IBIMA Publishing
Communications of International Proceedings
https://ibimapublishing.com/p-articles/45ICT/2025/4539025/
Vol. 2025 (21), Article ID 4539025

Prediction of Spectrum Occupancy for Dynamic Spectrum Access Using Recurrent Neural Network*

Adrian KUBOWICZ and Jerzy LOPATKA

Military University of Technology, Warsaw, Poland

Correspondence should be addressed to: Adrian KUBOWICZ, adrian.kubowicz@student.wat.edu.pl

* Presented at the 45th IBIMA International Conference, 25-26 June 2025, Cordoba, Spain

Abstract

Spectrum handoff upon collision detection is considered an insufficient mechanism in cognitive radio, motivating the development of spectrum occupancy prediction methods for radiocommunication systems. Prediction of transmission opportunities within the bandwidth of primary users represents a promising direction that requires novel approaches supported by advancements in Machine Learning (ML) and Deep Learning (DL). Beyond performance, DL model complexity and computational cost are key parameters, both during training and inference. While numerous modern architectures have been proposed for prediction tasks, a gap still exists between complex and efficient models. We propose a simple yet effective method using a Recurrent Neural Network (RNN), specifically Long Short-Term Memory (LSTM), where the input feature is directly represented as time differences between transmission events. Raw input signals are reduced and processed through a fast signal-processing path, allowing the LSTM to infer from an activity history buffer on demand. The model is implemented using the standard toolset with the Keras framework. The proposed approach is evaluated using synthetic data simulating ON–OFF activity patterns, under assumptions tailored to a representative channel scenario. Results demonstrate that the time-differential LSTM approach can effectively predict and identify transmission window opportunities. LSTM provides learned immunity to the random nature of primary user activity, offering a tradeoff between model complexity and prediction capability.

Keywords: Time Differential RNN, LSTM, Cognitive Radio, DSA

Introduction

The need to use the radio spectrum more flexibly has led to the concept of cognitive radio, where data transmission relies on techniques that monitor the radio environment to gather the information needed to decide whether transmission in a given frequency band is possible or not (Hwang and Yoon, 2008). This becomes especially important when dealing with licensed bands occupied by primary users, where cognitive (secondary) users should not cause any interference. This challenge applies both to modern wireless systems like cellular networks, Wi-Fi, or Bluetooth, and proprietary systems where new devices operate alongside non-cognitive components within available spectrum limits. Military communications and Electronic Warfare are examples, but civilian use of the Internet of Things (IoT) could also benefit from smarter solutions.

There are two main methods of spectrum sharing for secondary users. The first is termed underlay; a transmission uses spread spectrum techniques, which appears to primary users as acceptable background noise. The second is overlay within licensed communication. This involves searching for temporarily unused frequency bands known as white spaces. The cognitive system transmits in these gaps and handoffs when primary user activity reappears, i.e., the basic paradigm of Dynamic Spectrum Access (DSA).

Cite this Article as: Adrian KUBOWICZ and Jerzy LOPATKA, Vol. 2025 (21) “Prediction of Spectrum Occupancy for Dynamic Spectrum Access Using Recurrent Neural Network " Communications of International Proceedings, Vol. 2025 (21), Article ID 4539025, https://doi.org/10.5171/2025.4539025

In cognitive radio systems, it is insufficient to consider using only a single frequency. Time and location also play a role (Skokowski et al., 2022), and simply stopping transmission upon detecting a collision may be inadequate. This also applies to deciding when a transmission can begin. Based on the historical spectrum occupancy in a spectrum band, prediction could provide information on when to start a transmission to maximize the probability of successful reception. Due to the complex wireless environment, delays, fading, multipathing, etc., avoiding collisions cannot be guaranteed at spectrum handoff. For the scenario of interest where sparse data transmission should occur between primary user activities, time windows with the lowest collision probability must be identified even if the secondary transmission must be postponed.

Algorithms' ability to predict the probability of wireless spectrum occupancy in the near future depends on analysis of the features of previous transmissions. These must contain information from which the most probable signal behavior can be inferred. Since the received signal features are sensitive to real-world conditions (Tekbiyik et al., 2019) such as radio channel effects, this becomes a complex task. Methods like Fast Fourier Transform, Wavelet Transform, Power Spectral Density, higher-order statistics, cyclostationary n-order detection, as well as instantaneous amplitude and phase analysis are often used to address this. Key components within a radio frame are the preamble (Lin et al., 2022), sequence word, or other synchronization constant, which must be repeatable and predictable. This feature may facilitate synchronization with primary user transmissions, although modern interfaces tend to apply data whitening to the payload, thus complicating the procedure.

Progress in Machine Learning and Deep Learning has advanced the concept of cognitive radio. ML methods extract signal features using a specifically designed algorithm, which undergoes either classification or regression, whereas DL methods require less expert-based manual signal preprocessing, assuming that the Neural Network (NN) will implicitly extract features. Research is ongoing on various aspects of wireless communication applications using DL. Integrating smart mechanisms into existing infrastructure or embedding them into next-generation design, like 4G/5G (Wasilewska and Bogucka, 2020) or 6G, would be advantageous. For this purpose, Transformer variants (Pan et al., 2025) (Liu et al., 2021) (Zhou et al., 2021) and diverse Convolutional Neural Network (CNN) architectures, with or without pretrained models, are among the novel NNs being investigated. However, their computation requirements limit their suitability for lightweight mobile or hardware applications where devices are continuously operating. Considering both that prediction accuracy is expected to degrade over time and the complex radio environment, spectrum occupancy must be predicted within a reasonable time window when spectrum access for the secondary user is viable.

Research on ML/DL regarding the radio frequency spectrum is mainly focused on Automatic Modulation Classification (AMC) (Rajendran et al., 2018). However, similar models can be adapted for prediction tasks, depending on feature engineering, output layers for NNs, and training. Among many examples, they include various implementations of Support Vector Machine (SVM) (Tekbiyik et al., 2019), Decision Tree (DT), and K-Nearest Neighbors (KNN) (Ali et al., 2025). Deep Learning models often employ the CNN (Olesiński and Piotrowski, 2023) (Rajesh et al., 2022), Autoencoder (Almazrouei et al., 2021), and RNN-type Long Short-Term Memory (LSTM) architectures (Al-Tahmeesschi et al., 2021) (Ali et al., 2021). Envelope methods take into consideration parallel output from different models, like DeepSig (Qiu et al., 2024), which applies a three-path NN for AMC. Nonetheless, there is a gap between the faster ML methods, which depend on feature preprocessing, and the computationally demanding DL models (Soni et al., 2022).

RNNs – specifically LSTMs and simplified Gated Recurrent Units (GRUs) have been widely studied in the context of wireless spectrum access. While LSTMs have been successfully implemented due to their ability to extract long- and short-term time-series features, they have been critiqued for their computational complexity and limited scalability. Digital sampling for spectrum sensing and reception tends to produce large volumes of data (Almazrouei et al., 2021), which must undergo size reduction prior to NN processing. It has been noted that (Toma and Lopez-Benitez, 2021) imperfect spectrum sensing conditions cause synchronization-like problems with a reduced data stream, highlighting the relevance of event-driven feature processing. RNNs with integrated attention mechanisms have also been proposed (Cao et al., 2021) (Wen and Li, 2023) (Liu et al., 2020), and although they are declared as accurate, they are not considered lightweight. In addition, when working on raw data, the LSTMs have been criticized as being computationally slow (Tekbiyik et al., 2019). Researchers have investigated using data compression by quantization (Jagannath and Jagannath, 2022) and even binarization (Rajendran et al., 2018) for LSTMs but point to its inefficiency.

Numerous studies, unrelated to the wireless spectrum, suggest that incorporating time step information makes RNNs more efficient. Time-LSTM (Zhu et al., 2017) and Phased-LSTM (Neil et al., 2016) are extended architectures that use time gates. The latter is more suited for non-regular sampled data than LSTM. Inspired by this, ABBA-LSTM (Elsworth and Güttel, 2020) reduces data complexity by converting signals into symbolic representations. The addressed problem concerns predicting channel occupancy by the primary user's transmission activity, which follows an ON-OFF pattern. The occurrence of each transmission is affected by time uncertainty that can be described by probability distribution. Regular or cyclic types of user activity, e.g., by an

IoT node, can be valid for a certain period until a new pattern emerges requiring adaptation. For this scenario, we propose using an LSTM model to predict binary channel occupancy based on extracted channel-activity features in the form of time differences between transmissions. The model depends on the history of channel occupancy to learn. This approach reduces the LSTM hyperparameters regarding computational complexity, thus allowing lightweight hardware platforms to facilitate online learning. Implementing the direct Time Difference LSTM rather than using traditional techniques allows the learning process to be automated.

Signal Model

The computational complexity of one forward step by the LSTM and its simplified variant GRU can be estimated by the input vector value length *X* and the predicted output vector length *Y*:

$$\mathrm{O}(4(XH + H^2)) + \mathrm{O}(H \cdot Y) \tag{1}$$

$$\mathrm{O}(3(XH + H^2)) + \mathrm{O}(H \cdot Y) \tag{2}$$

The computational complexity scales linearly with the sequence length and quadratically with the number of hidden units *H*. Therefore, it is crucial to fine-tune the RNN and signal feature extraction for lightweight channel occupancy modeling.

Prior to applying the RNN, *N* signal samples must be processed. A basic approach is to perform energy detection to determine transmission presence. If $Ed > Th$ where the threshold Th is based on noise estimation, then Ed can be calculated as squared magnitude of the Fourier Fast Transform (FFT) over N points (Arjoune et al., 2018):

$$Ed = \sum_{n=1}^{N} (\mathrm{FFT}[n])^2 \tag{3}$$

For a propriety system, information about user activity can be extracted from RSSI data or taken from the receiver frame detection times. In the proposed solution, the resulting sample stream is reduced to a series of activity start times, each stored as a value in vector *t*. The next signal processing step converts *t* into time differences *td*:

$$td_i = [t_i - t_{i-1}] \tag{4}$$

where index *i* of the samples is the latest value in the buffer.

The proposed RNN uses a tanh activation function, which expects input values in the range [-1, 1]. Therefore, *td* must be normalized using the minimum td_{min} and maximum td_{max} values:

$$tdnorm_i = -1 + 2\,\frac{td_i - td_{min}}{td_{max} - td_{min}} \tag{5}$$

The lower bound td_{min} can be interpreted as the expected mean time of one user transmission. A drawback is that the maximum value td_{max} must be specified as an arbitrary value representing the longest time between transmissions. The implementation must interpret the maximum value of occurrence, i.e., transmission did not occur, and estimate the lowest probability of collision.

The processed signal is used both for learning and inference. For RNN online learning, the data must be augmented, which requires that each time difference value in the sample buffer must be locally randomized following a Gaussian distribution with an estimated standard deviation, σ. This is a key RNN functionality, enabling temporal flexibility and robustness to impaired user activity data for the learning process.

RNN Model

Fig. 1 depicts using one feature used for the RNN input-output model.

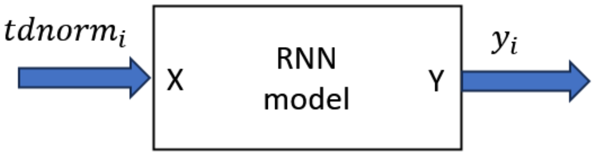

Fig 1. RNN model: Input-output

The input vector of size *X* is equal to or greater than 1 per processing cycle *T*. Larger *X* values allow the RNN to process longer window buffers simultaneously. It is then possible to reset the RNN's internal states for every buffer, provided that the learning method allows it. Conversely, when *X*=1, the only possibility is to infer each value sequentially at input without resetting the internal RNN states (except for off times and online learning requirements), which takes advantage of the LSTM variant's long-term memory. This is a feasible approach for event-driven sampling applied to RNNs. Output *Y* is the predicted continuation of *X*, based on learned weights and internal states depending on the *X* sequence. A *Y* of length 1 means that only one step ahead is predicted. To account for processing time and to estimate a sufficiently long period of low transmission probability in the near future, it is desirable that *Y* is inferred over multiple prediction steps. The predicted time relevance should be determined by summing all output *Y* time differences not exceeding the processing time when prediction is required.

The resulting signal processing structure is shown in Fig. 2.

The decision about predicted channel occupancy is taken by the last block, which applies the policy required by the application. This block incorporates additional SNR feedback, allowing the current SNR to be considered, such as for multichannel selection or validating the current prediction. Meanwhile, the incoming activity time differences samples are collected in a separate buffer as input for the online learning sequence generator.

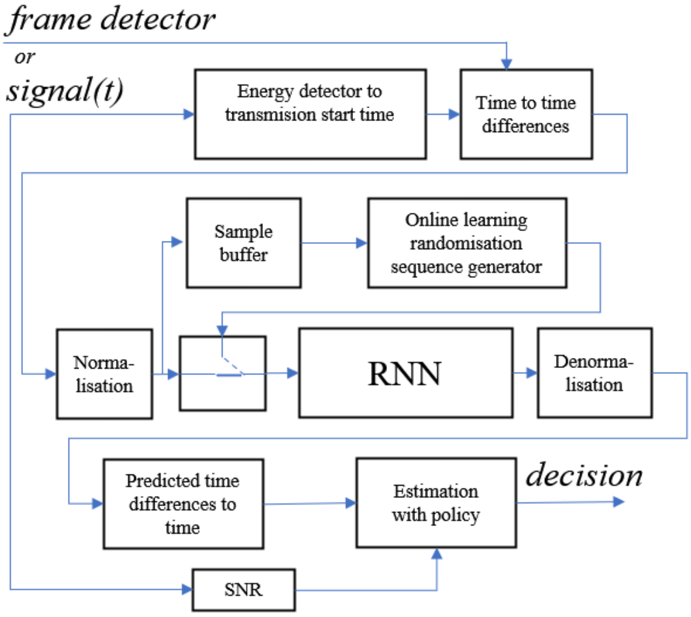

Fig 2. Signal processing flowgraph

Notably, the classic RNN approach cannot simultaneously learn online and infer. As (Corlay et al., 2024) discusses, RNNs are not well-suited for online implementation. However, rather than processing raw data as the input, online learning operates on preprocessed features and is regarded as re-learning, which is triggered when the RNN prediction does not match the observed SNR.

Selected LSTM model parameters are shown in Table (1). Python, Keras API, and Tensorflow comprise the toolset.

Table 1: LSTM parameters

Input	X_len
features	1
Units H	64
Ouput	Y_len
Optimiser	adam
Activation	linear

Experiment Results

Publicly available datasets, mainly designed for AMC, comprise short signal samples representing various conditions and SNRs but are not best suited for our prediction task. Therefore, we provided a randomly selected transmission start time list *t*, to evaluate the RNN functionality. The list is converted into *td* time differences:

$$t = [6, 13, 18, 27, 41, 50, 60, 95] \tag{6}$$

$$td = [6, 7, 5, 9, 14, 9, 10, 35] \tag{7}$$

and cyclically repeated to fit into the *X* length of the RNN input; for this example, it is set to 16 (*X_len* = 16 in Table 1), enabling initialization and forward step inference.

The input vector *X* can be 1 or greater and is provided to the RNN in a single step, supporting *one-to-many* or *many-to-many* regression task. *Y* is arbitrarily set to 16 (*Y_len* = 16 in Table 1), matching the length of *X*. During training, *Y* is the expected continuation of *X* and is input to the RNN. For a given *td,* Eq.(6), there are 8 combinations of sequence shifts; thus the training dataset had 8 samples, enabling the RNN to distinguish patterns from any starting point. This base set was multiplied using 25 randomization cycles with a standard deviation of 0.2, yielding 200 training samples. The number of required cycles was estimated experimentally. Fig. 3 shows the first two training samples. Note that noise was added only to *X*, leaving prediction *Y* unaltered.

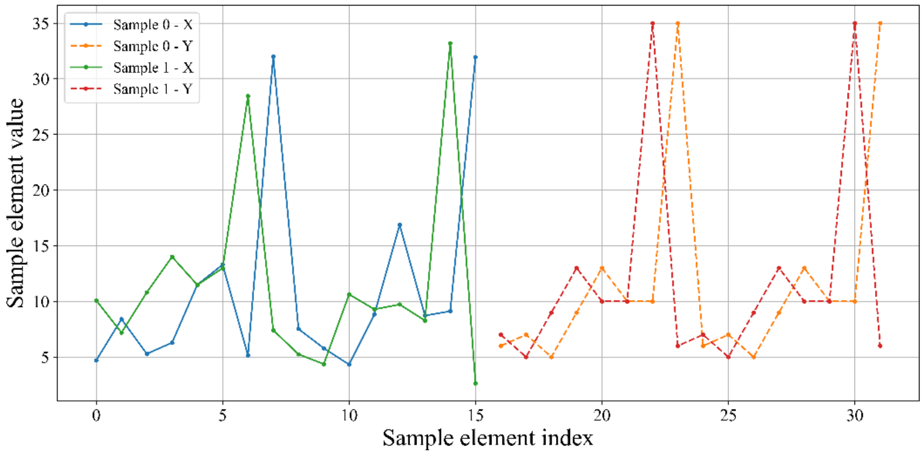

Fig 3. Training samples, X + continuing Y

Following the training step, which produced 53810 parameters, the RNN performed inference with input values of different noise realizations. This is shown in Fig. 4 using two samples, where *Yt* and *Yp* lines overlap.

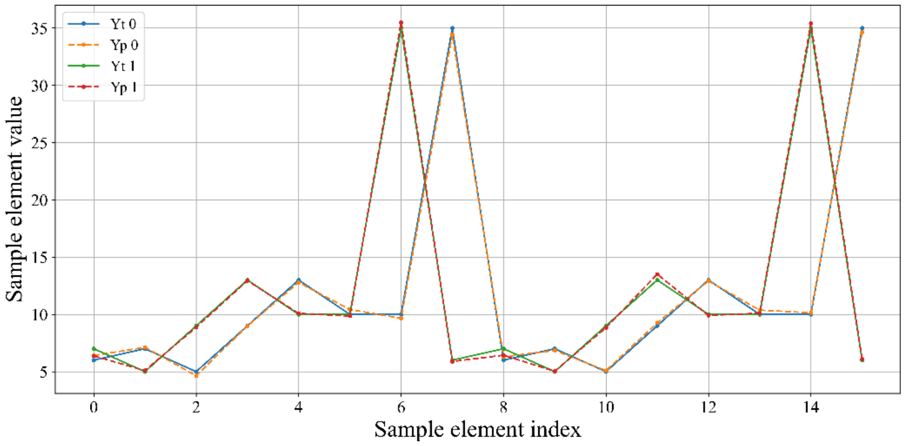

Fig 4. Prediction output (Yp0, Yp1) versus testing (Yt0, Yt1) *Y* data

The error distribution of predictions across the test batch is shown in Fig. 5. Each index corresponds to a *td* value. In the boxplot, the blue boxes correspond to the interquartile range (IQR), and the circles are outlier values. Although the mean IQR value across all indexes is 0.45 for this realization, several outliers exceed the value of the smallest *td* value. Among the samples being compared ($200 \cdot 16 = 3200$ elements), the IQR is 1,28% of the longest value in *td*. The error distribution for all indexes is similar, corresponding to the pattern of elements created from the original 8 *td* values.

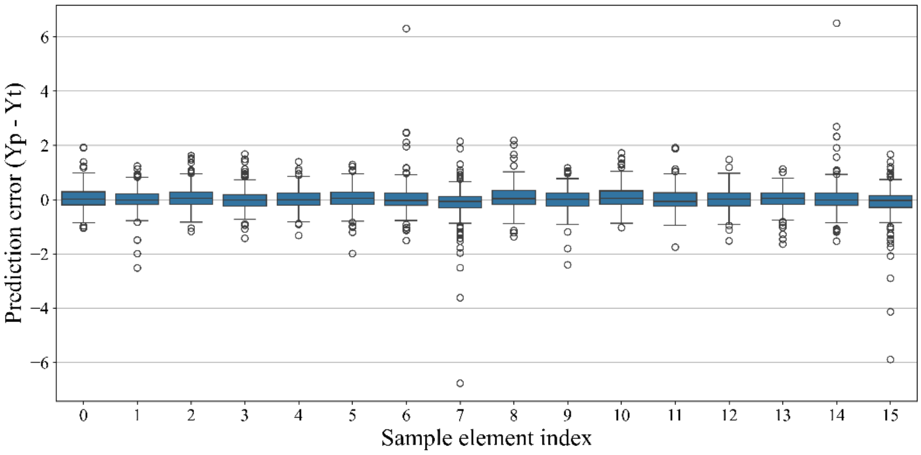

Fig 5. Prediction output versus testing Y data samples error (pattern aligned)

Using the same setup but with an LSTM model comprising 128 units produced an IQR of 0.3 and fewer outliers.

To simulate online learning, the next experiment trained an RNN on the first batch of 200 samples from the previous *t* activity times list. It was trained again on a batch generated from a different input *t*. The prediction outputs for both input samples are compared in Fig. 6. Only the outputs (lines *Yt1_0, Yp1_0, Yt1_1, Yp1_1* in Fig. 6) are properly followed, suggesting that the previously learned pattern was overwritten by the new input.

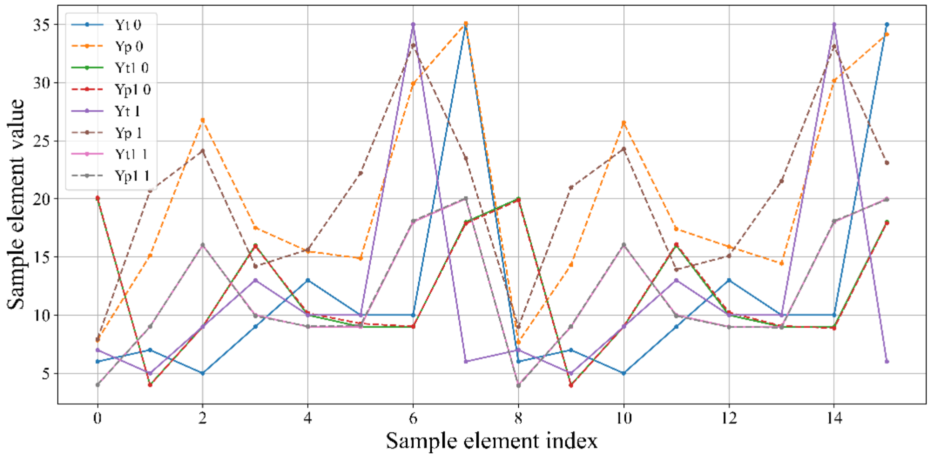

Fig 6. Result for old and new input samples

Summary

The proposed DL model, using time-difference input preprocessing, has significant potential for lightweight implementations. This method can serve as the basis for spectrum occupancy rate estimation but can be expanded to a more elaborate solution, i.e., multichannel decision-making to identify sufficiently long white spaces for secondary user transmission. In the simulated cyclic activity scenario, the LSTM operated like a correlation-based methods but with learned robustness to activity distribution. Using time differences as the RNN input minimizes complexity; however, the experimental activity list would produce a large vector of raw samples because of sparse primary user activity.

Online learning is equivalent to training the RNN from a reset state because adjusting to a new input pattern requires a fully randomized sample batch. In the analyzed scenario, this was enabled by reducing the complexity and computational burden.Another design consideration involves choosing between RNN types. Although the LSTM architecture was selected for these experiments, similar results were achieved with a GRU model using H=32, further reducing complexity. However, this configuration was not validated for longer sequences and a *one-to-many* input-output setup. Our future work includes designing a decision block and improving jamming resilience by detecting invalid transmission frame lengths.

Acknowledgements

This work was financed by Military University of Technology under research project UGB/22-053/2025.

References

- Al-Tahmeesschi, A., Umebayashi, K., Iwata, H., Lehtomaki, J. and Lopez-Benitez, M. (2021) ‘Feature-Based Deep Neural Networks for Short-Term Prediction of WiFi Channel Occupancy Rate’, IEEE Access, 9, pp. 85645–85660.
- Ali, M., Çırpan, H.A. and Arslan, H. (2025) ‘Machine Learning-Based Spectrum Occupancy Prediction: A Comprehensive Survey’, Frontiers in Communications and Networks, 2, article 1482698.
- Ali, M., Çırpan, H.A. and Arslan, H. (2021) ‘Efficient Spectrum Occupancy Prediction Exploiting Multidimensional Correlations through Composite 2D-LSTM Models’, Sensors, 21(1), p. 135.
- Almazrouei, E., Gianini, G., Almoosa, N. and Damiani, E. (2021) ‘Robust Computationally-Efficient Wireless Emitter Classification Using Autoencoders and Convolutional Neural Networks’, Sensors, 21(7), p. 2414.
- Arjoune, Y., El Mrabet, Z., El Ghazi, H. and Tamtaoui, A. (2018) ‘Spectrum Sensing: Enhanced Energy Detection Technique Based on Noise Measurement’, 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, pp. 828–834. IEEE.
- Cao, K., Jiang, M. and Gao, S. (2021) ‘Spectrum Availability Prediction Based on RCS-GRU Model’, Physical Communication, 49 (December), article 101479.
- Corlay, V., Nakazato, T., Yamaguchi, K. and Nakajima, A. (2024) ‘Differentiable High-Order Markov Models for Spectrum Prediction’, arXiv preprint, arXiv:2412.00328.

- Elsworth, S. and Güttel, S. (2020) ‘Time Series Forecasting Using LSTM Networks: A Symbolic Approach’, arXiv preprint, arXiv:2003.05672.
- Hwang, J. and Yoon, H. (2008) ‘Dynamic Spectrum Management Policy for Cognitive Radio: An Analysis of Implementation Feasibility Issues’, 2008 3rd IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks, Chicago, IL, USA, pp. 1–9. IEEE.
- Jagannath, A. and Jagannath, J. (2022) ‘Multi-Task Learning Approach for Modulation and Wireless Signal Classification for 5G and Beyond: Edge Deployment via Model Compression’, Physical Communication, 54 (October), article 101793.
- Lin, Y., Lin, Y., Zha, H., Zhang, J., Wang, Y., Gui, G. and Mao, S. (2022) ‘Large-Scale Real-World Radio Signal Recognition with Deep Learning’, Chinese Journal of Aeronautics, 35(9), pp. 35–48.
- Liu, Y., Duan, J. and Meng, J. (2020) ‘Difference Attention Based Error Correction LSTM Model for Time Series Prediction’, Journal of Physics: Conference Series, 1550(3), p. 032121.
- Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S. and Guo, B. (2021) ‘Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows’, Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 10012–10022.
- Neil, D., Pfeiffer, M. and Liu, S.-C. (2016) ‘Phased LSTM: Accelerating Recurrent Network Training for Long or Event-Based Sequences’, Advances in Neural Information Processing Systems (NeurIPS), 29, pp. 3882–3890.
- Olesiński, A. and Piotrowski, Z. (2023) ‘A Radio Frequency Region-of-Interest Convolutional Neural Network for Wideband Spectrum Sensing’, Sensors, 23(14), p. 6480.
- Pan, G., Wu, Q., Zhou, B., Li, J., Wang, W., Ding, G. and Yau, D.K.Y. (2025) ‘Spectrum Prediction With Deep 3D Pyramid Vision Transformer Learning’, IEEE Transactions on Wireless Communications, 24(1), pp. 509–525.
- Qiu, K., Zheng, S., Zhang, L., Lou, C. and Yang, X. (2024) ‘DeepSIG: A Hybrid Heterogeneous Deep Learning Framework for Radio Signal Classification’, IEEE Transactions on Wireless Communications, 23(1), pp. 775–788.
- Rajendran, S., Meert, W., Giustiniano, D., Lenders, V. and Pollin, S. (2018) ‘Deep Learning Models for Wireless Signal Classification With Distributed Low-Cost Spectrum Sensors’, IEEE Transactions on Cognitive Communications and Networking, 4(3), pp. 433–445.
- Rajesh, R., Darak, S.J., Jain, A., Chandhok, S. and Sharma, A. (2022) ‘Hardware–Software Co-Design of Statistical and Deep-Learning Frameworks for Wideband Sensing on Zynq System on Chip’, TechRxiv preprint.
- Skokowski, P., Malon, K. and Łopatka, J. (2022) ‘Building the Electromagnetic Situation Awareness in MANET Cognitive Radio Networks for Urban Areas’, Sensors, 22(3), p. 716.
- Soni, B., Patel, D.K., Shah, S.B., Lopez-Benitez, M. and Govindasamy, S. (2022) ‘PU-DetNet: Deep Unfolding Aided Smart Sensing Framework for Cognitive Radio’, IEEE Access, 10, pp. 98737–98751.
- Tekbiyik, K., Akbunar, Ö., Ekti, A.R., Görçin, A. and Karabulut Kurt, G. (2019) ‘Multi-Dimensional Wireless Signal Identification Based on Support Vector Machines’, IEEE Access, 7, pp. 138890–138903.
- Toma, O.H. and Lopez-Benitez, M. (2021) ‘Traffic Learning: A Deep Learning Approach for Obtaining Accurate Statistical Information of the Channel Traffic in Spectrum Sharing Systems’, IEEE Access, 9, pp. 124324–124336.
- Wasilewska, M. and Bogucka, H. (2020) ‘Space-Time-Frequency Machine Learning for Improved 4G/5G Energy Detection’, International Journal of Electronics and Telecommunications, 66(1), pp. 217–223.
- Wen, X. and Li, W. (2023) ‘Time Series Prediction Based on LSTM-Attention-LSTM Model’, IEEE Access, 11, pp. 48322–48331.
- Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H. and Zhang, W. (2021) ‘Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting’, Proceedings of the AAAI Conference on Artificial Intelligence, 35(12), pp. 11106–11115.
- Zhu, Y., Li, H., Liao, Y., Wang, B., Guan, Z., Liu, H. and Cai, D. (2017) ‘What to Do Next: Modeling User Behaviors by Time-LSTM’, Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), pp. 3602–3608.