Anomaly Detection in Autonomous Vehicle's Lidar Sensor Data Using Variational Autoencoders

Nourhane Sboui[1], Mohamed Hadded[1], Hakim Ghazzai[2], Mourad Elhadef[1], Gianluca Setti[2]
[1]College of Engineering, Abu Dhabi University, Abu Dhabi, United Arab Emirates
[2]CEMSE Division, King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia

***Abstract*—LiDAR sensor data is essential for autonomous vehicle navigation, traffic flow monitoring, obstacle detection, and passenger safety. However, the reliability of LiDAR data can be compromised by anomalies caused by sensor malfunctions, environmental conditions, or unexpected road events. To address this, detecting anomalies in spatial-temporal (ST) LiDAR data is critical for ensuring safety. This paper proposes a novel low-complexity unsupervised framework named CNN-BiLSTM VAE for anomaly detection (AD) in non-image LiDAR data. The framework combines variational auto-encoder (VAE) reconstruction, CNN for spatial learning, and bidirectional LSTM for time-series learning in a mirror-to-mirror (M2M) architecture. Experimental results show that this method effectively detects anomalies in multidimensional ST LiDAR data, thereby maintaining robustness under various environmental conditions.**

***Index Terms*—Autonomous Driving, Sensors, LiDAR Data, Anomaly Detection, Variational Auto-Encoder.**

I. Introduction and motivation

Light detection and ranging (LiDAR) sensors play a pivotal role, furnishing crucial data for autonomous vehicles to make real-time decisions and navigate safely through complex urban landscapes. As autonomous driving advances, the integration of LiDAR within intelligent transportation systems (ITS) has the potential to revolutionize transportation, bringing in a new age of safer, more efficient, and more sustainable alternatives to driving [1]. However, in the field of sensor data analysis, particularly with LiDAR data, detecting spatio-temporal (ST) anomalies is essential because the data are highly sensitive to environmental and technical abnormalities. These abnormalities, ranging from isolated outliers to systemic errors, can hide incidents or reveal significant data changes. For example, rain-induced water droplets may alter the trajectory of the laser beam, resulting in unwanted reflections and erroneous identification of obstacles [2].

As a result, reliable identification of these abnormalities is crucial to making more informed decisions and better understanding of the dynamic processes in ST dimensions for pre-processed LiDAR data specifically, which is the aim of this study. By effectively addressing these anomalies, we can harness the full potential of LiDAR sensors to advance various fields that rely on precise environmental observations. However, in such problem domains, acquiring labeled training, either raw or pre-processed LiDAR data, for all forms of anomalies, or even a subset of these anomalies, is extremely difficult and expensive, if not impossible [3]. This underscores the imperative for employing unsupervised approaches to detect anomalies within LiDAR data.

In recent years, a variety of machine learning and deep learning methods have been explored for outlier detection in ITS sensor data. Among these, we can cite ST outlier detection methods based on the density-based spatial clustering algorithm such as ST-DBSCAN [4] and LDBSCAN [5]. The latter offers a locality-based outlier detection method for unsupervised anomaly detection (AD) problems in spatial databases with noise. In these clustering methods, choosing the right number of clusters is very important and challenging. One-class classification (OCC) algorithms, such as one-class SVM (OC-SVM) [6], offer an alternative to unsupervised learning in AD by dividing test samples into a specific class, but their effectiveness depends on appropriate parametering.

However, with multivariate time series data where strong temporal dependencies between time steps can exist, these distance- and clustering-based approaches struggle to capture these dependencies accurately. Furthermore, high-dimensional data may suffer from data sparsity, making the definition of data locality ambiguous. Therefore, deep learning-based approaches have recently been proposed as alternative solutions for AD applications. Over the last decade, to protect AVs against sensor errors and security threats, Rezaei et al. [7], have proposed a detection technique based on GAN called GAAD. Thus, the authors have used a GAN-based auto-encoder (AE) method for detecting anomalies, building on a framework already proposed in the literature called GANomaly. They further extended the GANomaly architecture by incorporating an extra AE. The hypothesis is that the AE can refine the reconstructed output of the generator and correct errors introduced during the reconstruction of anomalous behavior. Similarly, Jin et al. [8] have exploited the advantages of combining camera data with the LiDAR point cloud to provide an AD framework that facilitates robust autonomous navigation against disturbances. The authors have defined three main components in this framework: (i) joint representation learning for camera and LiDAR data fusion using VAE, (ii) AD learning to identify anomalous readings; and (iii) anomaly reconstruction and navigation policy learning that help reduce the anomaly effects.

While the deep learning methods mentioned above have proven effective, a significant drawback is their reliance on labeled normal or abnormal data for training, thus classifying them as supervised or semi-supervised models. In this paper, we investigate the problem of AD in LiDAR sensor data from self-driving systems. A novel low-complex unsupervised method named CNN-BiLSTM VAE is developed. The propopsed approach combines the generative capabilities of variational auto-encoders (VAE) with the learning abilities of convolutional neural networks (CNN) for spatial features

and bidirectional long-short-term memory (BiLSTM) networks for temporal dependencies in a mirror-to-mirror (M2M) architecture. Our technique is tailored to multidimensional domains with continuous-valued features, which characterizes the pre-processed LiDAR data. The goal is to detect, in an unsupervised manner, the abnormal behavior and unusual ST patterns that can be caused by a variety of causes, particularly environmental factors. Specifically, the proposed approach operates in scenarios where no prior knowledge is available regarding the presence or proportion of noise and outliers within the real-world LiDAR data. The proposed combination of deep learning techniques seeks to tackle the challenges inherent in AD within pre-processed LiDAR data, where conventional approaches may prove inadequate. We evaluate the performance of the proposed framework against an existing hybrid deep learning method named MC-CNN-LSTM [9]. Through several experimental validations, we showcase the efficiency of our proposed method in detecting anomalies within multidimensional ST pre-processed LiDAR data, achieving superior testing accuracy improvement.

II. Proposed Hybrid Model

A. Anomaly Detection Problem Formulation

In the context of ST AD, we believe that VAE [10] presents a promising approach to learning latent representations of multidimensional, continuous-valued ST data. By encoding both spatial and temporal features into a lower-dimensional latent space, VAE can effectively capture the underlying structures and dependencies within the data. This latent space representation serves as a foundation for AD, as deviations from the learned normal behavior within the latent space can signal anomalies in the original data. Furthermore, the probabilistic nature of VAE enables uncertainty estimation, facilitating the differentiation between anomalies and normal data points. Thus, integrating VAE into AD frameworks for ST data not only enables robust and interpretable modeling of complex data patterns but also enhances the effectiveness of AD in real-world applications.

The proposed CNN-BiLSTM VAE is employed for AD by measuring the difference between the reconstructed and original data, with the reconstruction error (RE) serving as the anomaly metric, as shown in Fig. 1. Operating under an unsupervised learning objective, VAE aims to reconstruct the input data as output during training by minimizing a predefined loss function. Anomalies are identified based on data points with significant REs exceeding a predetermined threshold value [10]. In this section, we present a comprehensive overview of the complete AD process employing the proposed CNN-BiLSTM VAE framework along with its proposed architecture.

We denote the VAE encoder by E and the decoder by D. We also consider a m-dimensional ST $\mathcal{D}$ containing T sequences $\{X_t\}_{t=1}^{T}$, where each X_t represents a multidimensional ST sequence with a sequence length L. The objective is to capture the ST dependencies within the data, reconstruct the input sequences while optimizing a defined loss function, and finally detect anomalies that differ significantly from the normal reconstructed data with a high RE. The ST sequence X_t can be represented as a matrix:

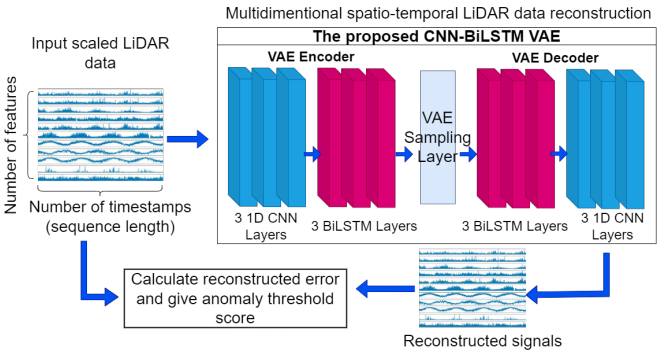

Fig. 1: Anomaly detection workflow using the proposed M2M CNN-BiLSTM VAE model.

$$X_t = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{L1} & x_{L2} & \cdots & x_{Lm} \end{pmatrix}, \tag{1}$$

where x_{ij} represents the value of the i-th time step and j-th feature or dimension. The proposed approach for training the CNN-BiLSTM VAE model consists of three main stages:

B. Data Normalization

At this stage, the multidimensional ST data set is processed in such a way that the proposed CNN-BiLSTM VAE network can exploit the spatial and temporal contexts jointly. The data set $\mathcal{D}$ is pre-processed as follows: First, relevant features are selected and standardized using the Robustscaler as follows:

$$X_{\text{t,scaled}} = \frac{X_t - \text{median}(X_t)}{\text{IQR}(X_t)}, \tag{2}$$

where $\text{IQR}(X_t)$ represents the interquartile range of X_t. Robust scaling techniques like Robustscaler are crucial for preprocessing multidimensional ST datasets with prevalent anomalies and outliers, as they are adept at handling such irregularities and preserving the underlying data distribution. These methods have lower susceptibility to outliers compared to other scaling approaches, adjusting data based on percentiles rather than being distorted by extreme values. Utilizing robust scaling is well-suited for anomaly identification tasks on ST data originating from real-world environments. After scaling the data, data sample sequences are created with a specified sequence length L.

C. LiDAR Data Reconstruction and Model Architecture

Traditional AD methods struggle with multidimensional data, especially in spatio-temporal (ST) AD scenarios where features are characterized by different distributions. In this paper, we introduce M2M architecture for the VAE as shown in Fig. 1 where the layers in the encoder and decoder are arranged in a mirrored fashion, with the encoder's layers encoding the input data and the decoder's layers decoding the latent representations back into the original input space. This symmetrical arrangement helps in learning a meaningful latent representation of the input data, which can then be used for various tasks such as reconstruction or generation.

1) VAE Encoder: The encoder consists of two primary components:

• **Three 1D CNN Layers (Spatial Features Encoding):** The input sequence data are processed using three 1D convolutional layers equipped with the "Swish" activation functions [11]. These CNN layers play a pivotal role in efficiently extracting spatial features from the input data, enabling the model to capture inherent spatial patterns and structures within the ST data. These layers also process the input data by utilizing filters with a kernel size of 3 and employing 'same' padding to maintain the input size. Note that we propose not using the max-pooling layers after each convolutional layer, which allows the encoder to retain more detailed spatial information for accurate analysis and interpretation.

• **Three BiLSTM Layers (Temporal Features Encoding):** The encoder integrates CNN layers followed by two BiLSTM layers to capture temporal dependencies within the data, processing input sequences in both forward and backward directions. Each BiLSTM layer comprises units equivalent to the batch size, ensuring comprehensive coverage of the entire sequence. Bidirectional processing allows BiLSTM layers to effectively capture long-range temporal dependencies critical to detecting anomalies unfolding over extended periods or involving complex temporal relationships.

The described encoder E, parameterized by θ_E, transforms the input sequence X_t into a latent space representation z_t, consisting of mean μ_t and logarithmic variance $\log(\sigma_t^2)$:

$$z_t = E_{\theta_E}(X_{t,\text{scaled}}) = \begin{pmatrix} \mu_t \\ \log(\sigma_t^2) \end{pmatrix}, \tag{3}$$

This proposed combined representation of CNN and BiLSTM layers within the encoder architecture serves as a foundation for generating latent representations in subsequent layers, which in turn enhances the model's ability to detect anomalies accurately with varying complexities and data distributions.

2) VAE Sampling Layer: Following the encoder architecture, a sampling layer computes the mean and variance of the encoder's output, serving as the latent space. This enables the generation of probabilistic latent representations by scaling and shifting random samples, integrated seamlessly into the model for end-to-end training. The probabilistic approach enhances the model's ability to capture complex patterns, provide uncertainty estimates, and distinguish anomalies from noise, thus preventing overfitting and improving robustness.

The latent space representation z_t is then sampled using the reparameterization trick: where $\epsilon \sim \mathcal{N}(0, 1)$.

$$\tilde{z}_t = \mu_t + \epsilon \cdot \log(\sigma_t^2), \tag{4}$$

3) VAE Decoder: As shown in the model architecture given in Fig. 1, the decoder mirrors the encoder. Therefore, it is responsible for reconstructing the input data from the latent space representations generated by the encoder. The output of the latent layer passes through a dense layer with a "Swish" activation function and a reshaping layer. These layers are used to increase the dimension of the output data to match the shape of the input data and to reshape it to match the input shape required by the subsequent BiLSTM layers utilized to reconstruct the temporal features of the input data. Followed by two 1D CNN layers to reconstruct the spatial features of the input data. The final output is generated by a final convolutional layer with a linear activation function for the reconstructed output data, where the number of filters is equal to the number of input features. The decoder D, parameterized by θ_D, reconstructs the latent space representation $\tilde{z}_t$ into the output sequence $\hat{X}_t$:

$$\hat{X}_t = D_{\theta_D}(\tilde{z}_t) = \begin{pmatrix} \hat{x}_{11} & \hat{x}_{12} & \cdots & \hat{x}_{1m} \\ \hat{x}_{21} & \hat{x}_{22} & \cdots & \hat{x}_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{x}_{L1} & \hat{x}_{L2} & \cdots & \hat{x}_{Lm} \end{pmatrix}, \tag{5}$$

where $\hat{x}_{ij}$ represents the reconstructed value of the i-th time step and j-th feature or dimension of the output sequence $\hat{X}_t$.

D. Anomaly Detection

At the third stage, the AD is conducted by computing the RE using the loss function, which serves as the anomaly score. Anomalous data points typically exhibit large REs due to their deviation from the subspace patterns present in the data. Consequently, the aggregated REs across the t-th ST sequence are utilized as the anomaly score for the VAE-based proposed framework. Any multidimensional ST data points or sub-sequences with notably high REs are classified as anomalies. Subsequently, the primary objective is to minimize the loss error during the reconstruction process of the input sequence $X_{\text{t,scaled}}$ and the reconstructed sequence $\hat{X}_t$. This loss is defined as the mean absolute error (MAE) and serves as the anomaly score in the AD phase, denoted L_{VAE}:

$$L_{\text{VAE}} = \frac{1}{L \times m} \sum_{i=1}^{L} \sum_{j=1}^{m} |x_{ij,scaled} - \tilde{x}_{ij}|. \tag{6}$$

where $x_{ij,scaled}$ represents the value of the i-th time step and j-th feature or dimension of the input sequence $X_{\text{t,scaled}}$.

From this metric, we drive another metric that helps assess the conformity of the reconstructed output sequence to the input sequence: the VAE model accuracy score (Acc) which is formulated as follows:

$$\text{Acc} = 1 - \frac{L_{\text{VAE}}}{\max(|X_{\text{t,scaled}}|)}, \tag{7}$$

where $\max(|X_{\text{t,scaled}}|)$ presents the maximum absolute value of the scaled input data. In this formulation, the accuracy metric is calculated, ensuring that higher accuracy values correspond to lower REs, indicating better performance of the VAE model in reconstructing the input sequences and consequently in detecting anomalies when it is closer to 1.

E. Statistical Anomaly Injection

In order to test the proposed CNN-BiLSTM VAE model's reliability and performance in the AD phase and to compare it with existing deep learning AD models, we propose to inject different forms of dynamic obstacles as environmental anomalies into a subset of our data to mimic real-world scenarios in the presence of some environmental factors that can alternate the LiDAR data. Dynamic obstacles in pre-processed LiDAR data can significantly impact autonomous vehicle decision-making and pose challenges to navigation

systems. These obstacles can alter vehicle trajectory, necessitate rapid identification and adjustments, and potentially lead to traffic disruptions or accidents. To simulate real-world scenarios where unexpected environmental factors may influence vehicle behavior, we inject dynamic obstacles into the data set using a costume function that randomizes attributes such as positions, and velocities within predefined ranges. This generates a specified number of obstacles, each characterized by unique spatial coordinates, dimensions, and velocities.

III. Experiments and Discussion

In this study, we utilize the Lyft motion prediction [12]; a self-driving vehicle containing over 1,000 hours of data. This was collected by a fleet of 20 autonomous vehicles along a fixed route in Palo Alto, California, over a four-month period. consists of 170,000 scenes, each 25 seconds long, totalling over 1,118 hours of logs collected by a fleet of self-driving vehicles driving along a fixed route. The sensors for perception include 3 LiDARs. The sensors are positioned as follows: one LiDAR is on the roof of the vehicle, and two LiDARs on the front bumper. The roof LiDAR has 64 channels and spins at 10 Hz, while the bumper LiDARs have 40 channels. The used dataset will be spitted into 70 percent for training 15 percent for validation and 15 percent for testing.

The proposed CNN-BiLSTM VAE framework is implemented using the Keras deep learning library in Python, running atop the TensorFlow back-end [13]. All hidden layers incorporate the "Swish" activation function, facilitating faster convergence. A linear activation function is utilized for the final CNN layer. Optimization is carried out using the ADAM optimizer algorithm. Training is conducted in mini-batches of size 16 for 50 epochs, with a learning rate set to 10^{-4} and a sequence length ($L = 20$). These optimized parameters for training epochs, batch size, learning rate, and activation function are determined through grid search.

In this part, we present the training, validation, and testing outcomes of our proposed CNN-BiLSTM VAE framework for unsupervised AD using Lyft motion prediction LiDAR data, comparing these results with existing AD detection methods. During training and validation, we assess the model's learning performance, reconstruction, and generalization abilities in handling multidimensional ST data with potential outliers, using the MAE as the reconstruction error metric, as illustrated in (6). For this evaluation, another hybrid deep AE-based framework, named MC-CNN-LSTM, is used in this evaluation [9], which is a fully unsupervised AD approach. This framework exhibits superior capabilities in representing arbitrary data distributions for AD compared to commonly used alternatives such as principal component analysis (PCA) [14] or matrix factorization. The MC-CNN-LSTM architecture integrates a multichannel CNN network for learning spatial structures and an LSTM network for capturing temporal patterns, addressing the unsupervised AD problem in non-image multivariate ST data that can be gathered from multiple types of sensors for different application domains. Also, this framework has demonstrated exceptional effectiveness, outperforming many clustering- and distance-based AD methods, particularly within multidimensional ST datasets.

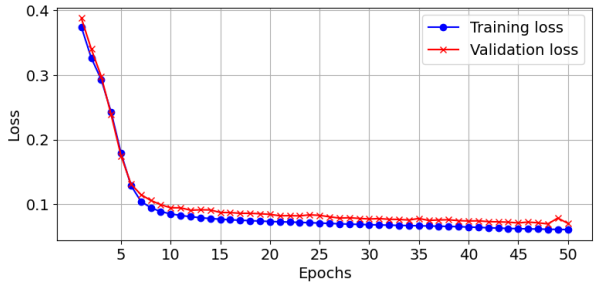

Fig. 2: Loss curves of the CNN-BiLSTM VAE proposed model.

TABLE I: Data structure parameters of the Layers of the CNN-BiLSTM VAE and MC-CNN-LSTM AE models

MC-CNN-LSTM AE	Value	CNN-BiLSTM VAE	Value
3D CNN Encoder	95,136	CNN_BiLSTM Encoder	12,592
ConvLSTM Decoder	197,376	CNN_BiLSTM Decoder	12,519
FCNN Layer	6,921		

During training phase, the result of the loss functions is illustrated in Fig. 2. Over the course of 50 epochs, the MAE values progressively decrease during both training and validation steps until reaching a minimum value of 0.075 for training and validation, presenting the final minimum difference between the input data and the reconstructed output. As a result, this value represents the reconstruction ability of the proposed CNN-BiLSTM VAE-based framework within LiDAR data. Table II also compares the reconstruction and learning performances of our proposed model which maximum accuracy of 0.951 compared to 0.235 as the minimum MAE and 0.852 as maximum accuracy for the MC-CNN-LSTM AE model. These results, obtained using the same preprocessed LiDAR data and evaluation metrics (Equations 6 and 7), highlight the superior performance of the CNN-BiLSTM VAE in data reconstruction. In addition, based on these results, we can claim that the proposed CNN-BiLSTM VAE AD model surpasses distance- and clustering-based algorithms, which rely basically on distance metrics that may inadequately capture temporal dependencies across various time steps. Additionally, it eliminates the need for dimensional reduction techniques, thereby preventing the loss of crucial information that could be significant. This superior performance becomes even more apparent when comparing the complexities of the two models, as illustrated in Tables I. In these tables, we present the detailed data structure complexities of the MC-CNN-LSTM AE and the proposed CNN-BiLSTM VAE models, respectively. Upon comparing the parameters in each layer, it becomes evident that our model achieves superior results during the reconstruction phase while maintaining a lower level of complexity than the MC-CNN-LSTM AE, with a total parameter set of 25,111 (98.09 KB) for the proposed CNN-BiLSTM VAE and 299,433 (1,170.39 KB) for the MC-CNN-LSTM AE.

During the testing phase, the metrics employed for evaluating the AD are precision (P), the F1 score (F1), which is a measure of the accuracy of a test, calculated as the harmonic mean of precision, and recall (R), also known as the true positive rate. As not all compared AD methods provide a mechanism for selecting anomaly thresholds, in

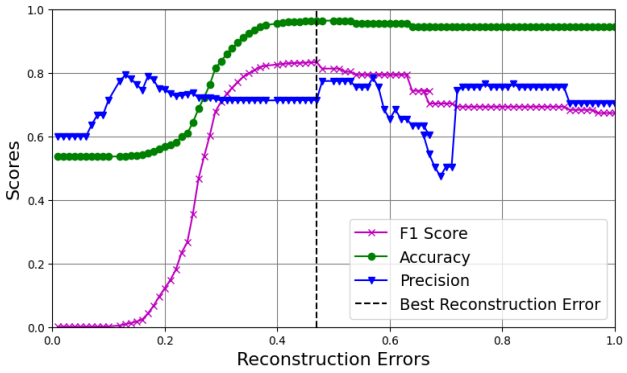

Fig. 3: F1-score, precision and accuracy values vs threshold REs of the proposed CNN-BiLSTM VAE model.

TABLE II: Comparison of the proposed CNN-BiLSTM VAE with the hybrid MC-CNN-LSTM AE

Models	MAE	Acc	Testing Acc.	F1	R	P
MC-CNN-LSTM AE	0.235	0.852	0.851	0.815	**0.988**	0.694
CNN-BiLSTM-VAE	**0.075**	**0.951**	**0.902**	**0.826**	0.787	**0.789**

this study, different anomaly thresholds for AE-based models were evaluated during the testing phase using a with 40 percent injected dynamic obstacles. The optimal threshold for the CNN-BiLSTM VAE model was determined to be 0.47, while the MC-CNN-LSTM AE model's optimal threshold was 1.25, based on the highest F1-score, precision, and accuracy. The evaluation revealed that the CNN-BiLSTM VAE model achieved the highest testing accuracy of 0.9021, with an optimal F1-score and precision at the 0.47 threshold. The results showed that our CNN-BiLSTM VAE model outperformed the MC-CNN-LSTM AE model, with an F1-score of 0.815, recall of 0.988, and precision of 0.694. The confusion matrix also demonstrated its effectiveness in detecting anomalies in LiDAR data, achieving an accuracy of 0.9021 but with a misclassification rate of 0.0979. These findings highlight the robustness and superior performance of the CNN-BiLSTM VAE model in AD in LiDAR data.

IV. Conclusion

In this paper, we have developed a robust AD method for pre-processed LiDAR data combining VAE architecture with CNN networks and BiLSTM networks to jointly handle spatial and temporal dependencies. This approach enhances the learning of latent representations, leading to accurate anomaly detection (AD). We evaluated our framework's performance on dynamically injected obstacle anomalies, demonstrating its effectiveness in detecting environmental distortions in pre-processed ITS sensor data. Comparative analysis with another established AD method highlighted our method's adaptability and robustness across different applications with similar data characteristics. As a future work, we will further enhance the performance of our framework by optimizing and fine-tuning the model and training on diverse multi-ITS sensor datasets encompassing various types of anomalies while considering computational speed to enable an ultra-fast AD approach for real-time smart mobility applications.

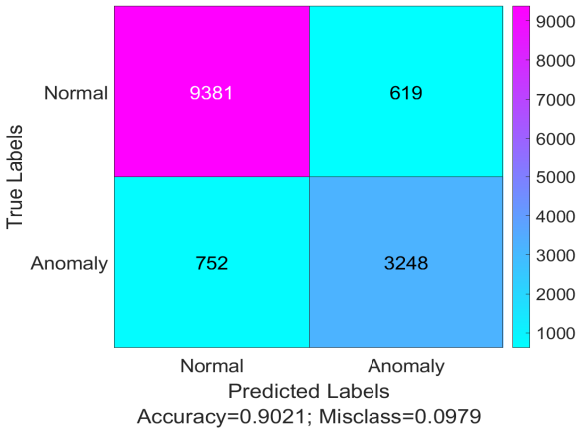

Fig. 4: CNN-BiLSTM VAE Confusion matrix results.

Acknowledgment

This work was supported by Abu Dhabi University Office of Research and Sponsored Programs under Grant 19300788.

References

[1] M. C. Lucic, H. Ghazzai, and Y. Massoud, "Elevated lidar placement under energy and throughput capacity constraints," in *IEEE International Midwest Symposium on Circuits and Systems (MWSCAS)*, Springfield, MA, USA, Aug. 2020.

[2] A. S. Mohammed, A. Amamou, F. K. Ayevide, S. Kelouwani, K. Agbossou, and N. Zioui, "The perception system of intelligent ground vehicles in all weather conditions: A systematic literature review," *Sensors*, vol. 20, no. 22, 2020.

[3] S. Baccari, M. Hadded, H. Ghazzai, H. Touati, and M. Elhadef, "Anomaly detection in connected and autonomous vehicles: A survey, analysis, and research challenges," *IEEE Access*, vol. 12, pp. 19 250–19 276, 2024.

[4] O. Nicolis, L. Delgado, B. Peralta, M. Díaz, and M. Chiodi, "Space-time clustering of seismic events in chile using st-dbscan-ev algorithm," *Environmental and Ecological Statistics*, 2024.

[5] L. Duan, L. Xu, F. Guo, J. Lee, and B. Yan, "A local-density based spatial clustering algorithm with noise," *Inf. Syst.*, vol. 32, pp. 978–986, 2007.

[6] K. Yang, S. Kpotufe, and N. Feamster, "An efficient one-class svm for anomaly detection in the internet of things," 2021.

[7] D. Bogdoll, E. Eisen, M. Nitsche, C. Scheib, and J. M. Zollner, "Multimodal detection of unknown objects on roads for autonomous driving," in *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Oct. 2022.

[8] Y. Zhang, H. Gao, X. Hu *et al.*, "Structure-based discovery and structural basis of a novel broad-spectrum natural product against the main protease of coronavirus," *Journal of Virology*, vol. 96, no. 1, p. e0125321, 2022.

[9] Y. Karadayı, M. N. Aydin, and A. S. Öğrenci, "A hybrid deep learning framework for unsupervised anomaly detection in multivariate spatio-temporal data," *Applied Sciences*, vol. 10, no. 15, 2020.

[10] F. Waseem, R. P. Martinez, and C. Wu, "Visual anomaly detection in video by variational autoencoder," *arXiv 2203.03872*, 2022.

[11] N. Ravikumar, A. Zakeri, Y. Xia, and A. F. Frangi, "Chapter 16 - deep learning fundamentals," in *Medical Image Analysis*, ser. The MICCAI Society book Series, A. F. Frangi, J. L. Prince, and M. Sonka, Eds. Academic Press, 2024, pp. 415–450.

[12] J. Houston and et al., "One thousand and one hours: Self-driving motion prediction dataset," in *Proceedings of the Conference on Robot Learning (CoRL 2021)*, ser. Proceedings of Machine Learning Research, vol. 155. PMLR, 2021.

[13] P. Schneider and F. Xhafa, "Chapter 8 - machine learning: Ml for ehealth systems," in *Anomaly Detection and Complex Event Processing over IoT Data Streams*, P. Schneider and F. Xhafa, Eds. Academic Press, 2022.

[14] A. Maćkiewicz and W. Ratajczak, "Principal components analysis (pca)," *Computers Geosciences*, vol. 19, no. 3, pp. 303–342, 1993.