Deep Learning and Recurrent Signature Based Classification for Sensor Based HAR: Addressing Explainability and Complexity in 5G Networks

[1]Karthikeyan R, [2]Usha S, [3] Dineshbabu V, [4]Jeena R, [5]Anitha Govindaram and [6]Jegatheesan A
[1]Department of Computer Science and Engineering, Artificial Intelligence and Data Analytics, Sri Ramachandra Institute of Higher Education and Research, Chennai, Tamil Nadu, India.
[2]Department of Computer Applications, Faculty of Science and Humanities, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India.
[3]Department of Information Technology, Karpagam Institute of Technology, Seerapalayam, Tamil Nadu, India
[4]Department of Computer Science and Engineering, Saveetha school of Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, Tamil Nadu, India.
[5,6] Institute of Computer Science and Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai, Tamil Nadu, India.
[1]karthikeyanr@sret.edu.in, [2] ushas6@srmist.edu.in, [3] dineshbabukit@gmail.com, [4] jeenar.sse@saveetha.com, [5]gani3086@gmail.com, [6] jegatheesanphdcse@gmail.com

Correspondence should be addressed to Karthikeyan R : karthikeyanr@sret.edu.in

Article Info
Journal of Machine and Computing (http://anapub.co.ke/journals/jmc/jmc.html)
Doi : https://doi.org/10.53759/7669/jmc202404098
Received 27 January 2024; Revised from 02 May 2024; Accepted 02 August 2024.
Available online 05 October 2024.

Abstract – When it comes to clinical applications, sensor-based human activity recognition (HAR) is invaluable, and numerous machine learning algorithms have effectively used to obtain excellent presentation. Using a variety of on-body sensors, these systems attempt to ascertain the subject's status relative to their immediate surroundings. There was a time when feature extraction was done by hand, but now more and more people are using Artificial Neural Networks (ANNs). A number of innovative approaches to HAR have surfaced since the advent of deep learning. Problems arise, however, for sensor-based HAR classification algorithms in today's communication networks. Among these, you can find solutions to problems like deal with complicated and large-scale data signals, extract characteristics from complicated datasets, and meet explainability standards. For complicated 5G networks, these difficulties become even more apparent. In particular, explainability is now critical for the broad use of sensor-based HAR in 5G networks and beyond. The research suggests a classification approach based on path signatures, recurrent signature (ReS), to address these issues. This cutting-edge model employs deep-learning (DL) approaches to circumvent the tedious feature selection challenge. Furthermore, the study investigates how to improve the ReS model's classification accuracy by using graph-based optimisation methods. To test how well the suggested framework worked, to dug deep into the publicly available dataset, which included a separate set of tasks. The paper's empirical results on AReM datasets achieved an average accuracy of 96%.

Keywords – Human Activity Recognition, Recurrent Signature, Graph-Based Optimization Techniques, Artificial Neural Network, Heterogeneous Sensors.

I. INTRODUCTION

The extensive use of activity recognition utilising wearable sensors in several industries, such as healthcare and the military, has made it a popular subject of research in recent years [1]. The sensor data provided by modern mobile devices is extensive and useful for applications such as activity recognition. Picture and video data, in addition to data collected from various sensors, have been used to identify human activities [2]. Time series gyroscopes and 3-axis accelerometers, is the primary emphasis of this letter since it is simple to collect, has a low data volume, and is straightforward to distribute online, all of which are necessary for recognising human activity [3]. The challenge of recognising different sorts of human activities from data collected by different sensors or from image and video sequences is called human activity recognition. The need to detect human actions in many fields—medicine, security, the smart home, monitoring, sports, etc.—has made it one of the most sought-after areas of research [4]. The ability to recognise abrupt changes in the activities of the old or unwell and respond accordingly is an essential requirement of

HAR. More accurate and dependable findings have been obtained thus far through the employment of a variety of methods [5]. Human activity recognition makes use of data collected from both images and sensors. The data gathered from various sensors included into smartphones accelerometers, gyroscopes, magnetometers, Global Positioning Systems (GPS), and more, can be described as time series data [6]. During the activity detection process, sensors gather low-level data, which is then processed using a number of algorithms to acquire higher-level information about the activities. Because sensor data is less bulky, easier to acquire, more accurate, and less likely to compromise privacy than image-based data, sensor-based HAR is gaining popularity [7].

Automated activity recognition using deep learning algorithms has recent research. How the various sensors, such as those for kinetics, vision, inertia, acoustics, pressure, strain, and physiological data, operate in tandem determines this method [8]. Wireless sensor networks (WSNs) are now a reality, thanks to advancements in wireless communications. These networks are known for their simplicity, low cost, light weight, and low power consumption [9]. A variety of environmental data can be collected by these tiny sensors. One subtype of these sensors is the wireless body area network (WBAN) sensor [10]. Wireless body area networks (WBANs) are tiny electronic sensors that people can wear on their bodies to track their movements and other vital signs like temperature, blood pressure, and heart rate. The internet of things, smart cities, healthcare, and sports are just a few of the numerous applications of body sensor networks [11].

This field presents a number of obstacles to investigation. Problems arise, first, when trying to manage the timing of the massive volumes of data collected by these sensors and then process it correctly [12]. The introduction of neural networks was a response to the increased workload and labour intensity of manually extracting features. In spite of lacking exhaustive information, Artificial Neural Networks are able to learn from experiences and examples and then generate results based on those learnings [13]. While more neurones improve a neural network's efficiency and performance, they may also make it more susceptible to overfitting and reduce its capacity for generalisation [14]. Recurrent Neural Networks (RNNs) like GRU and Long Short-Term Memory (LSTM) are examples of neural networks [15]. CNN variations include Inception and Residual Neural Network ResNet. When it comes to identifying complicated actions and abrupt changes in condition, current methods fall short. This has led to the necessity of using HAR in real-time and the introduction of new technology in this area [16]. The following is a synopsis of the research questions answered by this study:

1. What is the process for developing a DL model for human activity recognition?
2. How can to enhance feature extraction for data collected from multiple modalities of sensors?
3. How can the presentation of the model be confirmed?

In this study, to provide a deep model that can identify complicated actions with more efficiency and accuracy. In order to tackle the mentioned problems, to present a classification model termed recurrent signature (ReS) that uses path signatures to support XAI. Both the Verbesserung (Verbe) and the signature recurrent neural networks (SigRNN) are essential parts of ReS. Using 1D convolutional neural networks, the Verbe module augments data. It keeps the intrinsic flow features of the original data while improving it. One of the first feature-extraction modules to have XAI-enabled attributes is SigRNN. As a signature feature interpreter, it transforms time series into paths with several dimensions and calculates them. Through comparison with state-of-the-art approaches, extensive testing on public dataset confirm that ReS is superior.

Here is how the rest of paper is prepared: Section 2 provides a bibliography of relevant publications; Section 3 provides an in-depth explanation of the proposed technique; Section 4 discusses the analysis of the results; and Section 5 draws a conclusion.

II. RELATED WORKS

To train their model, Nguyen et al. [17] used unlabelled data from numerous time-synchronized sensors; however, for inference, they only needed data from a single sensor. To improve unimodal categorisation, to use contrastive learning to take advantage of multimodal correlation. When compared to training with a single sensor, virtual fusion produces far higher accuracy, and in certain instances, it even outperforms real fusion when tested with many sensors. Additionally, to broaden this approach to AFVF, which employs a subset of training sensors for inference, for a more generic version of the problem. Our approach attains state-of-the-art precision and F1 -score on the UCI-HAR besides PAMAP2 benchmark datasets. to can implement your request.

The novel AutoAugHAR, introduced by Zhou et al., [18], is an optimisation framework for data augmentation that uses a two-stage gradient approach. AutoAugHAR is built to consider the specifics of HAR duties, as well as the characteristics of potential augmentation procedures. The optimisation of the augmentation pipeline during HAR model training is impressive, especially considering that it does not significantly increase the training length. Tests on eight benchmark datasets based on inertial measurement units employing five HAR models showed that AutoAugHAR outperformed other top data augmentation frameworks in terms of robustness and efficacy. Thanks to its model-agnostic design, AutoAugHAR may be easily integrated with any HAR model without requiring any structural alterations. This is one of its most notable features. In addition, to show that AutoAugHAR is generalisable and extensible on four datasets from related but distinct areas. to will distribute it as an open-source tool1 and highly advocate it for use as a standard procedure in HAR model training.

Multiclass Autoencoder-based Active Learning (MAAL), developed by Park et al. [19], is an active learning approach that uses deep learning to learn latent representations by utilising the capabilities of Deep Support Vector). Connecting the HAR and selection models, MAAL uses a multiclass autoencoder to learn the typical properties of each activity class in latent space. This allows for an informative example selection during model training. to test our suggested MAAL on two open-source datasets. Up to 3.23% improvement in accuracy and 3.67% improvement in score were achieved during the overall active learning rounds, as shown in the performance results. In order to prove that the suggested MAAL is better than the other comparison methods, to also offer numerical consequences and an analysis of the sample selection process.

In order to improve environment-invariant activity classifications based on inertial measurement units (IMUs), Ray et al. [20] proposed techniques to transfer knowledge gained from active learning systems (ALS) to augment multi-modal and contrastive classifications. Our investigations on a real-world activity dataset show that ALS-HAR is very light-dependent, but that other HAR systems, such IMU-based classifiers, can still benefit from cross-modal information. In situations where ALS does not perform adequately, the extra information allows IMU-based classifiers to achieve an accuracy improvement of 4.2% and a macro F1 score improvement of 6.4%; in two of our three experimental settings, it even outperforms multi-modal sensor fusion models. Our research emphasises the unrealised possibilities of ALS integration in improving sensor-based HAR knowledge, which might lead to wearable activity recognition systems based on ALS that are both practical and efficient. These systems could find use in smart indoor environments, healthcare, and sports monitoring, among other fields.

An RMFSN, or residual multifeatured fusion shrinkage network, was suggested by Zeng et al., [21]. An enhanced residual network, the RMFSN incorporates a classifier network, a multi-branch architecture, and a channel). A 1D-CNN, a lightweight extraction approach are used by the special multi-branch framework to capture various activity features via numerous branches. While the classifier network generates the ultimate recognition results, the CASB is suggested to autonomously choose important features from the varied features for every action. For the public datasets UCI-HAR (98.13% accuracy), WISDM (98.35% accuracy), and OPPORTUNITY (93.89% accuracy), the suggested RMFSN has been demonstrated to be effective experimentally. With fewer model parameters needed, the suggested RMFSN could outperform current advanced approaches in terms of accuracy.

In order to identify human actions, Tehrani et al. [22] suggested and constructed a Bi-LSTM architecture, which is deep multilayer and bidirectional. Unlike conventional LSTM approaches, which only teach a single model, the Bi-LSTM strategy really presents two models: one to learn the input data sequence in one direction and another to learn the opposite sequence. Lastly, a fresh postprocessing method that enhances the average F1 score has been suggested, which is based on voting and windowing in the last phase. The suggested framework was tested extensively using three datasets, each of which had a unique collection of activities. The average F1 score for the three datasets used in this research—AReM, Mhealth, and PAMAP2—was 93.41%, 95.46%, and 95.79%, respectively. Additionally, the results showed that choosing the right window mechanism significantly affects the regular proportion of the F1 score.

III. PROPOSED METHODOLOGY

In this section, the research work tries to detect the sensor based HAR using deep learning model and that is publicized in **Fig 1**.

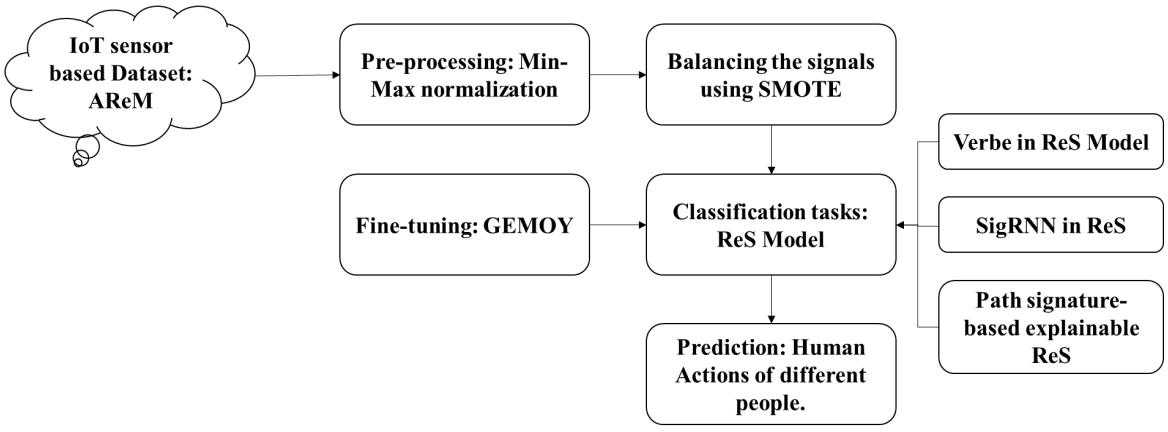

Fig 1. Workflow of the Proposed Methodology.

Characteristics of Datasets

Data can be collected in various ways. Particular nuances within each dataset will necessitate post-collection explanation. When it comes to activity detection, the most crucial aspects of a dataset are the quantity and variety of sensors, where they are located, the surrounding conditions, and the activities themselves. University College London's real-world

dataset of machine-learning human activity records was utilised to assess the suggested method's efficacy. Sensors worn by individuals were used to gather this dataset.

AReM Dataset

An activity detection scheme dataset utilising multisensor data fusion is relevant to our project [23]. The dataset is a collection of real-world criteria for activity recognition. Fifteen participants were used in a time series to classify data using WSNs. The user's motions trigger subtle adjustments to the wireless channel, which the activity detection system takes advantage of. Devices that measure the received signal strength (RSS) track the beacon packets that are sent and received on the WSN. An IRIS node programmed within a TinyOS-based Chipcon AT86RF230 radio subsystem, which supports the IEEE 802.15.4 standard, was used by RSS devices to gather the data. The subjects' left and right ankles, as well as their chests, were equipped with sensors that gathered this data.

Every fifty milliseconds, the sensors produced and transmitted a frequency. Due to the large amount of data, the dataset included frequency averages and variances computed every 250 ms. There were fifteen participants, and they cycled, sat, stood, bended twice, walked, and leaned back. There were 42,420 samples in the collection, with 480 sequences total. All actions were recorded for a duration of 2 minutes, or 120 seconds.

A virtual protocol operating in a 50-ms time step was used for communication through the exchange of beacon packets. When programming the node broadcasts, the token-passing protocol was utilised. In order to ensure that data gathering was secure and that nodes didn't accidentally send each other packets, the RSS was calculated whenever a node sent a packet. to took measures to reduce or remove background noise and frequencies when collecting data. When it comes to activity recognition software, the AReM dataset stands in for actual criteria. The datasetID.csv files include the RSS input flows; the ID is a unique number that recurs for every action taken. The use of wearable sensors allowed for the extraction of this dataset.

Preprocessing

Normalisation has been accomplished using the Min-Max approach. An arbitrary interval values is used to map each dataset in this simple method. Since it is easy to convert any interval to another interval, to will be transforming all features into the interval [0, 1]. This means that the values for feature variables will be 0 and 1, respectively.

Assume feature A is dataset among min_{A} and max_{A} to the min_{B} and max_{B}. The value v from the initial interval, for example, will be changed to v′ from the new interval as

$$v' = (v - min_A)\frac{newMax - newMin}{max_A - min_A} + (newMin) \tag{1}$$

Classification Using Recurrent Signature (ReS)

In this section, to introduce the projected ReS model for classification tasks.

ReS Explanation

Path P in R^{d} is from the intermission $[0,T]$ to R^{d}, which is written as $P{:}\,[0,T] \to R^{\mathrm{d}}$. If to use sensor signals as an example, to can think of the feature (such the average time it takes to send two data packets ahead) as a path Pt with a length of l. To be more precise,, $P_t = P_t^1 = mean_t$ and t undertake values from a set of discrete time steps $\{t_1, t_2, \ldots, t_l\} \in [0,T]$. NTS flows with multiple features can be considered paths, characterized as $P_t = \{P_t^1, P_t^2, \ldots, P_t^d\}$, $t \in [0,T]$. Each $P_t^k = \{x_{t_1}^k, x_{t_2}^k, \ldots, x_{t_l}^k\}$ signifies a feature, where $k \in \{1, 2, \ldots, d\}$. **Table 1** symbols used in ReS

Table 1. Meaning Of the Main Notations

Meaning	Notation
The k_{th}path;l is the measurement of a path	$p_k = \{x_{k_1}, x_{k_2}, \ldots, x_{k_l}\}$
A interval categorisation normalized in [0,1]	$t_k = \{t_{k_1}, t_{k_2}, \ldots, t_{k_l}\}$
V_k is the gathering of feature categorizations of p_k, v_k^i $(i \in [0, 1, \ldots, m])$ is the ith channel of V_k; channel size of V_k	$V_k = \{v_k^1, v_k^2, \ldots, v_k^m\}$
$a_k^{\theta_1}$ is sequences once Verbe; θ_1 is the limit of 1D-CNN in Verbe	$a_k^{\theta_1} = \{\{p_k\}, \{t_k\}, \{V_k = \{v_k^1, v_k^2, \ldots, v_k^m\}\}\}$
R_k is the throng of final feature structures of p_{k}; oc is sizes of RNN in SigRNN	$R_k = \{r_k^1, r_k^2, \ldots, r_k^{oc}\}$
s_k is group of slices yielded by window; K is the number of slices	$s_k = \{s_{k,1}, s_{k,2}, \ldots, s_{k,k}\}$
Sig_k^N is the N truncated alleyway moniker of s_{k}; N is truncation order	$Sig_k^N = \{Sig_{k,1}^N, Sig_{k,2}^N, \ldots, Sig_{k,K}^N\}$

To guarantee steady and quicker scales, the min-max normalisation is done to numerical feature vectors using ReS, a classification model based on XAI. to resample the data using one-sided synthetic minority oversampling technique

(SMOTE) [24] because, in real-world temporal datasets, the data is imbalanced. While SMOTE enhances model generalisability by adding synthetic minority samples, one-sided decreases overfitting. The second stage is to improve the data by merging the features extracted from the 1D-CNN model (called Verbe) with the time-series signals. Then, more explainable characteristics are iteratively extracted using SigRNN. Building a stream of streams through the enlarged windows is a crucial step. Lastly, a fully linked layer is used for feature organisation.

Verbe in ReS Model

In it is usual to reduce the path signature to the Nth-order, which results in the omission of higher-order details. Hence, it is essential to improve the data prior to computing the path signature. Pointwise augmentation is used to address this issue while preserving the data's streamlike quality. This allows for the capturing of relevant information from higher-order phrases all the way down to lower-order terms via a Nth-order truncated signature. Intuitively, let $F(x) = (x, f(x))$ and f be fixed feature- example $f(x) = x^2$. Assuming $x \in R$ Prior to augmentation, the data has a shape of l×1. With the help of pointwise augmentation, the form transforms into $l \times 2$. This, allowing of $F(x)$ instead of x, so that more relevant features can be extracted. But because every dataset is different, to need a feature-mapping function f that is distinct to each dataset. It takes a lot of time to choose the best value of f. Our preference is for f to be data-dependent instead of pre-fixed. Consequently, to train f using a DNN technique, making it learnable and labelling it as f^{θ}.

Let $p_k = \{x_{k_1}, x_{k_2}, \dots, x_{k_l}\}$, where $x_{k_i} R^d (i \in [1, l])$. First, pointwise painstaking. Let the feature mapping be denoted by $f^{\theta}: R^d \to R^m$. The removed given by

$$a_k^{\theta} = \left(f(x_{k_1}, \theta), f(x_{k_2}, \theta), \dots, f(x_{k_l}, \theta)\right) \tag{2}$$

where $f(x_{k_i}, \theta) \in R^m (i \in [1, l])$. In order to decrease parameters and improve nonlinearity, 1D-CNNs often take the place of FC layers for pointwise augmentation. Using a 1D-CNN with a kernel size of one allows for the replacement of FC layers. As scanned progressively along the time axis, the stream-like aspect of the data is preserved regardless of the kernel size. One would assume, on the surface, that for a size $kernal \in N$ and feature mapping $f^{\theta}: R^{d \times kernal} \to R^m$, the extracted features are expressed by

$$a_k^{\theta} = \left(f\left(x_{k_1}, x_{k_2}, \dots, x_{k_{(kernalt)}}\theta\right), f\left(x_{k_2}, x_{k_3}, \dots, x_{k_{(kernal+1)}}\theta\right)\right) \dots, f\left(x_{k_{(1-kernal+1)}}, x_{k_{(1-kernal+)}}\right) = \left(f_{conv}(p_k, \theta)\right) \tag{3}$$

In this case, $f\left(x_{ki}, x_{ki+1}, \dots, x_{k_{(1+kernal-1)}}, \theta\right), \in R^m$ (where $i \in [1, l - kernal + 1]$). As depicted in (3), the arrangement length shifts from $l\ to\ l - kernal + 1$. This enables us to extract flow, resultant in faster perfect training.

In order to traverse the complete feedforward network along the stream, to first use a 1D-CNN with a kernel one (see to (4)).

$$a_k^{\theta_{1,1}} = f_{conv}\left(p_k, \theta_{1,1}\right) = \sigma(W^1 * pk + b^1) \tag{4}$$

Parameter θ1 characterizes the weight limits that the learn. Exactly, $\theta_{1,1} = \{W_1, b_1\}$ refers to the limits of layer in the network besides W_1 stands for the CNN's convolutional kernel. The convolution process is represented by the symbol $\prod$, and the activation function ReLU() is represented by the symbol σ(). Furthermore, to accomplish pointwise augmentation and retrieve the extracted feature, an m-layer 1D-CNN network is incorporated. $a_k^{\theta_1}$, as shown in

$$\begin{aligned} a_k^{\theta_{1,2}} &= f_{conv}\left(a_k^{\theta_{1,1}}, \theta_{1,2}\right) = \sigma\left(W^2 * a_k^{\theta_{1,1}} + b^2\right), \dots, \\ a_k^{\theta_{1,m+1}} &= f_{conv}\left(a_k^{\theta_{1,m}}, \theta_{1,m+1}\right) = \left(W^{m+1} * a_k^{\theta_{1,m}} + b^{m+1}\right) \end{aligned} \tag{5}$$

Due to the feature $a_k^{\theta_1}$, to truncate the novel path p_{k} to align it with $a_k^{\theta_1}$. Additionally, to treat time $t_k = \{t_{k1}, t_{k2}, \dots, t_{klt}\}$ as a feature, where $t_{ki}(where\ i \in [1, lt])$ is in the range of [0,1], clear as $t_{ki} = \frac{1}{lt} \times i$. Finally, to concatenate p_k, t_k, and $a_k^{\theta_1}$ in order to acquire the verbe-extracted features along the channel dimension. The final form of the input data is $lt \times (m + 2)$, utilising m feature paths recorded by the CNN in addition to one original path and one time info path. So, to fix the data imbalance, Verbe adds more information to the original route.

SigRNN in ReS

In order to account for interruptions in the data stream, path names are usually applied during the last stage of feature extraction. It is only reasonable to think about reusing path signatures for better feature extraction, though, given their

strong feature extraction capabilities. Hence, to build a data flow stream of streams, to suggest an extended-window technique. It is critical to maintain the sequential and time-based characteristics of sensor-based categorisation jobs. Stream of streams is a method of creating a hierarchical stream by organising lower-level streams in a specific order. For instance, five groups of twenty packets each can be formed from one hundred packets throughout a single period. After sorting, the segments are combined into a five-stream higher-order stream, with 20 packets per stream. Our extended-window method keeps the starting location fixed while increasing the size, unlike the average distribution method. Each iteration adds a set length adjust length (al) to the initial window size (w), allowing us to widen the intervals. The simplified version of the stream, represented as a_k^(θ_1), is $a_k^{\theta_1} = \{a_{k1}, a_{k2}, \ldots, a_{k(lt)}\}$, where $a_{ki}(i \in [1, lt]) \in R^d$, to flows using

$$\begin{aligned} s_k &= \{s_{k,1}, s_{k,2}, \ldots, s_{k,K}\} \\ &= \{\{a_{k1}, a_{k2}, \ldots, a_{kw}\}, \{a_{k1}, a_{k2}, \ldots, a_{kw+al}\}, \ldots, \{a_{k1}, a_{k2}, \ldots, a_{k(it)}\}\} \end{aligned} \tag{6}$$

According to Chen's identity, the path signature's efficiency stays high even though the data volume increases due to this expansion method. In particular, to think about two-dimensional path segments: $P_1: [0, T_1] \rightarrow R^d and\ P_2: [T_1, T] \rightarrow R^d$. Computing the signature of the full path created by segments is a logical first step. Discovering Chen's identity proves that a signature may be easily calculated by

$$S(X * Y)_{0,T} = S(X)_{o,T_1} \otimes S(Y)_{T_1,T} \tag{7}$$

where $*$ represents the process of paths.

Next, to compute each path in s_k. Let $P: [0, T] \rightarrow R^d$ be a path with T > 0. For any optimistic integer $k \geq 1$ and $i_1, i_2, \ldots, i_k \in \{1, 2, \ldots, d\}$, the kth- numeral of path P is distinct by

$$S(P)_{0,T}^{i_1,\ldots,i_k} = \int_{0<t_k<T} \ldots \int_{0<t_1<t_2} dP_{t_1}^{i_1} \ldots, dP_k^{i_k} \tag{8}$$

The Nth- name of sk, signified by $Sig_{k,i}^N = \left(Sig_{k,1}^N, Sig_{k,2}^N, \ldots, Sig_{k,K}^N\right)$ is derived using

$$Sig_{k,i}^N = \left(1, S(P_i)_{0,T}^1, \ldots, S(P_i)_{0,T}^d, \ldots, S(P_i)_{0,T}^{1,\ldots,1}, \ldots, S(P_i)_{0,T}^{\overbrace{d,\ldots,d}^{N}},\ i \in [1, k]\right) \tag{9}$$

If $d > 1$, then $Sig_{k,1}^N (i \in [1, K]) \in R^{(d^{N+1}-1)/(N-1)}$, and $Sig_{k,1}^N \in R^{K \times (d^{N+1}-1)/(N-1)}$. If $d = 1$ and $Sig_k^N \in R^{K \times k}$. It stream, Sig_k^N is a vector of length 1 count of $(d^{N+1} - 1)/(N - 1)$. The data in Sig_k^N is no longer a stream of streams, Sig_k^N continues to behave like a stream, making it possible to use path signatures for feature extraction again.

And lastly, compared to CNNs, RNNs are better at capturing temporal dependencies since their recurrent connections keep information over time steps. Contrarily, convolutional neural networks (CNNs) are optimised for spatial analysis and do not have sequential memory. A regular RNN combined with path signatures is adequate for feature extraction in an NTS since it usually does not rely on anything for the long term. Therefore, to further extract temporal information, to use a three-layer RNN (see equations (10) and (11) for details).

$$h_{k,t} = \sigma\left(U^1 a_k^{\theta_1} + W^1 h_{k,t-1}\right) \tag{10}$$

$$R_k^{\theta_{2,1}} = f_{rnn}\left(a_k^{\theta_1}, \theta_{2,1}\right) = \sigma\left(V^1 h_{k,t}\right) \tag{11}$$

where $\theta_{2,1} = \{U_1, W_1, V_1\}$ The first hidden parameters (θ2) will be learnt in the second ReS group. The layer, besides $\sigma()$ characterises the activation function. By echoing (10) and (11) three times, to acquire the structures $R_k^{\theta_2}$ was taken from the initial layer of SigRNN. Finally, to get the feature order to were after by feeding it into SigRNN as a new path and doing it all over again, this time three times. $R_k^{\theta_4}$.

Classification

The final topographies SigRNN are approved to layer to get the forecast standards for diverse classes [25], as exposed in

$$out = f_{dense}\left(R_k^{\theta_4}, \theta_5\right) = W R_k^{\theta_4} + b \tag{12}$$

where $\theta_5 = W, b$ characterizes the limits of the FC layer in the ReS system.

Let F^{θ_1} characterize the method of Verbe, W denote the edifice of streams, R^{θ_i} (for $i \in [2,4]$) characterize the RNN layers, and f exemplifies coupled layers. The full process of ReS is represented by

$$[P_1, P_2, \dots, P_n] = p_k\left(F^{\theta_1} \otimes WSig^N \otimes R^{\theta_2} \otimes \dots \otimes W \otimes Sig^N \otimes R^{\theta_4} \otimes f\right) \tag{13}$$

To avoid overfitting, a multi-class function is used, along with L2 regularisation. Equation (14) shows the multi-classification function.

$$L = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} y_i^k log\left(\hat{y}_i^k\right) + \frac{\lambda}{2N}\sum_{i=1}^{N} w_i^2 \tag{14}$$

where N is the total illustration count, besides K is count. The term y_i^k denotes the ith sample and $\hat{y}_i^k$ the weight parameters are denoted by w, while the L2 regularisation penalty coefficient is represented by λ.

Path Signature-Based Understandable ReS Model

Our ReS sensor-based classification model is explainable because to combined DNN with the path signature. Understanding the model's decision-making process is improved by examining the signature characteristics extracted by SigRNN. Here is a brief summary of ReSig's explainability:

Intuitiveness of Path Representation

To establish continuous pathways from discrete data streams, to use piecewise linear interpolation. In order to visually represent the evolution of a time series, SigRNN converts aθ k1 into pathways. By examining the pathways' geometric forms fluctuations (such as slopes and inflection points), one might derive an intuitive interpretation for the data.

Explainability of Feature Vectors

SigRNN generates feature that capture a time series' varied properties. Mean, standard deviation, and other statistical measurements and descriptive aspects like trends and periodicity are examples of the types of path features that relate to each dimension. By examining these vectors, to can better comprehend time-series categorisation and find the most important aspects.

Geometric Explanation

For topographies $s_{k,1} = \{a_{k1}, a_{k2}, \dots, a_{kw}\}$ in (5), to concept path A. Assume that path A is 2-D, signified as $A: [T_1, T_w] \rightarrow R^2$. According to (8), the 1st-order path A can be spoken as $[S(A)^1_{T_1,Tw}, S(A)^2_{T_1,Tw} = \left[A^1_{T_w} - A^1_{T_1}, A^2_{T_w} - A^2_{T_1}\right]$. This hint at that the 1st-order path cross of A characterizes the increases in A along the over interval $[T_1, T_w]$. This concept can also be extended order is defined in (15)–(18):

$$S(A)^{1,1}_{T_1,T_w} = \frac{1}{2!}\left(A^1_{T_w} - A^1_{T_1}\right)^2 \tag{15}$$

$$S(A)^{2,2}_{T_i,T_w} = \frac{1}{2!}\left(A^1_{T_w} - A^1_{T_1}\right)^2 \tag{16}$$

$$S(A)^{1,1}_{T_1,T_w} = \int_{T_1<t_2<T_w}\int_{T_1<t_2<T_w} d0A^{i_1}_{t_1} dA^{i_2}_{t_2} \tag{17}$$

$$S(A)^{2,1}_{T_1,T_w} = \int_{T_1<t_1<T_w}\int_{T_1<t_2<T_w} dA^{i_2}_{t_1} dA^{i_1}_{t_2} \tag{18}$$

It is relative to the square dimension over the interval that is, as mentioned in (15) and (16). $[T_1, T_w]$. The line signifies path A, orange to $S(A)^{1,1}_{T_1,T_w}$ and $S(A)^{2,1}_{T_1,T_w}$, individually.

Fine-Tuning Using Graph-Based Exploration for Mining Besides Optimization of Yields (GEMOY Technique)

To need to integrate parts of graph theory, optimisation algorithms, to create a new mathematical model for graph-based exploration that optimises mining and other firms' returns. Representing the relevant complex systems, optimising the allocation of parameters to the ReS model, and improving overall efficiency will be the main goals of this formulation.

Graph Representation

Define a graph $G = V, E$ where:

- V is the set of vertices representing entities (e.g., sensor fields).
- E is the set of edges representing relationships or interactions (e.g., transportation routes, dependencies).

Each edge $e \in E$ has an associated weight $w(e)$ which can represent cost, distance, or capacity.

Objective Function
Let x_i be the yield or output at node $i \in V$. The primary objective is to maximize the total yield while minimizing costs and other constraints.

$$Maximize\ Z = \sum_{i \in V} x_i - \sum_{e \in E} w(e).f(e) \tag{19}$$

Where $f(e)$ is the flow or utilization of edge e.

Constraints
Resource Constraints

$$\sum_{j \in \mathcal{N}(i)} f(i,j) \leq R_i, \forall i \in V \tag{20}$$

Where $\mathcal{N}(i)$ is node i and R_i is the resource limit at node i.

Capacity Constraints

$$0 \leq f(e) \leq c(e), \forall e \in E \tag{21}$$

Where $c(e)$ is the capacity of edge e.

Flow Conservation (for directed graphs)

$$\sum_{j \in \mathcal{N}^+(i)} f(i,j) - \sum_{j \in \mathcal{N}^-(i)} f(i,j) = b_i, \forall i \in V \tag{22}$$

Where $\mathcal{N}^+(i)$ and $\mathcal{N}^-(i)$ are the sets of outgoing and incoming neighbors of node i, respectively, and b_i is the supply or demand at node i.

Graph-Based Algorithms

1) Shortest Path with Yield Maximization

$$Minimize \sum_{e \in P} w(e) \tag{23}$$

$Subject\ to$:
$P = \{path\ from\ source\ to\ destination\}$
$maximizing \sum_{i \in P} x_i$

2) Max Flow with Yield Optimization

$$maximizing \sum_{i \in V} x_i \tag{24}$$

$Subject\ to$:

$$\sum_{i \in E} f(e) \leq \sum_{e \in E} c(e), f(e) \geq 0,\ \forall e \in E \tag{25}$$

3) Node and Edge Weight Updates using Machine Learning
Implement a Graph Neural Network (GNN) to predict yields and update weight

$$h_v^{(k)} = \sigma\left(W^k.AGGREGATE\left(\left\{h_u^{(k-1)}: u \in \mathcal{N}(v)\right\}\right)\right) \tag{26}$$

Where $h_v^{(k)}$ is the feature node v at layer k, W^{k} is the weight matrix, and σ is an activation function.

Optimization Technique Integration
Linear Programming (LP)
Formulate the yield optimization as a linear program:

$$Maximize\ c^T x \tag{27}$$

Subject to
$Ax \leq b,\ x \geq 0$

Genetic Algorithms (GA)
Initialize a population of possible solutions.
Evaluate fitness based on the objective function Z.
Apply crossover and mutation operations to change the population toward optimal solutions.

Reinforcement Learning (RL)
Define a reward function $R(s,a)$ that incorporates yield optimization and resource costs. Update the policy $\pi(a|s)$ based on the reward feedback:

$$\pi(a|s) = \pi(a|s) + a(R(s,a) + \gamma_{a'}^{max} Q(s',a') - Q(s,a)) \quad (28)$$

IV. RESULTS AND DISCUSSION

The trials are conducted on a PC with an $Intel\ Core\ i5-7200\ CPU, 8\ GB\ of\ RAM$, besides a processing speed of 2.7 GHz. Jupyter Notebook (Python 3.7) Environment and Windows 10, a 64-bit OS, are utilised to execute the processes. By manipulating the training and testing data shown in **Tables 2** and **3**, the experimental analysis of the planned model is conducted using existing approaches. Visual representations are provided in **Fig 2 to 3**, and the study activity involves implementing the basic models on our considered datasets and averaging the outcomes. The existing models use different datasets.

Table 2. Comparative Analysis of proposed model with existing models for 80%-20%

Classifiers	Accuracy	Sensitivity	Specificity	Precision	F-Score
AE	0.892	0.865	0.375	0.917	0.940
ANN	0.914	0.92	0.643	0.929	0.949
DBN	0.92	0.94	0.75	0.929	0.949
CNN	0.924	0.96	0.8	0.929	0.949
LSTM	0.931	0.952	0.833	0.933	0.952
RNN	0.943	0.968	0.917	0.923	0.945
ReS- GEMOY	0.962	0.971	0.945	0.968	0.969

The proposed model outperformed existing models by 80%-20% in the AE procedure, with an accuracy at 0.892, sensitivity of 0.865, specificity of 0.375, 0.917, and 0.940, respectively. The ANN technique achieved accuracy of 0.914, sensitivity at 0.92, specificity of 0.643, precision of 0.929, besides f1-score at 0.949, respectively. The DBN technique accomplished an accuracy of 0.92, specificity of 0.94, specificity of 0.75, precision of 0.929, besides f1-score of 0.949. The CNN technique achieved accuracy of 0.924, specificity of 0.96, specificity of 0.8, precision of 0.929, and a f1-score of 0.949. The LSTM technique accomplished an accuracy of 0.931, specificity of 0.952, precision of 0.833, and f1-score of 0.952. The RNN technique achieved accuracy of 0.943, specificity of 0.968, precision of 0.923, and f1-score of 0.945, respectively. The ReS-GEMOY technique achieved accuracy of 0.962, specificity of 0.97, precision of 1 (0.945), and f1-score of 0.969.

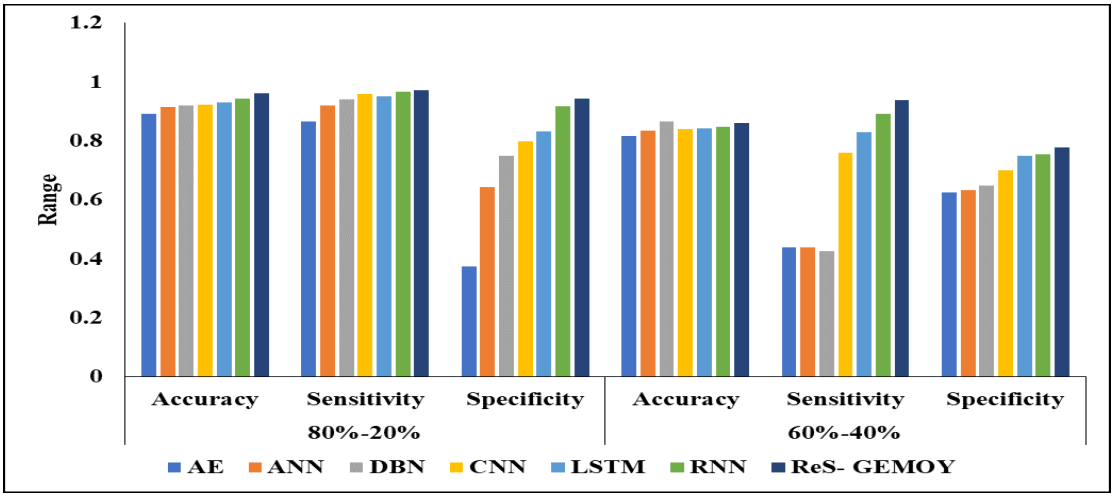

Fig 2. Comparative Analysis of Proposed Model on Different Metrics.

Table 3. Validation Analysis of Different Models on 60%-40%

Classifiers	Accuracy	Sensitivity	Specificity	Precision	F-Score
AE	0.818	0.439	0.625	0.658	0.526
ANN	0.835	0.44	0.634	0.663	0.529
DBN	0.867	0.427	0.649	0.669	0.521
CNN	0.839	0.76	0.7	0.730	0.752
LSTM	0.843	0.83	0.75	0.800	0.860
RNN	0.849	0.893	0.756	0.808	0.866
ReS-GEMOY	0.86	0.938	0.778	0.818	0.874

In the Validation analysis of different models on 60%-40% as in the AE classifier technique accuracy as 0.818 also sensitivity of 0.439 also specificity as 0.625 also precision as 0.658 and f1-score as 0.526 similarly. Then the ANN technique accuracy as 0.835 also sensitivity of 0.44 also specificity as 0.634 0.663 and f1-score as 0.529 similarly. Then the DBN technique accuracy as 0.867 also sensitivity of 0.427 also specificity as 0.649 also precision as 0.669 and f1-score as 0.521 similarly. Then the CNN technique accuracy as 0.839 also sensitivity of 0.76 also specificity as 0.7 also precision as 0.730 and f1-score as 0.752 consistently. Then the LSTM technique accuracy as 0.843 also sensitivity of 0.83 also specificity as 0.75 also precision as 0.800 and f1-score as 0.860 similarly. Then the RNN technique accuracy as 0.849 also sensitivity of 0.893 also specificity as 0.756 also precision as 0.808 and f1-score as 0.866 harmoniously. Then the ReS- GEMOY technique accuracy as 0.86 also sensitivity of 0.938 also specificity as 0.778 also precision as 0.818 and f1-score as 0.874 correspondingly.

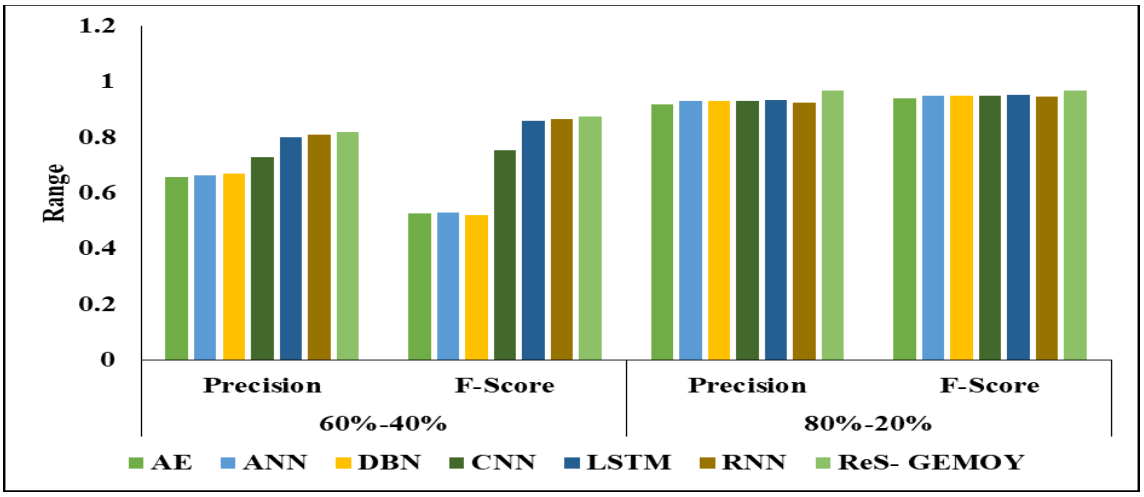

Fig 3. Visual Representation of Different Models on Various Metrics.

V. CONCLUSION

This article presents a new way to increase the recognition rate and identify human activity. to utilised the AreM dataset and implemented the explainable ReS network for classification. By presenting ReS, a network classification model enabled by XAI and based on path signatures, this study enhances cybersecurity for 5G networks and beyond. The combination of DL and path-signature techniques improves the efficiency and explainability of identification. This takes care of the problem of missing data because of compressed signatures. It also optimises the model's parameters using GEMOY, which increases its accuracy, and it captures spatiotemporal features that can be explained. The findings demonstrated that the dataset parameters had a important impact on the network's performance, highlighting the significance of finding an efficient way to obtain the desired values. Lastly, the suggested method surpassed the state-of-the HAR, as shown by the architecture and results. Despite the proposed replica's generally strong performance rate in identifying complex activities in the dataset under consideration, it achieved a low F1-score when it came to distinguishing between a handful of activities that were nearly identical. On top of that, activities can improve their recognition rate for complex tasks by receiving sensor-wise concentration for various features.

Data Availability

No data was used to support this study.

Conflicts of Interests

The author(s) declare(s) that they have no conflicts of interest.

Funding
No funding agency is associated with this research.

Competing Interests
There are no competing interests.

References

[1]. F. Serpush, M. B. Menhaj, B. Masoumi, and B. Karasfi, "Wearable Sensor-Based Human Activity Recognition in the Smart Healthcare System," Computational Intelligence and Neuroscience, vol. 2022, pp. 1–31, Feb. 2022, doi: 10.1155/2022/1391906.
[2]. I. Dirgová Luptáková, M. Kubovčík, and J. Pospíchal, "Wearable Sensor-Based Human Activity Recognition with Transformer Model," Sensors, vol. 22, no. 5, p. 1911, Mar. 2022, doi: 10.3390/s22051911.
[3]. V. Bijalwan, V. B. Semwal, and V. Gupta, "Wearable sensor-based pattern mining for human activity recognition: deep learning approach," Industrial Robot: the international journal of robotics research and application, vol. 49, no. 1, pp. 21–33, Aug. 2021, doi: 10.1108/ir-09-2020-0187.
[4]. V. Uma Maheswari, S. Stephe, R. Aluvalu, A. Thirumalraj, and S. N. Mohanty, "Chaotic Satin Bowerbird Optimizer Based Advanced AI Techniques for Detection of COVID-19 Diseases from CT Scans Images," New Generation Computing, Aug. 2024, doi: 10.1007/s00354-024-00279-w.
[5]. Y. J. Luwe, C. P. Lee, and K. M. Lim, "Wearable Sensor-Based Human Activity Recognition with Hybrid Deep Learning Model," Informatics, vol. 9, no. 3, p. 56, Jul. 2022, doi: 10.3390/informatics9030056.
[6]. H. Park, N. Kim, G. H. Lee, and J. K. Choi, "MultiCNN-FilterLSTM: Resource-efficient sensor-based human activity recognition in IoT applications," Future Generation Computer Systems, vol. 139, pp. 196–209, Feb. 2023, doi: 10.1016/j.future.2022.09.024.
[7]. A. Ferrari, D. Micucci, M. Mobilio, and P. Napoletano, "Deep learning and model personalization in sensor-based human activity recognition," Journal of Reliable Intelligent Environments, vol. 9, no. 1, pp. 27–39, Jan. 2022, doi: 10.1007/s40860-021-00167-w.
[8]. S. Baswaraju, A. Thirumalraj, and B. Manjunatha, "Unlocking the Potential of Deep Learning in Knee Bone Cancer Diagnosis Using MSCSA-Net Segmentation and MLGC-LTNet Classification," Sustainable Development Using Private AI, pp. 190–213, Jul. 2024, doi: 10.1201/9781032716749-10.
[9]. D. Bhattacharya, D. Sharma, W. Kim, M. F. Ijaz, and P. K. Singh, "Ensem-HAR: An Ensemble Deep Learning Model for Smartphone Sensor-Based Human Activity Recognition for Measurement of Elderly Health Monitoring," Biosensors, vol. 12, no. 6, p. 393, Jun. 2022, doi: 10.3390/bios12060393.
[10]. H. M. Balaha and A. E.-S. Hassan, "Comprehensive machine and deep learning analysis of sensor-based human activity recognition," Neural Computing and Applications, vol. 35, no. 17, pp. 12793–12831, Mar. 2023, doi: 10.1007/s00521-023-08374-7.
[11]. S. Mekruksavanich and A. Jitpattanakul, "Hybrid convolution neural network with channel attention mechanism for sensor-based human activity recognition," Scientific Reports, vol. 13, no. 1, Jul. 2023, doi: 10.1038/s41598-023-39080-y.
[12]. S. Kobayashi, T. Hasegawa, T. Miyoshi, and M. Koshino, "MarNASNets: Toward CNN Model Architectures Specific to Sensor-Based Human Activity Recognition," IEEE Sensors Journal, vol. 23, no. 16, pp. 18708–18717, Aug. 2023, doi: 10.1109/jsen.2023.3292380.
[13]. A. Dahou, M. A. A. Al-Qaness, M. A. Elaziz, and A. M. Helmi, "MLCNNwav: Multilevel Convolutional Neural Network With Wavelet Transformations for Sensor-Based Human Activity Recognition," IEEE Internet of Things Journal, vol. 11, no. 1, pp. 820–828, Jan. 2024, doi: 10.1109/jiot.2023.3286378.
[14]. C. Anitha, T. Rajesh Kumar, R. Balamanigandan, and R. Mahaveerakannan, "Fault Diagnosis of Tenessee Eastman Process with Detection Quality Using IMVOA with Hybrid DL Technique in IIOT," SN Computer Science, vol. 4, no. 5, Jun. 2023, doi: 10.1007/s42979-023-01851-9.
[15]. S. Geravesh and V. Rupapara, "Artificial neural networks for human activity recognition using sensor based dataset," Multimedia Tools and Applications, vol. 82, no. 10, pp. 14815–14835, Oct. 2022, doi: 10.1007/s11042-022-13716-z.
[16]. F. Duan, T. Zhu, J. Wang, L. Chen, H. Ning, and Y. Wan, "A Multitask Deep Learning Approach for Sensor-Based Human Activity Recognition and Segmentation," IEEE Transactions on Instrumentation and Measurement, vol. 72, pp. 1–12, 2023, doi: 10.1109/tim.2023.3273673.
[17]. D.-A. Nguyen, C. Pham, and N.-A. Le-Khac, "Virtual Fusion With Contrastive Learning for Single-Sensor-Based Activity Recognition," IEEE Sensors Journal, vol. 24, no. 15, pp. 25041–25048, Aug. 2024, doi: 10.1109/jsen.2024.3412397.
[18]. Y. Zhou et al., "AutoAugHAR: Automated Data Augmentation for Sensor-based Human Activity Recognition," Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, vol. 8, no. 2, pp. 1–27, May 2024, doi: 10.1145/3659589.
[19]. H. Park, G. H. Lee, J. Han, and J. K. Choi, "Multiclass autoencoder-based active learning for sensor-based human activity recognition," Future Generation Computer Systems, vol. 151, pp. 71–84, Feb. 2024, doi: 10.1016/j.future.2023.09.029.
[20]. Ray, L. S. S., Geißler, D., Liu, M., Zhou, B., Suh, S., & Lukowicz, P. "ALS-HAR: Harnessing Wearable Ambient Light Sensors to Enhance IMU-based HAR". arXiv preprint arXiv:2408.09527. (2024).
[21]. P. K. Lakineni, R. Balamanigandan, T. Rajesh Kumar, V. Sathyendra Kumar, R. Mahaveerakannan, and C. Swetha, "Securing the E-records of Patient Data Using the Hybrid Encryption Model with Okamoto–Uchiyama Cryptosystem in Smart Healthcare," Proceedings of Data Analytics and Management, pp. 499–511, 2023, doi: 10.1007/978-981-99-6553-3_38.
[22]. A. Sasi Kumar, T. Rajesh Kumar, R. Balamanigandan, R. Meganathan, R. Karwa, and R. Mahaveerakannan, "Cuttlefish Algorithm-Based Deep Learning Model to Predict the Missing Data in Healthcare Application," Proceedings of Data Analytics and Management, pp. 513–528, 2023, doi: 10.1007/978-981-99-6553-3_39.
[23]. F. Palumbo, C. Gallicchio, R. Pucci, and A. Micheli, "Human activity recognition using multisensor data fusion based on Reservoir Computing," Journal of Ambient Intelligence and Smart Environments, vol. 8, no. 2, pp. 87–107, Mar. 2016, doi: 10.3233/ais-160372.
[24]. Z. Wang, T. Liu, X. Wu, and C. Liu, "A diagnosis method for imbalanced bearing data based on improved SMOTE model combined with CNN-AM," Journal of Computational Design and Engineering, vol. 10, no. 5, pp. 1930–1940, Aug. 2023, doi: 10.1093/jcde/qwad081.
[25]. L. Sun, Y. Wang, Y. Ren, and F. Xia, "Path signature-based XAI-enabled network time series classification," Science China Information Sciences, vol. 67, no. 7, Jun. 2024, doi: 10.1007/s11432-023-3978-y.