International Journal for Modern Trends in Science and Technology, 9(05): 51-59, 2023

ISSN: 2455-3778 online
DOI: https://doi.org/10.46501/IJMTST0905009
Available online at: http://www.ijmtst.com/vol9issue05.html

An Examination of Reservoir Computing in Latest Technologies

Karan Chawla

Ashoka Unviersity

To Cite this Article
Karan Chawla. An Examination of Reservoir Computing in Latest Technologies. International Journal for Modern Trends in Science and Technology 2023, 9(05), pp. 51-59. https://doi.org/10.46501/IJMTST0905009

Article Info
Received: 12April 2023; Accepted: 04 May 2023; Published: 05 May 2023.

ABSTRACT

The basic purpose of reservoir computing is to accelerate machine learning algorithms.The term "reservoir" describes a dynamical system. The reservoir is made up of a numberof parts that are repeatedly joined and randomly linked. Therefore, reservoir computing,as mentioned earlier, uses a recurrent neural network, but instead of updating all of theparameters of the network, it only updates a few, leaving the others fixed by choosingthem at random. Energy harvesting from renewable resources like solar and wind isattracting a lot of interest from both academia and business due to the continuous growthin global energy demand and recent breakthroughs in this field. Energy harvestingtechnology is anticipated to power up to 80% of smart grid components, including smartmetres and sensors, which would significantly reduce the cost of replacing batteries andongoing maintenance of smart grids. Establishing a network for smart water delivery is the first step in developing smart cities. In order to link aging waterinfrastructures—some of which have been in place for more than a century—to otherparts of the system and the city, IoT technologies must be upgraded and improved.Due to its capacity to choose the data/metadata that are important or worth recording forreal-time post-processing, edge intelligence devices are becoming more and more indemand. These devices prevent pointless data transfers to the cloud. This review paperanalyzes three of the latest technologies in reservoir computing which is cybersecurity,smart cities and edge intelligence as well as gives future directions for the same

KEYWORDS:Reservoir Computing, Machine Learning, Edge Intelligence, IOT technologies,Solar Energy.

1. INTRODUCTION

A Recurrent Neural Network is used in reservoir computing (RC), however not all of the network's parameters are updated. Only some parameters are updated; the other parameters are fixed and chosen at random. It maps input signals to higher dimensional computational spaces by using the dynamics of a fixed, nonlinear system called a reservoir. A basic readout mechanism is then taught to read the reservoir's state and map it to the required output after the reservoir is utilized as a black box and an input signal is fed into it. Only at the readout stage is training conducted since the reservoir dynamics are fixed. In traditional reservoir computing, the reservoir must possess two characteristics: first, it must be made up of distinct, non-linear units; second, it must be able to store data. Reservoir computing is a technique that is mainly used to speed up machine learning algorithms. 'Reservoir' refers to a dynamical system.

A mathematical function that explains how a location in space behaves over time is used to identify dynamical systems. You might be able to forecast where that point in space will be in the future if you are familiar with these systems. The reservoir consists of a number of randomly linked recurrently connected pieces. So, as previously mentioned, reservoir computing makes use of a recurrent neural network, but instead of updating all of the network's parameters, it only updates a select few while leaving the others fixed by selecting them at random. Processing temporal or sequential types of data is a task for which reservoir computing is quite effective. This occurs as a result of its layout. Reservoir computing's paradigm is similar to one based on recurrent neural networks. Echo state networks, liquid state machines, and other recurrent neural network models produced the contemporary reservoir computing frameworks. In reservoir computing, the reservoir's role is to convert sequential inputs nonlinearly into a high-dimensional space such that a straightforward learning algorithm can effectively read out the inputs' characteristics. This greatly aids in accelerating machine learning algorithms and employing these quicker learning techniques. Other dynamical systems can also be used as reservoirs in addition to recurrent neural networks. Physical Reservoir Computing was also created as a result of this. Because reservoir computers are relatively simple to train, novice developers prefer to use this framework. Building systems with the capacity to handle information and data at a quicker rate with a reduced learning cost is the entire goal of reservoir computing. Due to the fact that machine learning often has a high power consumption while training big datasets, this is particularly crucial.

2.CYBER-SECURITY

Due to the continued rise in global energy consumption and recent developments in this sector, energy harvesting from renewable resources like solar and wind is receiving a lot of interest from both academia and business. Up to 80% of smart grid components, such as smart meters and sensors, are expected to be powered by energy harvesting technology, which will drastically lower the cost of replacing batteries and the continuous maintenance of smart grids. The dependability of smart grids as a whole depends on cybersecurity. The most dangerous type of potential cyber-attack is false data injection, or FDI. These attacks can be carried out by adversaries infiltrating smart meters to add fraudulent measurements1. If the state estimation is impacted by these malicious measurements, the power grid control algorithms may be misled, which might have disastrous effects like widespread blackouts. As a result, the most crucial stage in reducing FDI-related harms is attack detection. Smart grid performance can be significantly impacted by the effectiveness and efficiency of FDI detection. Because the spatio-temporal correlation of the data is not taken into consideration during training, feedforward neural networks have been employed for FDI detection but have not produced satisfactory results. The FDI problem in smart grids was originally discussed and the benefits and drawbacks of each solution are discussed. For FDI in smart grids, several algorithms have so far been devised. The state vector estimation technique is one of the earliest ones used in these methods. Additionally, machine learning methods have been used for the FDI detection of smart grids. To be more precise, current work on FDI detection has used feedforward neural networks, K-nearest neighbors, support vector machines, and sparse logistic regression. Recurrent neural networks (RNNs), on the other hand, are discovered to be able to take advantage of the underlying correlation in the data. It was demonstrated that RNNs are perfect approximations of dynamic systems given very moderate and generic assumptions. However, it may sometimes be extremely challenging or even impossible to train a fully linked RNN. Reservoir computing (RC), which uses straightforward training techniques, has lately received a lot of interest due to the difficulties of training conventional RNNs. The two most common RC systems are liquid state machines (LSM) and echo state networks (ESN). In contrast to ESN, which works with ordinary data that is not spiked, LSM requires spiking trains as the input, which must be encoded using temporal or other encoding strategies. The reservoir, readout/output layer, and input layer make up the three main layers of a conventional RC system. However, the majority of these methods rely on manually selected meta-characteristics and model-specific parameters. Although the feedforward neural network permits some autonomy, it often performs severely below optimally when working with linked input. When used on IEEE test systems, machine learning techniques produce superior outcomes to support vector estimating techniques. In order to enhance the efficiency of state vector estimation, the effectiveness of Precision Measurement Units (PMUs) has been thoroughly examined. Cramer investigated extended distributed state estimation (EDSE). Each power system is divided into numerous subsystems by EDSE using graph partition techniques, and the buses in each subsystem are categorized into three primary groups: boundary buses, internal buses, and nearby buses. Compared to conventional state estimate approaches, EDSE-based methods perform better. The investigation of the current link between the physical characteristics of the power system and FDI allows for the identification of compromised nodes. The reservoir is mostly made up of neurons that are linked randomly, with the weights of those connections remaining constant throughout training. The reservoirs are combined linearly in the readout/output layer to get the required output. It has been demonstrated that in many cases, RC systems outperform conventional RNNs. It has been shown that delayed feedback networks (DFNs) can function as RC systems as well. A nonlinear node replaces the collection of sparsely linked neurons (reservoirs) in LSM and ESN. This method not only makes RC systems' structure simpler, but it also shows a very high level of computing efficiency. A nonlinear node into which the input is introduced can easily alter the parallelism that exists in many other artificial neural network designs. DFN's performance has been shown to be very similar to that of other RC systems. The network can imitate fleeting brain responses because of the short-term dynamic memory created by delayed networks with feedback. The correlation between the input and output signals is represented mathematically by transfer functions. In RC, the necessary nonlinear mapping is accomplished via nonlinear transfer functions. We created a small analogue delay-based reservoir node based on the Mackey-Glass function. The newly presented delayed feedback reservoir has a single nonlinear node with a delay loop, just like conventional delayed feedback reservoir designs. Due to the nonlinear mapping of the delayed feedback reservoir's input to a higher dimensional space, the spiking nonlinear neural node also fulfills the same function. The neuronal information has been encoded using a variety of approaches. The two most common ones are rate encoding and temporal encoding. A code in rate encoding is made up of several spikes that arrive in a period of time after the stimulus. The three basic categories of temporal encoding are latency code, interspike intervals, and firing phase. The timing of the initial spike is utilized for encoding in latency code. Another coding method that uses the gaps between various spikes is known as interspike interval coding. The local field power (LFP) phase is utilized to encode the information in the temporal encoding utilizing the firing method. According to studies, rate encoding loses information faster than interspike interval encoding. As a result, in this paper, the encoder of our RC systems is interspike interval temporal encoding. We will be able to conduct anomaly detection in cyber physical systems (CPS) efficiently and effectively utilizing RC if we have the platform of analogue spiking RC architecture. In this research, we specifically demonstrate how to quickly and accurately identify attacks on smart grids utilizing DFNs and MLPs. Our proposed architecture exhibits a significant amount of robustness with regard to numerous attack variants when compared to existing attack detection methods in smart grids. Delayed feedback RC systems perform almost equally to conventional RC systems. Delay loop plus a single nonlinear node make up the delayed feedback reservoir, which is different from the conventional reservoir. The reservoir's output will go through a training procedure using a training algorithm. The goal of the training is to make sure that the state's weighted sum is close to the desired output value. The nonlinear node receives the input directly. A masking approach is used before the nonlinear node to make up for the loss of parallelism. The input signals are scaled during the masking process so that they will be in the transient regime. Following the masking step, the signals are sent to the nonlinear node where the nonlinear mapping is done. The only learned weights, just like in conventional RC, are the connections for the output weights. Rate encoding and temporal encoding are the two main categories of encoding techniques. The input information is represented by the amount of spikes, with other spike properties being disregarded, according to the rate encoding technique. Contrarily, temporal encoding incorporates information into the pauses between spikes. Analogue signals will be converted into spike-based information via temporal encoding, which has the advantage of being both compact and energy-efficient. In our design, we employ temporal encoding, and the temporal encoder adopts an iterative structure with an exponential connection between the number of neurons and the number of spikes. So, fewer neurons would be required to generate the same amount of spikes. The signals are then sent to the nonlinear node where the nonlinear mapping is done after the masking process. The only training weights are the connections between the output weights, just like in conventional RC. The temporal encoder makes sure that just one neuron is operating in the dynamic mode, which has a far lower power need. In order to maximize the use of the device area, our presented temporal encoder was constructed utilizing a 180 nm CMOS technology and symmetry approach. Our solution incorporates both the internal verification method and the output temporal code, which has a high error-tolerance mechanism made possible by taking use of the extra inspection spikes. In addition to being very accurate, the newly developed neuron uses less power than existing cutting-edge neuron designs. For each sample in the measurement matrix, we were able to extract five alternative states. A multi-layer perceptron (MLP) will be trained using these states. The times at which spikes are happening for the associated state of each sample serve as the feature for training the MLP. The appropriate label of the attacked data for training the reservoir state is taken into consideration as one and zero otherwise since half of the samples are attacked. The performance of the system is then assessed using the test data after the MLP has been trained using the training data. Both MLP and SVM's performance are highly dependent on the quantity and size of attacks on the meters. Both MLP and SVM, in contrast to SVE, may find covert assaults. However, the assault settings have a significant impact on their detection capabilities. For instance, when the attack magnitude rises, both MLP and SVM's accuracy rises. As a result, assaults with enormous magnitudes may be reliably detected by MLP and SVM. MLP and SVM, on the other hand, will less reliably detect assaults when they have tiny magnitudes. For the instance of MLP specifically, the accuracy can range from 100% when the assault magnitude is 10 to as low as 70% when the attack magnitude is 0.1. In smart grids when the assault size might be arbitrary, this is not very ideal for attack detection. One can observe that the differences in attack magnitude for the RC-based DFN+MLP approach do not significantly alter accuracy. For the RC-based technique, the accuracy variance due to the change in assault magnitude is quite minor and is near to 100% for all attack magnitudes.

3. SMART CITY

The first stage in creating smart cities is creating a network for smart water supply. IoT technologies must be updated and modified in order to bring ageing water infrastructures—some of which have been in existence for more than a century—online and connect them to other components of the system and the city. Similar to smart energy systems, smart water systems employ IoT-enabled sensors to gather real-time data. By identifying leaks or keeping track of how water is dispersed across the network, this enables the optimisation of water infrastructure and helps users to manage water resources more wisely. For resolving static spatial issues, traditional computational intelligence designs like basic neural networks, Bayesian models, and kernel approaches are the best options. They lack feedback systems, hence they are unable to manage time-dependent issues. In actuality, they fail to successfully simulate situations that need for temporal processing, such data streams. Additionally, there are numerous studies that use online learning techniques to solve static spatial problems. For the purpose of ensuring many-core design, some have suggested a real-time online learning technique. Many-core offers input to the online learning algorithm based on core information and its behavior to the incoming data packet in order to prevent unanticipated attacks. For instance, these intelligent sensors may find leaks in water pipelines and promptly notify engineers to take action and lessen the effects. Energy, gas, and water systems are particularly vulnerable to cyberattacks as high-growth vital infrastructure assets. Water infrastructure is a particularly enticing target for numerous attack vectors, including insider, outsider, and terrorist attackers, due to its crucial role in our society and its growing reliance on linked systems. Data generated by several sources, including sensors, meters, and IoT/IIoT devices, must be continuously analyzed to ensure the security of these crucial infrastructures. These data are notable for their vast quantity, erratic nature, and rapid creation pace. At each data transfer, the suggested online learning method updates the model run-time based on feedback from many-core. Additionally, a cutting-edge support vector machine and self-organizing incremental neural network are both used in a novel network intrusion prevention system that has been proposed. The suggested system's structure enables a security solution that does not rely on signatures or rules and is very accurate in mitigating both known and unidentified threats in real time. Online learner evaluation is challenging, though. It can be challenging to get the algorithm to function "correctly" automatically for similar reasons. It might be challenging to determine whether a problem is with the infrastructure or the algorithm. Recurrent neural networks (RNNs), which have feedback connections built into them, were used to solve the aforementioned problem. The mechanism that takes time into account as a separate dimension is provided by the network's states being reliant on earlier ones. Traditional RNNs, however, have significant design and training issues. These restrictions were removed with the introduction of reservoir computing (RC). In essence, RC is a feedback neural network (NN) with time-varying input signals. A RCNN is made up of a reservoir, a high-dimensional, fixed (random), nonlinear dynamic system that is driven by a time-dependent input layer and has a linear output layer. A group of node units that are connected on a regular basis make up the reservoir. They often have a random connection topology, and the units are nonlinear. A linear combination of the reservoir's internal instantaneous states, which retain memory from prior inputs, results in the time-dependent output. The only system parameters that can be learnt are linear combination weights, considerably simplifying the training of feedback networks. In reality, the training process is often guaranteed to converge to a universal ideal using linear approaches. The RC training procedure is incredibly well suited for the modeling of complicated scenarios that demand time processing because of how flexible and simple it is. The echo state network (ESN), which utilises analogue neurons with sparse random connections in the hidden layer, and the liquid state machine (LSM), which employs leaky integrate and fire neurons (LIF) with a synaptic dependency model, are the two fundamental reservoir computing designs. To the best of our knowledge, the suggested model presents (for the first time in the literature) the design of a specialized ORC architecture for SCIP that has low computational resource needs; it is effective and suited for real-time data flow analysis. It's a brand-new ESN model that consists of analogue neurons with random connections at input levels and in the dynamical reservoir. The RLQ approach is used to train it at the output level. An iterative neural network, the echo state network (ESN) has an input, a sparsely connected reservoir, and a basic linear output (readout). Both the input weights and the connection weights in the ESN reservoir are arbitrary. To ensure echo state property (ESP), the reservoir weights are scaled. This is referred to as a condition that is somewhat influenced by the design of the reservoir and in which the reservoir acts as an "echo" of its whole entrance history. The input u(n) and output y(n), which are established by the issue, are the sole unique levels of the ESN. The number of the hidden levels, which are gathered in the dynamical reservoir (DR), is indistinguishable. The sparsity of the DR, which is obtained for each problem using the experimental technique, is dictated by the degree to which the neurons in the DR, x(n), are linked. A value that sets the weights characterizes the synaptic unions between the levels and the DR. Each input neuron in the ESN is coupled to each DR neuron by Winij weights (ith input neuron, jth DR neuron). Although normalized, these weights are chosen at random from the start, and their values are set because they do not change as a result of training. Additionally, every neuron in the dynamic reservoir is connected to every other neuron in the DR via weights Wjk (jth DR neuron, with kth, where j k). These weights are likewise chosen at random prior to training and remain constant. Finally, the output neurons are connected to every DR neuron. Only these readout layer weights are trained prior to reaching their final values. It is shown that the given technique outperforms the corresponding HAT and S-Pegasos algorithms in terms of classification accuracy. On the other hand, it exhibits comparable behavior in terms of the demands placed on the available computing resources, where the models are noted as being almost identical. The issue at hand is also changing in real time because of its nature. The suggested technique successfully manages data by utilizing temporal difference within the confines of temporal windows thanks to its reservoir computing design. This method considers any correlations and dependencies that could be present in the sequence of the data stream. The cheap computational cost of developing the ESN network, which benefits from all the common RNN feedback methods but does not experience sluggish convergence, was a significant reason in the choice to pick this design. Since the weights are assigned randomly to the input-level neurons and the DR, there is no significant delay, as would otherwise be the case with methods like backpropagation. It also relates to a significant drawback of the suggested approach, namely the need for specialized expertise and extensive testing to fully comprehend how the perfect network functions in each scenario. The development of a real-time data flow analysis approach, which is extremely effective, functional, and quick without requiring structures with large processing costs, is the system's most significant innovation. This model incorporates the benefits of a number of approaches that are well-known and in use, and it can be quickly used to provide protection in the real-world security arena. The use of this technique to identify digital assaults on contemporary mechatronic networks is another crucial advance. A multifaceted and intricate security issue like the one under discussion is difficult to solve.

4. EDGE INTELLIGENCE

Implementing low-power machine learning (ML) methodologies in a single chip platform is necessary for the creation of effective Internet of Things (IoT) systems. massive chip areas and substantial parallelism are needed for ML applications to analyze massive amounts of data quickly (devices that are not currently accessible on edge devices). Sending the data that was recorded to cloud servers and then waiting for the answer that the servers processed is a state-of-the-art approach. This approach calls for a significant amount of data transfer, which in turn causes network congestion and a reliance on servers. As a result, there is an increasing need for edge processing optimization, notably in applications for smart devices and the Internet of Things. The area and power constraints of edge nodes prevent the adoption of conventional deep learning techniques, which afterwards imply substantial computer power, hence research on edge intelligence (EI) is still in its infancy. Therefore, the ability to effectively implement precise and energy-efficient EI chips is of great interest to the microelectronic industry. The capacity of artificial neural networks (ANNs) to resolve common real-world issues like image or sound identification makes them one of the primary ML approaches used to create AI systems. The convolutional neural network (CNN), which uses a feed-forward neural network made up of several sequentially connected convolution and information reduction layers (like max-pooling or average-pooling), is the most often used ANN. When used to solve image or sound identification issues, CNNs exhibit cutting-edge performance, but at the cost of having to execute several multiply-and-accumulate (MAC) operations. When evaluating CNNs as a viable approach for low-power EI applications, this can have serious negative effects on latency, power, and energy consumption. Recent developments in low-power vocal activity detection and keyword spotting implementations employing reasonably small CNNs, which include spectrogram-based feature extraction techniques comparable to the one taken into consideration in this study, are relevant in this context. Because of its straightforward learning procedure and the use of fixed weights inside the ANN structure that are independent of the training process, reservoir computing (RC) is an appealing ANN training framework with relatively simple computation. A random recurrent neural network (RNN), also known as the reservoir, and a set of inputs that are randomly connected to it make up most RC systems. While training is typically done using ordinary least squares (OLS) over the reservoir states, all internal and input connections to the reservoir are kept fixed. By employing a certain ring topology, it is possible to optimize RC systems for hardware implementation so that each neuron has a low fan-in, which makes hardware implementation easier. In addition to this ring topology, the reservoir connectivity may also be optimized by choosing certain weights in order to conduct just straightforward shift-and-add operations rather than computationally intensive MAC operations at each neural connection. Previous works on FPGA implementations have concentrated on the so-called single-node reservoir, which is based on just one physical node and can represent a ring topology using time division multiplexing with an input mask and particular nonlinearities with practicable electronic and optical implementations. Additionally, hardware implementations of RC have been used in spoken digit recognition in the past. Our work in this area varies from earlier works on FPGA implementations in two key ways: the training approach and the digital implementation. The primary distinction is that training is carried out utilising log-mel energies as the input characteristics on a per-frame basis. Additionally, because the implementation is register-based, fully parallel, and has a very minor nonlinearity at each node, the reservoir states are not kept in RAM. When used to solve time-series forecasting or equalisation issues, this optimized RC model has shown to have high accuracy and energy efficiency properties. A reservoir computing

(RC) system transfers input data to a higher dimensional space, making it more likely to distinguish the input data as the reservoir size rises. Only the output layer, which is linked to the reservoir, is trained in RC systems using OLS or cross-entropy loss minimization, while the connectivity of the reservoir as a whole is left fixed. Nodes may be built using conventional artificial neural networks (echo state networks), spiking neurons (liquid state machines), or cellular automata (ReCA systems) and the internal connection of the reservoir is generally sparse (typically 1% connectivity). A multidimensional audio event classification job has been chosen, and the topology used is ring topology or single cycle reservoir (SCR). Specifically, a feature extraction digital block pre-processes each audio signal in hardware and outputs 64 (M = 64) 8-bit log-mel spectral characteristics per frame that are utilized as reservoir inputs. The selected reservoir architecture is cyclic, with each neuron having two inputs: the external signal coming from one frequency channel and the output signal from the prior neuron in the ring. This design was chosen for a small hardware implementation. For internal inputs, the connection weights of the neurons are fixed to either r, and for input-to-reservoir connections, to either + v, v, or 0 (not connected). The random parameters ij are introduced, which alter the sign of the external inputs, to highlight the unpredictable character of external weights (which can be either positive, negative, or zero). These variables randomly select the values +1, 1, or 0.

5. ANALYSIS

For the convolutional neural network, automating the categorization of habitability, machine learning can help with the issue of habitability disposition. Synthetic Minority Oversampling Technique (SMOTE) was the method that dealt with the issue of class imbalance the best. In comparison to Random Oversampling and Random Undersampling, this approach yielded good results. After boosting the minority class data by synthetically creating new examples, the machine learning model was able to train and generalize successfully. Support Vector Classifier was also used to achieve cost-sensitive learning, but the outcomes were the same as those attained by employing SMOTE. The limitation of the study is that it is not easy to detect reflected light from a planet's atmosphere. For the k-means clustering method, Exhaustive tasks like scanning transiting light curves for planetary signals are jobs that ML techniques are capable of handling. One of the most challenging characteristics that prevent ML methods from performing to their full potential is noise in the light curve signals. By producing false positives or even obscuring the transit signals from the detection models, noisy features might trick AI algorithms. Human interaction is still necessary (for example, in feature extraction) even if the current ML algorithms lighten the workload for scientists working to validate exoplanet discoveries. Additionally, weak transit signals offer a fantastic chance to discover exoplanets that resemble Earth. The optimal machine learning model should be able to analyze weak signals. To solve the issue posed by transits seen in low SNR light curves, this calls for a better grade of detection and identification capability. For these reasons, MRA appears to be a viable method for finding tiny planets and validating the signals that are found. MRA may reduce the amount of the data while simultaneously collecting fine details from the light curves. This enhances the ML models' identification performance and considerably reduces the execution time.

6. ANALYSIS

In cyber-security, The assault employed in the initial research is not time-variant. For smart cities, The neural network is a powerful modeling tool. As a result, the test set used to optimize the topology of the network may not be sufficient to verify the network's generalizability. Some researchers do not make use of this additional validation set in our presentation. This is because there aren't any objects, and our presentation acts as an example. The root mean square error of prediction is used to validate the network performance. The easiest way to keep track of how well the network design can be generalized is to validate its performance using the ESN model. This presumption is supported by the observation that the network transforms into an extremely adaptable function mapper when additional neurons are added to the hidden layer. This increases the risk of overfitting in turn. The projections on the test data could not be as accurate as the learning data mapping. This idea serves as the foundation for the validation, which is crucial for determining the number of nodes in the hidden layers as well as the number of iterations in the learning process. For Edge intelligence, compared to state-of-the-art network topologies, the proposed reservoir computing system is significantly smaller and requires far less MAC procedures and parameters, which are ultimately connected to latency, power dissipation, and energy consumption. Additionally, it is demonstrated that the proposed sound recognition hardware classifier uses up to 40% less energy than a recently published sound recognition solution that relies on an inexpensive and low-power ARM Cortex-M4F microcontroller. Therefore, a tempting possibility to potentially reduce energy consumption in some tasks would be to add RC dedicated hardware to similar system-on-chip architectures. In contrast, for the specific multi-class audio event detection system, typical machine learning models like kNN or decision trees exhibit similar or worse accuracy in terms of error performance. In addition, it is straightforward enough to be a contender for a number of battery-operated edge situations, such as always-on inference scenarios, RC hardware acceleration, or co-processing in system-on-chip architecture, which includes mobile phones, smartwatches, or smart sensors. Simple audio tagging or detection, monitoring of physiological data, and channel equalization are some examples of potential use cases. Research has demonstrated the cyclic reservoir is effective enough to be employed for per-frame temporal feature extension at the algorithmic level. However, because background and foreground samples are mixed in the Urban Sound 8K dataset and some of them have noisy environments or additional background sources, generalisation is particularly challenging. For example, noise reduction, data augmentation techniques, or another post-processing method could be used to increase test set accuracy. As a result of optimising the reservoir structure for creating a fully parallelized ANN, it has been shown an ultra-low-power auditory event detection system with an energy efficiency in the sub-J/Inf range in this study. The technology is ideal for edge intelligence applications because of these features.

7. CONCLUSION

For Smart Cities, It is crucial to remember that the recommended practice is to employ reservoir computing with the suggested ESN architecture. The suggested paradigm is novel and addresses a genuine security issue with information systems since it is uncommon for all data streams to be of equal value. The technique is applied when the algorithm has to dynamically adapt to new patterns in the data or when the data are created as a function of time. The suggested strategy also works with catastrophic interference, a challenge that may be solved by using other teaching strategies. As the overall behavior of the model becomes less noisy (kappa statistic 81.77), the employed technique offers better prediction (accuracy 98.94) and stability (F-measure 0.990), and the overall risk of making a particularly poor choice is significantly decreased. The dispersion of the predicted error, which is near to the mean error value and strongly reflects the dependability of the system and its generalization potential, further supports the aforementioned premise. For Edge Intelligence, In every situation, input-to-reservoir binary weights match the best-known model chosen at random from a uniform distribution. The feature extraction block and readout behavior taken from FPGA measurements that were prompted by an input log-mel spectrogram. Dog barking in the front and humans conversing in the distance are both simultaneous sources. While the dog bark (DB) class was correctly predicted, the background people speaking was categorized as children playing (CP), which makes sense given that it is the most similar class.

8. FUTURE DIRECTIONS

The assault employed in the preliminary work does not have a temporal variation. The next stage would be to deploy a more difficult time-variant assault. Dynamic assault is the name of this attack. Dynamic assaults are carried out in a method that gradually manipulates the smart grid system's state in the direction the attacker wants it to go. More advanced networks must be given to counter more sophisticated assaults. To achieve this, a deep structure of DFNs with increased computing capacity will be suggested. Depth in time for DFR computing systems results from the delayed signal combining with the fresh input. However, a single reservoir does not provide any depth to space for RNNs or DFRs. By stacking several reservoirs on top of one another between the input and output layers, depth in space might be produced similarly to how feedforward neural networks are stacked in the deep learning area. If one looks at the possibilities of combining deep learning and DFR in addition to the analogue implementation of DFR, Deep DFN and MI-deep DFN, two deep DFN structures, are suggested. The output from the prior layer will be fed into the succeeding reservoir layers in the deep DFN model. For Smart Cities, The exploration of methods to automatically locate and optimize the system's parameters, in order to attain classification accuracy, without human interaction, is a factor that might be investigated in the direction of future expansion. Additionally, the addition of an automatic feature extraction and feature selection process from raw data, related to novel, uncharted situations, could be a significant potential future development. This would enable the improvement of its categorization skills, enabling the detection of fresh, unidentified assaults.

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

References

[1] A. Reuther, P. Michaleas, M. Jones, V. Gadepally, S. Samsi, and J. Kepner,"Survey of machine learning accelerators," in 2020 IEEE High PerformanceExtreme Computing Conference (HPEC). IEEE, 2020, pp. 1–12

[2] A. F. Atiya and A. G. Parlos. New results on recurrent network training: Unifyingthe algorithms and accelerating convergence. IEEE Neural Networks, 11:697,2000.

[3] D. V. Buonomano and M. M. Merzenich. Temporal information transformed into aspatial code by a neural network with realistic properties. Science,267:1028–1030, 1995.

[4] H. Burgsteiner. Training networks of biological realistic spiking neurons forreal-time robot control. In Proc. of EANN, pages 129–136, Lille, France, 2005

[5] X. Dai. Genetic regulatory systems modeled by recurrent neural networks. In Proc.of ISNN, pages 519–524, 2004

[6] K.-I. Funahashi and N. Y. Approximation of dynamical systems by continuoustime recurrent neural networks. Neural Networks, 6:801–806, 1993.

[7] Wikner, A. et al. Using data assimilation to train a hybrid forecast system thatcombines machine-learning and knowledge-based components. Chaos 31, 053114(2021).

[8] Pyle, R., Jovanovic, N., Subramanian, D., Palem, K. V. & Patel, A. B.Domain-driven models yield better predictions at lower cost than reservoircomputers in Lorenz systems. Philos. Trans. R. Soc. A Math. Phys. Eng. Sci. 379,24102 (2021).

[9] Gauthier, D. J. Reservoir computing: harnessing a universal dynamical system.SIAM News 51, 12 (2018)

[10] 0.Gonon, L. & Ortega, J. P. Reservoir computing universality with stochastic inputs.IEEE Trans. Neural Netw. Learn. Syst. 31, 100–112 (2020).

[11] . Platt, J. A., Wong, A. S., Clark, R., Penny, S. G. & Abarbanel, H. D. I. Robustforecasting through generalized synchronization in reservoir computing. Preprintat arXiv:2103.0036 (2021).

[12] Antonik, P., Marsal, N., Brunner, D. &Rontani, D. Bayesian optimisation oflarge-scale photonic reservoir computers. Cogn. Comput. 2021, 1–9 (2021).

[13] .Griffith, A., Pomerance, A. & Gauthier, D. J. Forecasting chaotic systems withvery low connectivity reservoir computers. Chaos 29, 123108 (2019).

[14] Griffith, A., Pomerance, A. & Gauthier, D. J. Forecasting chaotic systems withvery low connectivity reservoir computers. Chaos 29, 123108 (2019)

[15] Yperman, J. & Becker, T. Bayesian optimization of hyper-parameters in reservoircomputing. Preprint at arXiv:1611.0519 (2016).

[16] Pathak, J., Hunt, B., Girvan, M., Lu, Z. & Ott, E. Model-free prediction of largespatiotemporally chaotic systems from data: a reservoir computing approach.Phys. Rev. Lett. 120, 24102 (2018).

[17] Canaday, D., Pomerance, A. & Gauthier, D. J. Model-free control of dynamicalsystems with deep reservoir computing. to appear in J. Phys. Complex.http://iopscience.iop.org/article/10.1088/2632-072X/ac24f3 (2021).

[18] Butcher, J. B., Verstraeten, D., Schrauwen, B., Day, C. R. & Haycock, P. W.Reservoir computing and extreme learning machines for non-linear time-seriesdata analysis. Neural networks 38, 76–89 (2013).

[19] .Butcher, J., Verstraeten, D., Schrauwen, B., Day, C. & Haycock, P. Extendingreservoir computing with random static projections: a hybrid between extreme learningand RC. In 18th European Symposium on Artificial Neural Networks, pp. 303–308(2010).

[20] .Larger, L. et al. High-Speed Photonic Reservoir Computing Using aTime-Delay-Based Architecture: Million Words per Second Classification. Phys.Rev. X 7, 011015 (2017).

[21] Duport, F., Smerieri, A., Akrout, A., Haelterman, M. &Massar, S. Fully analoguephotonic reservoir computer. Sci. Rep. 6, 22381, doi:10.1038/srep22381 (2016).

[22] Paquot, Y. et al. Optoelectronic Reservoir Computing. Sci. Rep. 2, 287,doi:10.1038/srep00287 (2012).

[23] Zhou, C. &Kurths, J. Noise-Induced Phase Synchronization and SynchronizationTransitions in Chaotic Oscillators. Phys. Rev. Lett. 88, 230602 (2002).

[24] Toral, R., Mirasso, C. R., Hernandez-Garcia, E. &Piro, O. Analytical andNumerical Studies of Noise-induced Synchronization of Chaotic Systems. Chaos11, 665 (2001).

[25] .Jaeger, H. & Hass, H. Harnessing Nonlinearity: Predicting Chaotic Systems andSaving Energy in Wireless Communication. Science 304, 78 (2004).