Proc. of the European Safety and Reliability Conference (ESREL 2026)
Edited by José C. Matos, Paulo B. Lourenço, Michael Beer, Daniel Oliveira, Edoardo Patelli, Hélder S. Sousa, Rui A. Silva and Monica Santamaria

doi: 10.3850/ESREL2026061419_esrel26-p26301-cd

How Well Do Automatically Evolved Neurons Perform? A Comparative Study in Human Activity Recognition

Chau Thi Thuy Tran
Applied Data Science, Institute for Energy Technology, Norway.
E-mail: chau.tran@ife.no

Hoa Thi Nguyen
Applied Data Science, Institute for Energy Technology, Norway.
E-mail: hoathi.nguyen@ife.no

Lars Vidar Magnusson
Department of Computer Science and Communication, Østfold University College, Norway.
E-mail: lars.v.magnusson@hiof.no

Roland Olsson
Department of Computer Science and Communication, Østfold University College, Norway.
E-mail: roland.olsson@hiof.no

Neuroevolution, a subfield of AI leveraging evolutionary algorithms, offers an effective method for automatically generating data-adaptive equations that optimize fit for specific datasets while minimizing human design bias. This study investigates the efficacy of the WISDM-ARN, a specialized Adaptive Recurrent Neuron (ARN) synthesized via the Automatic Design of Algorithms through Evolution (ADATE) system, for Human Activity Recognition (HAR). In safety-critical applications, such as healthcare monitoring and industrial safety, the reliability of locomotion classification is paramount to prevent system failures and ensure user protection. We benchmark the evolved WISDM-ARN against three established architectures: Simple RNN, Long Short-Term Memory (LSTM), and Transformers. Experimental results demonstrate that the WISDM-ARN successfully adapts to the unique temporal characteristics of gyroscope data, achieving a testing accuracy of 97%, which is comparable to the baseline models with resulting 70%, 95%, and 92%, respectively. These findings highlight the potential of neuroevolutionary synthesis as a dependable alternative to manually engineered architectures. By integrating neuroevolution with modern deep learning, the ADATE-generated neuron represents a significant step toward automated reliable neural modeling for human activity recognition tasks and safety-enhanced time-series analysis.

Keywords: Evolved Recursive Neuron, Recurrent Neural Networks, Human Activity Recognition.

1. Introduction

Human activity recognition (HAR) is a typical classification problem which was first introduced in 1999. The pioneering work employed Electromyography signals from a calibrated accelerometer to classify tremor activity Foerster and Smeja (1999). Since then, HAR has evolved significantly, with ongoing research focused on improving both its performance and downstream applications. At its core, HAR involves using patterns in human gait to identify the locomotion state Gupta et al. (2013) or to authenticate users Kothamachu Ramesh et al. (2021). It has been applied across various domains, including healthcare Straczkiewicz et al. (2019) and smart homes Kalimuthu et al. (2021), where unique human movement patterns provide valuable insights. Research focuses on optimizing accuracy and efficiency across various benchmarks, such as MHealth Banos et al. (2014) and OPPORTUNITY Roggen et al. (2010). In this study, we utilize the WISDM dataset Kwapisz et al. (2011), notable for its large-scale participation of 51 individuals, as the primary case study for evaluating specialized neural architectures.

Early HAR research relied on traditional machine learning, such as MLP, SVM, and Random

Forest, combined with handcrafted features. On the WISDM dataset, MLP initially established a 91.7% baseline accuracy Kwapisz et al. (2011), while later SVM models reached 95.13% Khare et al. (2020). However, the reliance on manual feature extraction prompted a shift toward deep learning Bengio (2013). Modern architectures like CNNs, RNNs, and hybrid models like LSTM-CNN have since achieved results up to 95.75% on WISDM Xia et al. (2020) by automating the feature extraction process.

A distinct paradigm is *neuroevolution*, a methodology that automatically adapts and evolves neural structures based on the specific input-output characteristics of a dataset. Olsson (1995) proposed the *Automatic Design of Algorithms Through Evolution* (ADATE) system, an automated software architect that enables the synthesis of neural models tailored to a given domain. When ADATE is configured with a recurrent-based structure to evolve specific mathematical formulas and memory configurations, the resulting units are termed *Adaptive Recurrent Neurons* (ARNs). Previous research has successfully applied ARNs to safety-critical industrial tasks, such as detecting anomalies in oil wells using the 3W dataset Magnusson et al. (2023) and monitoring CNC machine operator actions Nguyen et al. (2024). These applications underscore the robustness of the ARN framework in high-stakes environments where reliability is essential.

The aim of this paper is to examine the effectiveness of the WISDM-ARN, a personalized neuron instance synthesized specifically for the WISDM dataset, by benchmarking its performance against a Vanilla RNN, an LSTM, and a Transformer-based network. Our findings reveal that the WISDM-ARN achieves a superior accuracy of 97%, demonstrating the efficacy of the neuroevolutionary approach.

2. WISDM Dataset

WISDM is a public dataset including samples from the controlled and natural environment. It is under the WISDM lab that focuses on mining the sensor data for identifying user authentication or human physical tasks. Its data is captured through the Actitracker application, available in the Google Play Store for Android devices Lockhart et al. (2011). Although the dataset has three versions Kwapisz et al. (2011), the term WISDM presents the first version throughout the report. The characteristics of WISDM are described in Table 1. Its sampling frequency is 20Hz, meaning 20 samples are collected per second. The WISDM team supervised the collection procedure for quality control.

Table 1. WISDM Dataset Description.

Description	Samples	Percentage
Total of Samples	1,098,207	100%
No of attributes	Six (6)	–
Missing Value	None	–
Class Distribution		
Walking	424.400	38.6%
Jogging	342.177	31.2%
Up-stairs	122.869	11.2%
Down-stairs	100.427	9.1%
Sitting	59.939	5.5%
Standing	48.395	4.4%

It is noticeable that the WISDM data is unbalanced distributed among users and activities. For instance, from Fig. 1, the “Walking” and “Jogging” classes dominate the class distribution, accounting for nearly two-thirds of the total dataset. Or user number 20 provided the most data points and five times larger than the smallest one (more than 50,000 examples).

Examining the activity patterns with x, y, z-axis values, as depicted in Fig. 2 (right), reveals

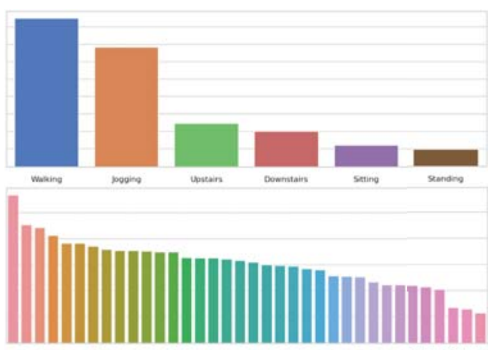

Fig. 1. Activities and users’ sample distribution in WISDM.

noteworthy observations. The static motions exhibit a consistent trend, with sensor data remaining relatively stable. Distinguishing between “Sitting” and “Standing” primarily revolves around the intersection between the x and z-axis magnitudes. In contrast, active actions display greater fluctuations in their graphs. While “Downstairs” and “Upstairs” showcase opposing trends along the x and z-axes, the two remaining activities exhibit similar variations. However, it is worth noting that “Jogging” exhibits higher-frequency oscillations compared to “Walking”. These findings underscore the presence of distinctive sequences within each movement, contributing to their discernible.

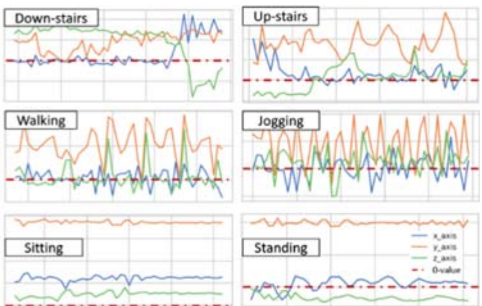

Fig. 2. Activities’ patterns in WISDM.

3. The Evolved Neuron for WISDM

The ADATE system acts as an automated software architect, employing principles of natural selection to evolve code through a hierarchical “kingdom” of candidate programs. It refines these programs using a suite of transformations: *Replacement* for logic substitution, *Abstraction* for helper function generation, and *Recursion* to facilitate self-calling functions. Crucially, ADATE utilizes “neutral” transformations to navigate the search space and overcome local optima. In this study, we utilized ADATE to synthesize the WISDM-ARN, a personalized recurrent neuron whose mathematical definitions and memory structures are tailored specifically for the WISDM dataset. While the full complexities of ADATE’s implementation are outside this study’s scope, comprehensive details are provided by Olsson Olsson (1995) and Løkketangen and Olsson Løkketangen and Olsson (2010).

The architectural innovations within the WISDM-ARN are characterized by three fundamental properties. First, it utilizes a set of five evolved memory cells, $\mathbf{z}^{(t)}_{\{0,1,2,3\}}$ and $\mathbf{f}^{(t)}$, to capture complex temporal dependencies. Second, it introduces auxiliary vectors $\mathbf{a}_{\{0,1,2,3,4\}}$ to calibrate signal flow in the absence of standard weight values at specific nodes. Lastly, it incorporates a sequence of nested activation functions $\sigma_{\{0,1,2,3\}}$, which provide the non-linear transformations necessary for complex pattern recognition:

$$\sigma_0(\star) = \tanh(\text{relu}(\star)) \tag{1a}$$
$$\sigma_1(\star) = \tanh(\tanh(\tanh(\star))) \tag{1b}$$
$$\sigma_2(\star) = \tanh(\tanh(\tanh(\text{srelu}(\tanh(\star))))) \tag{1c}$$
$$\sigma_3(\star) = \sigma_0(\star) \tag{1d}$$

While $\tanh$ and ReLU are known activation functions, this evolved neuron introduces sReLU($\star$), a saturating ReLU defined as $\max(-1, \min(1, \star))$. To bridge the gap between evolved code and neural implementation, we define the intermediate pre-activation stages $\overline{\mathbf{z}}^{(t)}_{\{0,1,2,3\}}$:

$$\overline{\mathbf{z}}^{(t)}_0 = \mathbf{a}_0 \odot \mathbf{W}_3\mathbf{z}^{(t-1)} + \mathbf{U}_0\mathbf{x}^{(t)} \tag{2a}$$
$$\overline{\mathbf{z}}^{(t)}_1 = \mathbf{W}_4\mathbf{z}^{(t-1)} \tag{2b}$$
$$\overline{\mathbf{z}}^{(t)}_2 = \mathbf{a}_1 \odot \mathbf{W}_2\mathbf{z}^{(t-1)} + \mathbf{a}_2 \odot \mathbf{f}^{(t-2)} + \mathbf{a}_3 \tag{2c}$$
$$\overline{\mathbf{z}}^{(t)}_3 = \mathbf{U}_2\mathbf{x}^{(t)} \tag{2d}$$

Here, $\mathbf{W}_i$ represents hollow recurrent weight matrices (with zero diagonals) for hidden-to-hidden connections, while $\mathbf{U}_i$ denotes input-to-hidden weights. The symbols $\odot$ and $*$ represent matrix multiplication and element-wise products, respectively. The final behavior of the memory cells, where $\mathbf{z}^{(t)}$ acts as the master state derived from the sub-cell interaction, and the quadratic output $\mathbf{y}^{(t)}$ are governed by:

$$\mathbf{z}^{(t)}_0 = \sigma_0(\overline{\mathbf{z}}^{(t)}_0) \tag{3a}$$
$$\mathbf{z}^{(t)}_1 = \sigma_1(\overline{\mathbf{z}}^{(t)}_1 * \mathbf{z}^{(t)}_0) \tag{3b}$$
$$\mathbf{z}^{(t)}_2 = \sigma_2(\overline{\mathbf{z}}^{(t)}_2) \tag{3c}$$

$$\mathbf{z}_3^{(t)} = \sigma_3(\overline{\mathbf{z}}_3^{(t)}) \tag{3d}$$
$$\mathbf{z}^{(t)} = \mathbf{z}_1^{(t)} - (\mathbf{z}_2^{(t)} + \mathbf{z}_3^{(t)}) \tag{3e}$$
$$\mathbf{f}^{(t)} = \mathbf{a}_4 \odot \mathbf{z}^{(t)} + \mathbf{W}_0\mathbf{z}^{(t-1)} \tag{3f}$$
$$\mathbf{y}^{(t)} = \mathbf{y}^{(t-1)} - \mathbf{z}^{(t)} * (\mathbf{z}^{(t)} - \mathbf{z}^{(t-2)}) \tag{3g}$$

In this configuration, $\mathbf{z}_1$ through $\mathbf{z}_3$ function as specialized feature extractors, while the final output $\mathbf{y}^{(t)}$ utilizes a quadratic form and a self-referential short-connection to its previous state to maintain long-term context.

4. Recurrent Neural Network Variants

In addition to the neural network constructed using the evolved neuron, three distinct variants of recurrent neural networks have been employed to facilitate the recognition of human activities within the WISDM dataset. These variants include the Vanilla Recurrent Neural Network, the Long Short-term Memory (LSTM), and the Transformer.

4.1. *Vanilla Recurrent Neural Network*

The traditional Recurrent Neural Network (RNN) is a specific class of artificial neural networks designed for the purpose of handling sequential data. What sets RNN apart from conventional feed-forward neural networks is its utilization of an internal cell, known as hidden states, to encapsulate prior information. Visually, the connections within RNN, whether among neurons or layers, configure a comprehensive directed graph (digraph). This digraph encompasses a forward pathway for data propagation as well as a feedback loop that retains historical memory derived from past inputs. Because RNN only depends on the time in an ambient space, the network input and output can be varied in size.

The hidden state $\mathbf{h}^{(t)}$ at each timesteps is defined as a linear combination of the current input $\mathbf{x}^{(t)}$, the preceding hidden state $\mathbf{h}^{(t-1)}$, and an additional bias $\mathbf{b}_h$, as shown in Eq. 4. Then, predicted outputs are calculated using Eq. 5.

$$\mathbf{h}^{(t)} = \sigma_h(\mathbf{U}\mathbf{x}^{(t)} + \mathbf{W}\mathbf{h}^{(t-1)} + \mathbf{b}_h) \tag{4}$$
$$\hat{\mathbf{y}}^{(t)} = \sigma_y(\mathbf{V}\mathbf{h}^{(t)} + \mathbf{b}_y) \tag{5}$$

Typically, a hyperbolic tangent function, $tanh$, is employed as the σ_h activation function to introduce non-linearity into the neuron, while a $softmax$ function, σ_y, is utilized to compute the output probabilities represented as $\hat{\mathbf{y}}^{(t)}$. $\mathbf{U}$, $\mathbf{W}$, and $\mathbf{V}$ serve as weight matrices for input, hidden state, and output, accordingly.

4.2. *Long Short-term Memory*

Long Short-Term Memory (LSTM) represents a specialized form of Recurrent Neural Network (RNN) designed to address longstanding issues like vanishing and exploding gradients, which hinder the handling of back-flow or long-term dependency problems Hochreiter and Schmidhuber (1997). Since its inception, LSTM has undergone enhancements to increase its efficiency and adaptability when dealing with sequential data.

Unlike the standard RNN with a chain-like structure of repeating modules, LSTM incorporates feedback connections that allow it to process entire sequences of data. It comprises multiple memory blocks interconnected in a recurrent manner. Each memory block encompasses a memory cell $\mathbf{c}^{(t)}$ and multiplicative units, known as an input block $\mathbf{z}^{(t)}$ and gates, such as an input gate $\mathbf{i}^{(t)}$), a forget gate $\mathbf{f}^{(t)}$, and an output gate $\mathbf{o}^{(t)}$. All of which represent innovative components central to the functionality of LSTM. These units are formulated in Eq. 6a, Eq. 6b, Eq. 6c, and Eq. 6d.

$$\mathbf{z}^{(t)} = \mathbf{V}_z\mathbf{y}^{(t-1)} + \mathbf{U}_z\mathbf{x}^{(t-1)} + \mathbf{b}_z \tag{6a}$$
$$\mathbf{i}^{(t)} = \mathbf{V}_i\mathbf{y}^{(t-1)} + \mathbf{U}_i\mathbf{x}^{(t-1)} + \mathbf{b}_i \tag{6b}$$
$$\mathbf{f}^{(t)} = \mathbf{V}_f\mathbf{y}^{(t-1)} + \mathbf{U}_f\mathbf{x}^{(t-1)} + \mathbf{b}_f \tag{6c}$$
$$\mathbf{o}^{(t)} = \mathbf{V}_o\mathbf{y}^{(t-1)} + \mathbf{U}_o\mathbf{x}^{(t-1)} + \mathbf{b}_o \tag{6d}$$

The cell and neuron output are defined as in Eq. 7 and Eq. 8.

$$\mathbf{c}^{(t)} = \sigma(\mathbf{i}^{(t)}) \odot \sigma(\mathbf{z}^{(t)}) + \mathbf{c}^{(t-1)} \odot \sigma(\mathbf{f}^{(t)}) \tag{7}$$
$$\hat{\mathbf{y}}^{(t)} = \sigma(\mathbf{o}^{(t)}) \odot \phi(\mathbf{c}^{(t)}), \tag{8}$$

where σ is the sigmoid function and ϕ is the hyperbolic tangent. Abstractly, the internal cells remember data during the entire chain, whereas the gates manage the data movements. They give LSTM the ability to store and select the information from the past.

4.3. *Transformer*

Since its introduction by Vaswani et al. Vaswani et al. (2017) in 2018, Transformers, along with their resampling variants, have emerged as predominant models across a wide spectrum of natural language processing tasks. The fundamental building blocks of Transformer models, namely the self-attention mechanism and the position encoder, stand out as pivotal features. These components are regarded as substantial enhancements when compared to traditional encoder-decoder RNNs employed in sequence-to-sequence applications, including machine translations Bahdanau et al. (2014). Subsequently, this architecture has found application in various other domains, notably in the realm of time series analysis. In this context, the model have demonstrated exceptional performance, particularly in the prediction of univariate time series data Li et al. (2019).

In this study, we utilize a customized variant of the Transformer model specifically tailored for the analysis of multivariate time series data, as originally proposed by Zerveas et al. (2021). This adaptation is employed to address tasks encompassing both time series regression and classification. Given the unique characteristics of the problem at hand, the model exclusively incorporates the encoder component from Vaswani's Transformer. Additionally, it integrates a learnable positional encoding into the input vectors to effectively capture the sequential nature inherent in time series data.

Since human activity recognition is categorized as a classification task, the output from the encoder is further processed by passing it through a softmax function. This step is taken to obtain a probability distribution across different classes, ensuring effective classification.

5. Methodology

5.1. *Data Preparation*

Since sensor-based data is inherently time-series in nature, we implemented a four-stage transformation process—comprising cleaning, splitting, segmenting, and standardizing—to prepare the WISDM dataset for model training. Initially, raw data from CSV files undergoes cleaning and reformatting to remove unwanted samples and ensure structural uniformity. In the splitting stage, the data is partitioned into training and validation sets using a 50% ratio; to address imbalances across users and activities, this division is performed individually for each user-activity combination.

Subsequently, both subsets are segmented into non-overlapping windows. Based on literature suggesting optimal performance within a one-to-three-second timeframe Preece et al. (2008); Henprasert-tae et al. (2011); Sun et al. (2010), we selected a window size of two seconds (40 samples). Finally, the segmented data is centralized and encoded to standardize feature variations, ensuring stable convergence during the neural network training process.

5.2. *Model Architecture and Optimization Strategy*

Regarding the structural configurations, the ARN, Vanilla RNN, and LSTM models utilize a streamlined four-layer architecture consisting of an input layer, a recurrent layer with 81 hidden units, a value determined optimal through preliminary testing, a dense layer, and an output layer with a linear activation function. In contrast, the Transformer model employs three sequential encoder blocks and eight multi-head attention mechanisms, adhering to the original setup proposed by Zerveas et al. Zerveas et al. (2021).

The training process for these models is governed by the interplay between the optimization algorithm and the learning rate scheduler: the former dictates the direction of weight updates, while the latter regulates their magnitude to ensure convergence toward the loss function's minima. In this study, we employ the Adam adaptive optimizer Kingma and Ba (2014) paired with a customized linear decay scheduler to dynamically adjust the learning rate across epochs. This scheduler is defined by the initial learning rate η_0, a decay factor α, and the total training steps T required for η_0 to reach its final value $\alpha\eta_0$. As formalized in Eq. 10, the learning rate $\eta^{(t)}$ at any given timestep t is determined by the decay function $d^{(t)}$:

$$d^{(t)} = 1 - (1-\alpha)\frac{t}{T} \tag{9}$$

$$\eta^{(t)} = \begin{cases} d^{(t)} \cdot \eta_0 & (t \leq T), \\ \alpha\eta_0 & (t > T). \end{cases} \tag{10}$$

5.3. *Evaluation Metrics*

Mean Squared Error (MSE) is used as the loss function while training for ARN, Vanilla RNN, and LSTM. It computes the average of the squared differences between the output value $\hat{y}$ and target y value. Its equation is presented in Equation 11.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2. \tag{11}$$

Regarding Transformer, we deploy the Categorical Cross Entropy (CCE) to calculate the error. It is defined as Eq. 12.

$$CCE = -\sum_{i=1}^{N} y_i \cdot log(\hat{y}_i). \tag{12}$$

For the testing phase, we assessed the final performance on the test data using the accuracy score. This score is determined by dividing the number of correctly classified instances by the total number of classified instances.

6. Results and Discussion

Table. 2 provides the result performance of four networks in term of accuracy and precision. WISDM-ARN outperforms the Vanilla RNN, LSTM, and Transformer, with an achievement of approximately 97%. In addition, we justified the result using a statistical method called McNemar Test. This number helps to determine if any two models perform better than others when the p-value exceeds the significance threshold α. After calculating between ARN and LSTM, the p-value we got is 4.76e-07, which is far below the chosen threshold.

Table 2. The results of four deployed networks regarding accuracy and precision.

Models	Accuracy	Precision
Vanilla RNN	0.705	0.714
LSTM	0.958	0.961
Transformer	0.924	0.922
ARN	0.970	0.969

When comparing the neuron structures of Vanilla RNN and LSTM, it is evident that the evolved neuron shares a similar selective gate structure with its predecessors. More specifically, $\mathbf{z}_3$ works as an input gate to determine which values are fed into the system. Concurrently, $\mathbf{z}_0$, $\mathbf{z}_1$, and $\mathbf{z}_2$ serve as control blocks that collectively regulate the update flow for the hidden state $\mathbf{z}$. However, the WISDM-ARN exhibits a more intricate architecture than its predecessors. It boasts five memories, in contrast to the two in LSTM or the one in RNN, and entails more complex formulas. Additionally, it captures the value of two timesteps as opposed to one. As the trade-off of the increase in resources and process time, more information is stored and processed at a timestep. Furthermore, the new neuron incorporates a sequence of activation functions, namely the combination of $tanh$, ReLU, and sReLU. This multifaceted activation sequence adds to its computational sophistication, enabling more complex transformations of data. Lastly, it incorporates a quadratic property into the output calculation. Based on the author's knowledge, the quadratic form has not been explored in the output function of a recurrent neuron network even though it has been used in the regression score function, named $\mathcal{R}^2$, as an evaluation metric.

7. Conclusion

This study demonstrates that the neuroevolutionary synthesis of specialized neurons via the ADATE system provides a robust alternative to manually engineered deep learning architectures. The resulting WISDM-ARN achieved a considerable accuracy of 97% on the WISDM dataset, outperforming established models such as the LSTM and Transformer without handcrafted feature extraction or efforts to address data imbalance. This performance indicates that neuroevolution can effectively discover data-adaptive pathways that optimize the classification of complex temporal patterns in human locomotion.

While the computational cost of the evolutionary search phase is significant, the resulting WISDM-ARN is highly efficient during inference, making it suitable for real-time deployment in smart environments and wearable healthcare devices. Future research will explore the flexibility of the ADATE system or neuroevolutionary approach in generating neurons for other high-stakes time-series domains, such as structural health monitoring and predictive maintenance in cognitive robotics. By tailoring neural logic to specific data distributions, we can move toward a more automated, reliable, and fail-safe framework for intelligent human-centric systems.

References

Bahdanau, D., K. Cho, and Y. Bengio (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Banos, O., R. Garcia, J. A. Holgado-Terriza, M. Damas, H. Pomares, I. Rojas, A. Saez, and C. Villalonga (2014). mhealthdroid: a novel framework for agile development of mobile health applications. In *International workshop on ambient assisted living*, pp. 91–98. Springer.

Bengio, Y. (2013). Deep learning of representations: Looking forward. In *International conference on statistical language and speech processing*, pp. 1–37. Springer.

Foerster, F. and M. Smeja (1999). Joint amplitude and frequency analysis of tremor activity. *Electromyography and clinical neurophysiology 39*(1), 11–19.

Gupta, J. P., N. Singh, P. Dixit, V. B. Semwal, and S. R. Dubey (2013). Human activity recognition using gait pattern. *International Journal of Computer Vision and Image Processing (IJCVIP) 3*(3), 31–53.

Henpraserttae, A., S. Thiemjarus, and S. Marukatat (2011). Accurate activity recognition using a mobile phone regardless of device orientation and location. In *2011 International Conference on Body Sensor Networks*, pp. 41–46. IEEE.

Hochreiter, S. and J. Schmidhuber (1997). Long short-term memory. *Neural computation 9*(8), 1735–1780.

Kalimuthu, S., T. Perumal, R. Yaakob, E. Marlisah, and L. Babangida (2021). Human activity recognition based on smart home environment and their applications, challenges. In *2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, pp. 815–819. IEEE.

Khare, S., S. Sarkar, and M. Totaro (2020). Comparison of sensor-based datasets for human activity recognition in wearable iot. In *2020 IEEE 6th World Forum on Internet of Things (WF-IoT)*, pp. 1–6. IEEE.

Kingma, D. P. and J. Ba (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kothamachu Ramesh, A., K. S. Gajjala, K. Nakano, and B. Chakraborty (2021). Person authentication by gait data from smartphone sensors using convolutional autoencoder. In *International Conference on Intelligence Science*, pp. 149–158. Springer.

Kwapisz, J. R., G. M. Weiss, and S. A. Moore (2011). Activity recognition using cell phone accelerometers. *ACM SigKDD Explorations Newsletter 12*(2), 74–82.

Li, S., X. Jin, Y. Xuan, X. Zhou, W. Chen, Y.-X. Wang, and X. Yan (2019). Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. *Advances in neural information processing systems 32*.

Lockhart, J. W., G. M. Weiss, J. C. Xue, S. T. Gallagher, A. B. Grosner, and T. T. Pulickal (2011). Design considerations for the wisdm smart phone-based sensor mining architecture. In *Proceedings of the Fifth International Workshop on Knowledge Discovery from Sensor Data*, pp. 25–33.

Løkketangen, A. and R. Olsson (2010). Generating meta-heuristic optimization code using adate. *Journal of Heuristics 16*(6), 911–930.

Magnusson, L. V., J. R. Olsson, and C. T. T. Tran (2023). Recurrent neural networks for oil well event prediction. *IEEE Intelligent Systems 38*(2), 73–80.

Nguyen, H. T., R. Olsson, and Ø. Haugen (2024). Automatic synthesis of recurrent neurons for imitation learning from cnc machine operators. *IEEE Open Journal of the Industrial Electronics Society 5*, 91–108.

Olsson, R. (1995). Inductive functional programming using incremental program transformation.

Artificial intelligence 74(1), 55–81.

Preece, S. J., J. Y. Goulermas, L. P. Kenney, and D. Howard (2008). A comparison of feature extraction methods for the classification of dynamic activities from accelerometer data. *IEEE Transactions on Biomedical Engineering 56*(3), 871–879.

Roggen, D., A. Calatroni, M. Rossi, T. Holleczek, K. Förster, G. Tröster, P. Lukowicz, D. Bannach, G. Pirkl, A. Ferscha, et al. (2010). Collecting complex activity datasets in highly rich networked sensor environments. In *2010 Seventh international conference on networked sensing systems (INSS)*, pp. 233–240. IEEE.

Straczkiewicz, M., P. James, and J.-P. Onnela (2019). A systematic review of smartphone-based human activity recognition for health research. *arXiv preprint arXiv:1910.03970*.

Sun, L., D. Zhang, B. Li, B. Guo, and S. Li (2010). Activity recognition on an accelerometer embedded mobile phone with varying positions and orientations. In *International conference on ubiquitous intelligence and computing*, pp. 548–562. Springer.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin (2017). Attention is all you need. *Advances in neural information processing systems 30*.

Xia, K., J. Huang, and H. Wang (2020). Lstm-cnn architecture for human activity recognition. *Ieee Access 8*, 56855–56866.

Zerveas, G., S. Jayaraman, D. Patel, A. Bhamidipaty, and C. Eickhoff (2021). A transformer-based framework for multivariate time series representation learning. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pp. 2114–2124.