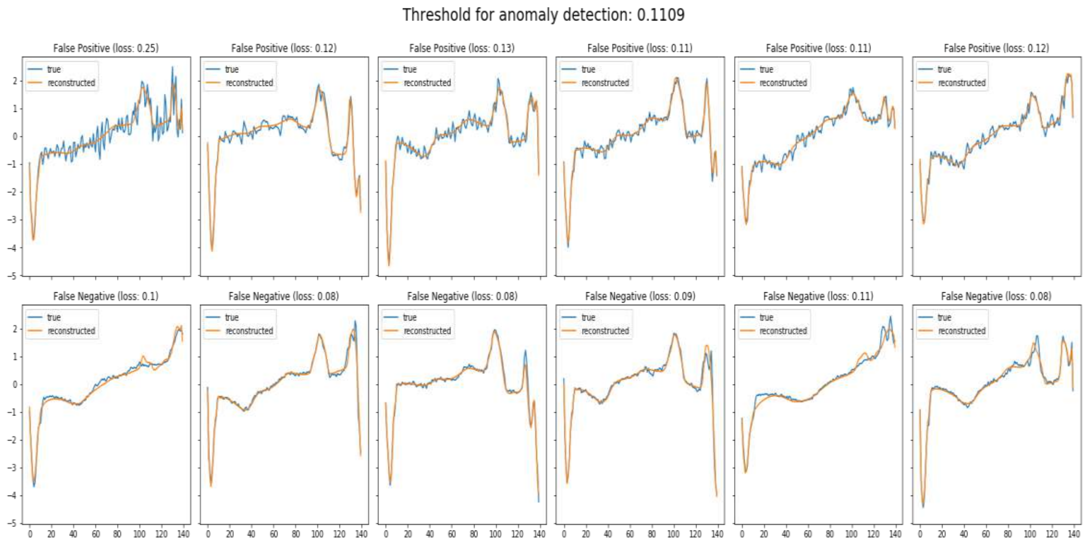

Figure 12. Reconstructed sequences for incorrectly predicted class for VrAE LSTM.

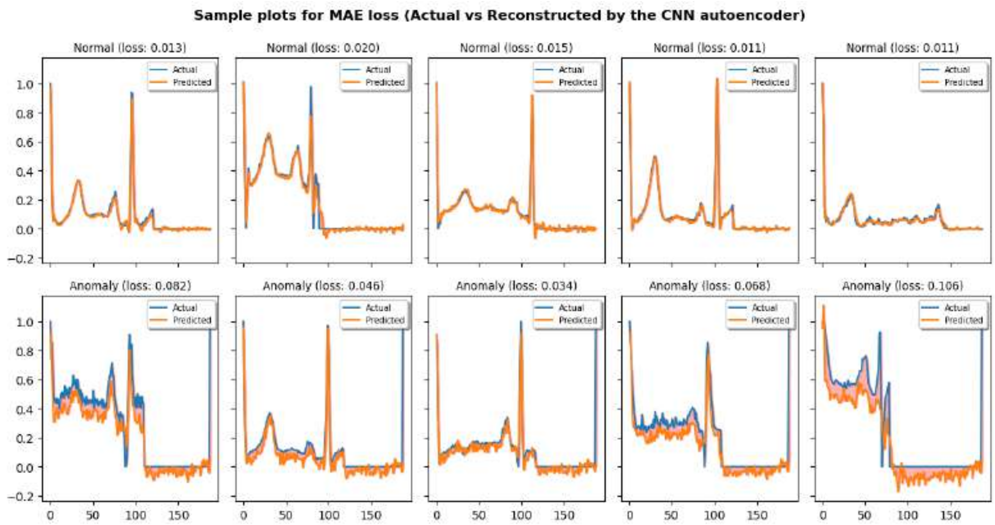

Figure 13. Reconstructed sequences for correctly predicted class CNN-AE.

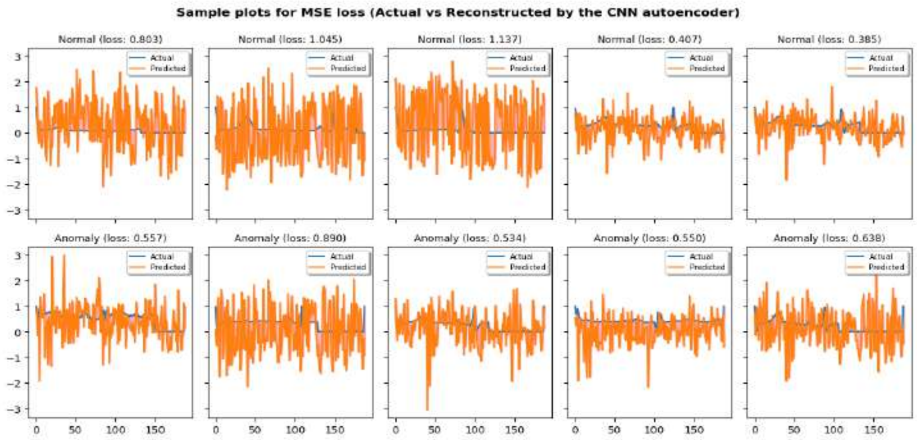

Figure 14. Reconstructed sequences for the incorrectly predicted class.

3.2 VrAE-BiGRU Results

In order to ensure and illustrate the effectiveness of the VAE with re-encoder, we carried out several tests on two kinds of time series datasets. Several variations of VAEs were built and tested during the process of building the model in Figure 15. The results of those models/variations are shown in Table 1 (The models named VrAE refer to VAE with re-encoder).

Table 1. Initial AUC for the tested models.

Model	Green Mobility	ECG
VAE LSTM	0.542.	0.928
VAE GRU	**0.563**	**0.991**
VAE Bi-LSTM	0.541	0.959
VAE Bi-GRU	0.482	0.992
VrAE LSTM	0.558	0.992
VrAE GRU	0.562	0.977
VrAE Bi-LSTM	0.559	0.996
VrAE Bi-GRU	**0.480**	**0.997**

The results for the ECG dataset are almost equally good for all the models. On the other hand, when the models were tested on the GM data, their performance was rather poor. Figures 15 and 16 illustrate the reconstruction of normal and abnormal GM data and ECG data, respectively, for the VrAE-BiGRU model.

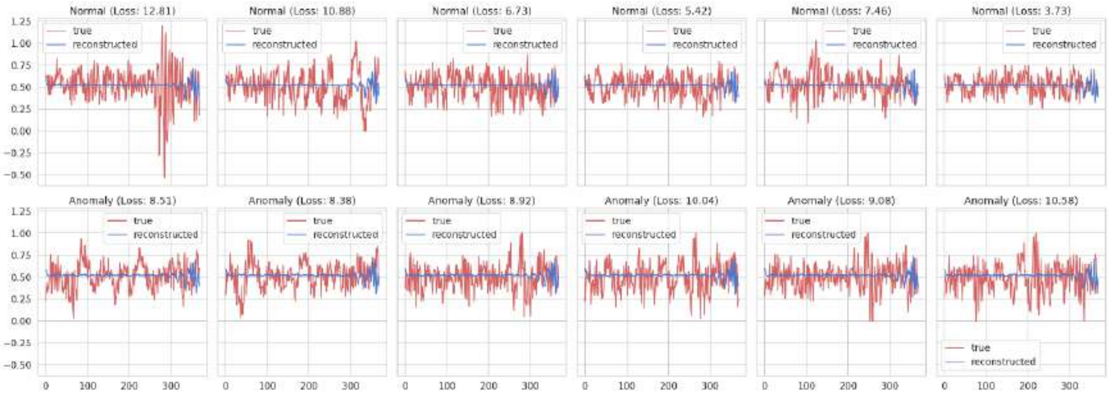

Figure 15. Reconstruction error for GM data for VrAE-BiGRU (Top: Normal, Bottom: Anomalies).

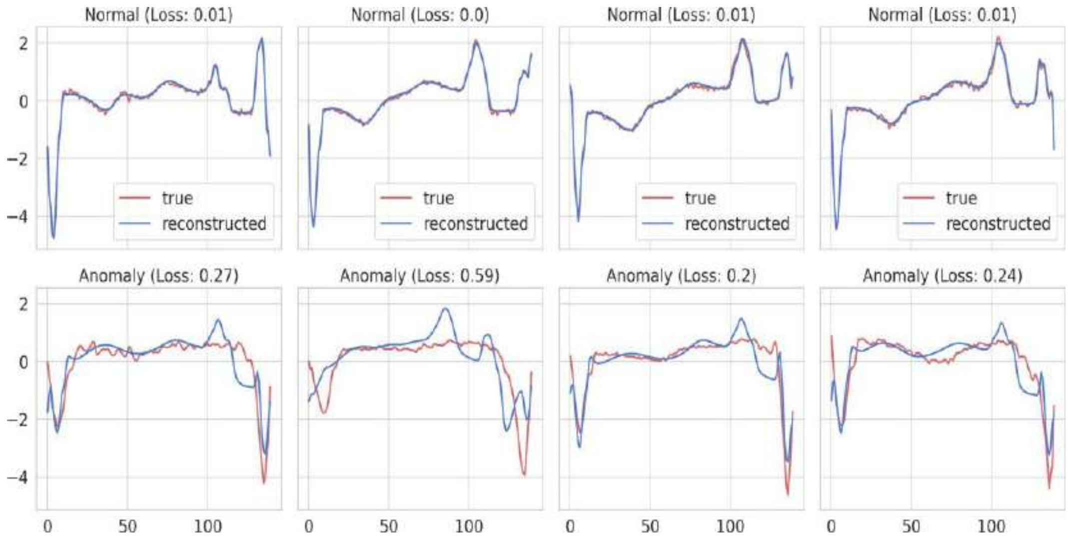

Figure 16. Reconstruction error for ECG data for VrAE- BiGRU (Top: Normal, Bottom: Anomalies).

Clearly, the normal data (Figure 16 top row) form similar patterns for the ECG data thus, the models could learn to reconstruct those patterns better. Additionally, in Figure 17 and Figure 18, we visualized how the latent space of the means is distributed along two PCA components.

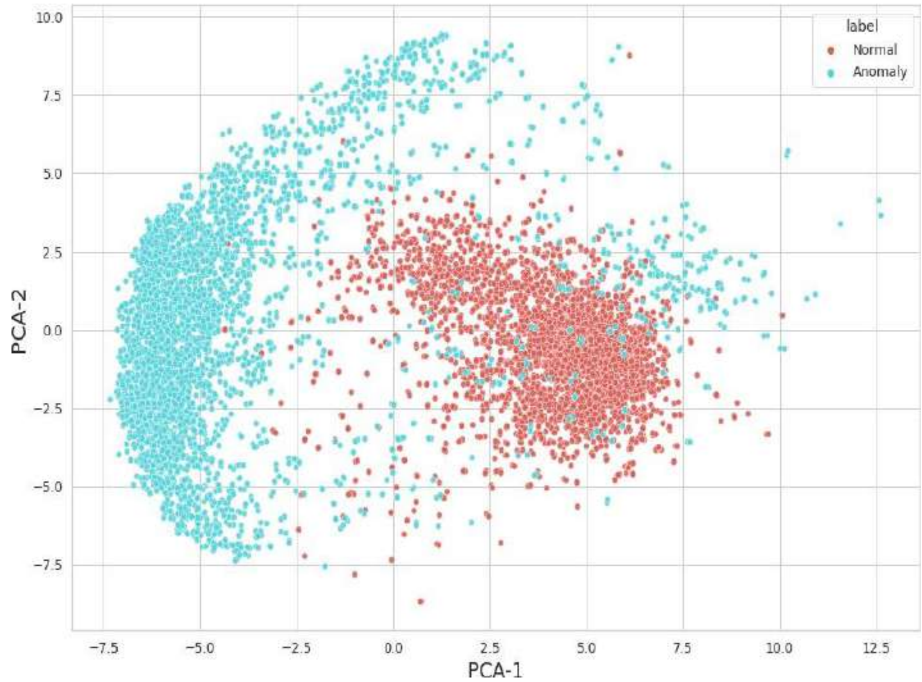

Figure 17. Projection of the latent space (with PCA) of re-Encoder for ECG data using VrAE-BiGRU.

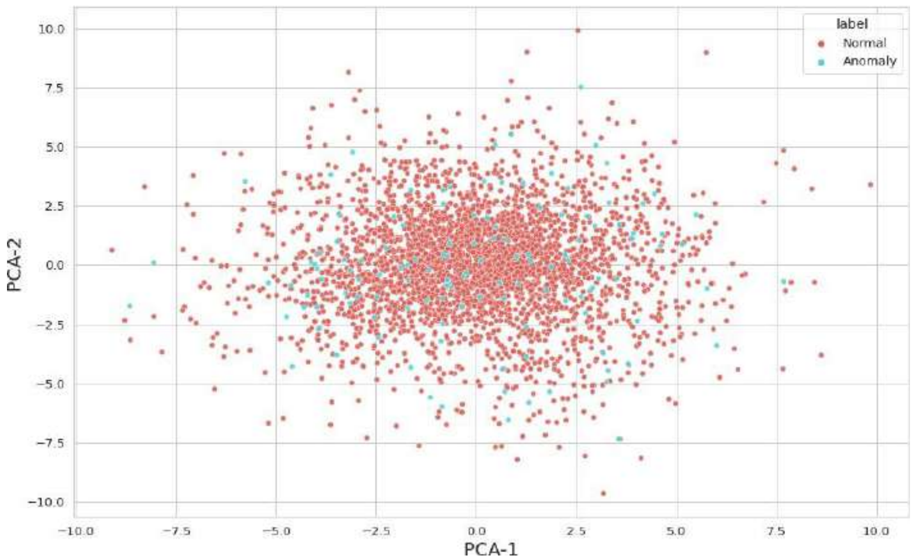

Figure 18. Projection of the latent space (with PCA) of re-encoder for GM data using VrAE-BiGRU.

We can see in the case of the ECG data there is some obvious clustering between the normal and the anomaly data. However, for the GM data it seems that this is not the case. The normal data have variations and noise that the model cannot easily learn to re-construct successfully. Also, the model reconstructs the main trends in the anomaly data rather well (Figure 15). In order to overcome the issue of poor performance on GM data, we used the Lilliefors test statistic to classify instead of mean squared error. The Lilliefors test is used to test whether data come from a normally distributed population when the null hypothesis does not specify which normal distribution [13]. In the present study, we subtract the original data from the predicted ones and perform the Lilliefors test on the residuals. In case the residuals formed a normal distribution (low test statistic) they were classified as normal or as anomaly otherwise. The AUC scores of the models after we implemented the statistical test are shown in Table 2.

Table 2. AUC scores with Lilliefor statistical test.

Model	Green Mobility
VAE LSTM	0.625
VAE GRU	0.701
VAE Bi-LSTM	0.609
VAE Bi-GRU	**0.754**
VrAE LSTM	0.646
VrAE GRU	0.706
VrAE Bi-LSTM	0.525
VrAE Bi-GRU	0.731

We can see that the AUC scores have increased for almost all the models. VAE-GRU, VAE-BiGRU and VrAE-BiGRU record a significant increase of the magnitude of 0.25 on average. On the other hand, the VrAE-GRU shows solely an increase of 0.019. The models with LSTM architecture show an increase of 0.079, while the VrAE-LSTM records a decrease of 0.029.

We suspect that this is due to the nature of the data, namely that the high-frequency oscillations from driving on a normal road can be seen as normally distributed noise, which is disrupted when the car travels over a rough patch. In Figure 18 we can see how the distribution of the reconstruction error for the normal and anomaly classes are highly overlapping. In contrast, the distribution of the Lilliefors' test statistic is more separated. (Figure 19) (Figure 20).

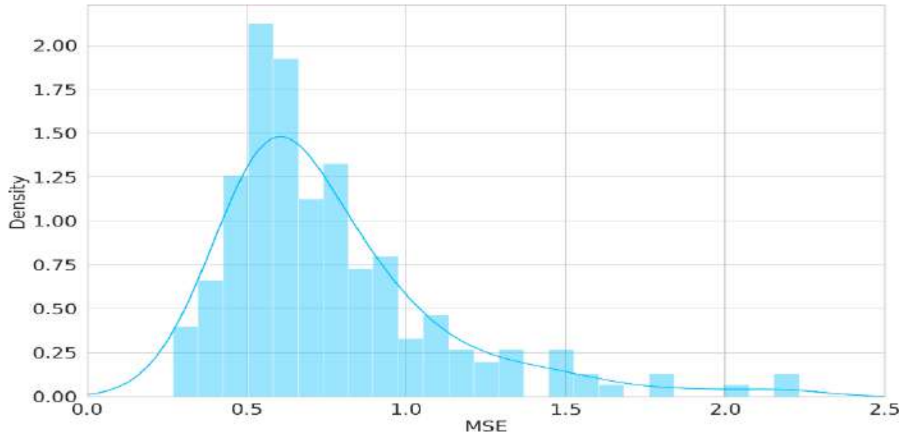

Figure 19. MSE distribution for Bi-GRU model on normal green mobility data.

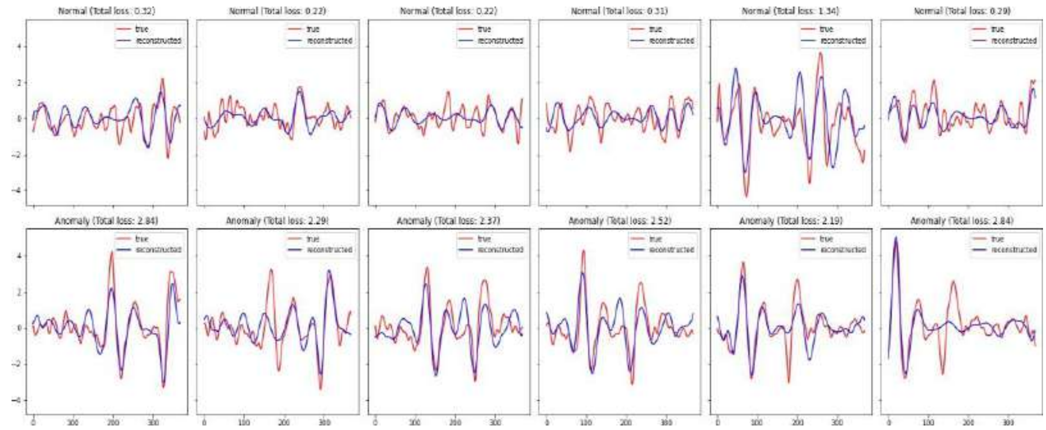

Figure 20. Reconstruction of de-noised green mobility data for VrAE-BiGRU.

4. Discussion

Throughout the paper, we tried to build a variation of the VELC model as described in. Both ECG and GM data were tested in numerous models/variations with LSTM and GRU architecture. The results for the ECG dataset recorded high and consistent AUC scores across all the models. The homogeneity of the normal/abnormal ECG data (see Figure 3) and lack of noise allowed the model to capture the normal and anomaly data very well. The GM data, on the other hand, recorded low AUC scores when using mean squared reconstruction error. The results, though are more or less consistent along the models. In order to improve the performance of the models in GM data, we applied another method of classifying normal and anomaly data. For that, we used the Lilliefors test. The results showed significant improvements in a few of the models (VrAE-BiGRU and VAE-BiGRU). In contrast, in others, the deviations of the AUC score were negligible compared to the initial results (VrAE-BiLSTM and VrAE GRU).

The overall results for the green mobility data indicate that our model has limitations, which could be due to a variety of reasons. From the plot in Figure 15, we observe that the data carry a lot of noise, which may not allow the model to capture the signal from the time series. We assume that if we remove the noise from the data, the model will extract solely the signal. Hence the model will learn better the underlying distribution. Figure 19 shows what data looks like after de-noise, and their corresponding AUC score is 0.638 (lillefors test) for VrAE-BiGRU.

5. Conclusion

Throughout the study, a VAE with a Re-encoder was chosen as the main model that could potentially provide the optimal solutions to fulfill the scope of the study. The outcome of the study, though, revealed that vanilla bidirectional GRU VAE has slightly better performance in detecting anomalies correctly. The limitations of the model, though, have a significant influence on the performance of all models, especially on the VrAE ones. Therefore, no certain conclusion can be made on which model is indeed the best. Future steps to ensure the consistent and robust performance of the models involve further data pre-processing, such as removing noise. Ex- tensive research to fully understand whether the implementation of the model has been applied correctly and how the constraint network could potentially affect the overall performance of the VrAEs compared to vanilla VAEs.

REFERENCES

[1] G. Zhu, H. Zhao, H. Liu, and H. Sun, "A novel LSTM-GAN algorithm for time series anomaly detection," in *Proc. 2019 Prognostics and System Health Management Conference (PHM-Qingdao)*, Qingdao, China, Oct. 2019, pp. 1-6.

[2] S. Schmidl, P. Wenig, and T. Papenbrock, "Anomaly detection in time series: a comprehensive evaluation," *Proc. VLDB Endowment*, vol. 15, no. 9, pp. 1779-1797, 2022.

[3] P. Liu, X. Sun, Y. Han, Z. He, W. Zhang, and C. Wu, "Arrhythmia classification of LSTM autoencoder based on time series anomaly detection," *Biomed. Signal Process. Control*, vol. 71, p. 103228, 2022.

[4] M. Roy, S. Majumder, A. Halder, and U. Biswas, "ECG-NET: A deep LSTM autoencoder for detecting anomalous ECG," *Eng. Appl. Artif. Intell.*, vol. 124, p. 106484, 2023.

[5] L. Goldberger et al., "PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215-e220, 2000.

[6] M. Thill, W. Konen, H. Wang, and T. Bäck, "Temporal convolutional autoencoder for unsupervised anomaly detection in time series," *Appl. Soft Comput.*, vol. 112, p. 107751, 2021.

[7] A. Pol, V. Berger, C. Germain, G. Cerminara, and M. Pierini, "Anomaly detection with conditional variational autoencoders," in *Proc. 2019 18th IEEE Int. Conf. Machine Learning Applications (ICMLA)*, pp. 1651-1657, 2019.

[8] J. An and S. Cho, "Variational autoencoder based anomaly detection using reconstruction probability," *Special Lecture on IE*, vol. 2, no. 1, pp. 1-18, 2015.

[9] D. Gong et al., "Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection," in *Proc. IEEE/CVF Int. Conf. Computer Vision*, 2019, pp. 1705-1714.

[10] C. Zhang, S. Li, H. Zhang, and Y. Chen, "VELC: A new variational autoencoder based model for time series anomaly detection," *arXiv preprint arXiv:1907.01702*, 2019.

[11] K. Dutta, R. Lenka, S. R. Nayak, A. Khandual, and A. K. Bhoi, "MED-NET: A novel approach to ECG anomaly detection using LSTM auto-encoders," *Int. J. Comput. Appl. Technol.*, vol. 65, no. 4, pp. 343-357, 2021.

[12] G. Li and J. J. Jung, "Deep learning for anomaly detection in multivariate time series: Approaches, applications, and challenges," *Inf. Fusion*, vol. 91, pp. 93-102, 2023.

[13] S. Maya, K. Ueno, and T. Nishikawa, "dLSTM: A new approach for anomaly detection using deep learning with delayed prediction," *Int. J. Data Sci. Anal.*, vol. 8, pp. 137-164, 2019.