Attention-Based Radar PRI Modulation Recognition With Recurrent Neural Networks

XUEQIONG LI[ID]**, ZHANGMENG LIU**[ID]**, AND ZHITAO HUANG**
Department of Electrical Science, National University of Defense Technology, Changsha 410073, China

Corresponding author: Xueqiong Li (lixueqiong13@nudt.edu.cn)

ABSTRACT Analyzing radar signals is a critical task in modern Electronic Warfare (EW) environments. However, the pulse streams emitted by radars have flexible features and complex patterns which are difficult to be identified from a statistical perspective. To solve this problem, pulse repetition interval (PRI) is used as a distinguishing parameter of emitters to be identified. However, traditional PRI modulation recognition methods can only deal with simple PRI modulations and their performance will further degrade with the increasing number of emitters or noisy environments. In this paper, we introduce an attention-based recognition framework based on recurrent neural network (RNN) to categorize pulse streams with complex PRI modulations and in environments with high ratios of missing and spurious pulses. Simulation results show that our model is robust to noisy environments and has a better performance than conventional methods.

INDEX TERMS Attention mechanism, electronic warfare, PRI modulation, recurrent neural network (RNN).

I. INTRODUCTION

With the rapid development and usage of advanced communications [1], [2], navigation [3]–[5] and radar systems [6], [7], the Electronic Warfare (EW) environment is more crowded nowadays. When there are signals transmitted by multiple radars simultaneously, an interleaved stream of pulses is received by an electronic support system. The following task is to separate these signals [8], [9] and thus to identify the source emissions [10].

In most previous literature, these signals are described by Pulse Description Words (PDWs), which can be processed statistically [11]–[14]. The PDWs contain several statistical features including pulse width (PW), carrier frequency (CF), pulse amplitude (PA), time of arrival (TOA), direction of arrival (DOA) [13]. However, not all of those features can be easily measured and used directly for signal analysis due to the crowded electromagnetic environment and changing PDW patterns. By observing a large amount of measurement data, it was found that TOA is a reliable feature of PDWs,

The associate editor coordinating the review of this manuscript and approving it for publication was Weimin Huang[ID].

which is stable and easy to detect [12]. In this case, the TOA is the first choice for analyzing signals.

Pulse Repetition Interval (PRI), which is the first-order difference of the TOA, presents the intrinsic patterns of the TOA sequences. PRI can be obtained by calculating the difference between two adjacent TOA values. PRI is a distinguishing feature since it indicates the periodicity of pulse streams emitted by the radars. The intentionally or unintentionally change of this parameter indicates a specific mission requirement. Therefore, recognizing PRI modulation is crutial for the identification of the emitter and its working pattern. However, with the diversification of radar functions and the need for anti-reconnaissance and anti-jamming, PRI modulations have become quite complex, so they are more difficult to be identified by common statistical methods [13].

Conventional PRI modulation recognition is typically performed by using a histogram of the pulse intervals of the radar signals [13]. Those methods can recognize PRI modulation as long as the amount of data is large enough and the PRI patterns are simple. Several intelligent recognition methods have been proposed in recent years. The first fully automatic method proposed in [15] uses multilayer perceptron (MLP) to recognize PRI modulation, which has good performance

but needs to extract high dimensional features as input to the MLP. Some other methods in [16]–[18] attempt to seek for features of PRI sequences and identify PRI modulation through these features. These methods can achieve better error recognition rates than the Noone's method [15] in the case of lost and spurious pulses, but they have a large performance degradation as the noise ratio increases because the features will lose their regularity. Kauppi *et al.* [19] utilized a two-stage hierarchical classification scheme and extracted subpatterns to identify the PRI modulation mode. This method seeks for features of PRI sequences to distinguish different types of PRI, but it cannot work when there are high ratios of noise which breaks the regularity of the subpatterns. In [20], an algorithm based on auto-correlation and normalization is proposed, which is robust to noise pulses but susceptible to pulse loss. A classification method based on convolutional neural network (CNN) is proposed in [21]. Although it can achieve better performance in a complex environment, it can only be realized after collecting all the pulses, which greatly hinders online applications due to CNN limitations. The latest research is [22], which uses several neural networks to recognize PRI modulation modes including MLP, CNN, Long-Short Term Memory (LSTM), etc. This thesis finds that LSTM can be more accurate than other models. However, the lost pulse ratio considered in this thesis is not large enough for many situations in modern electromagnetic environment (the loss rate is 25% in this thesis). In this case, a new approach capable of classifying complex PRI modulation modes in a real EW environment with high ratios of lost and spurious pulses is required.

In this paper, we propose to use the recurrent neural networks [23] and the attention mechanism [24] to solve the PRI modulation recognition problem. There have been a large number of studies on RNN, and much progress has been made since the proposal of LSTM in 1997 [25]. LSTM introduces a forgetting mechanism in the original RNN framework. In this case, the network architecture is modified so that the vanishing gradient problem is explicitly avoided. This progress has contributed greatly to the widespread applications of RNN [26]. Recently, gated recurrent units (GRU) [27] has been proposed as an alternative to LSTM. The GRU has no controlled exposure of the memory content that LSTM has. Bahdanau *et al.* [28] report that the two units performed comparably to each other according to their preliminary experiments on machine translation, but GRU is more computationally efficient with less complex structure. In this case, the GRU structure is used in this paper to solve the problem of PRI modulation recognition. In addition, considering the characteristics of the PRI sequence, the attention mechanism is also applied to the proposed model in order to better adapt to the complex environment of EW. The attention mechanism helps the network focus on specific areas and ignore other unwanted parts. It has been a long time since the attention mechanism was applied to neural networks, especially in image recognition [24]. Until recently, attention mechanisms have made their way into recurrent

neural network architectures for Natural Language Processing (NLP). [24] uses soft and hard attention mechanism to solve image caption generation problems. In [29], attention model is utilized for text automatic summarization. In [30], a hierarchical attention network for document classification was established. In [31], the pooling network based on the attention mechanism is employed in a question answering system. A more widely used application is described in [32], where RNN and attention mechanisms are used to solve a translation task. Later, [33] proposes a structure of GRU and attention mechanism applied in translation.

With the above motivations, we introduce an attention-based recognition framework based on recurrent neural network (RNN) to categorize pulse streams with complex PRI modulations and in environments with high ratios of missing and spurious pulses. The RNN is used to process TOA sequences and learn the intrinsic regularity of the time series, and attention mechanism is used because it can help the RNN model focus on the correct pulses and ignore the noise. During the training phase, all network parameters are automatically tuned based on the input PRI sequence and the output ground truth, and the trained network will output the corresponding PRI modulation mode when the test sequence is imported. No expert knowledge is required during the process of training and validation phases. In this way, the proposed model is capable of dealing with complex PRI modulations in noisy environments. Experiment results illustrate that the proposed attention-based RNN model can recognize six complex PRI modulation modes with high ratios of spurious and lost pulses.

The rest of the paper is organized as follows. In section II, we introduce PRI modulation modes and noise in the EW environment; in Section III, the attention-based RNN model is proposed in detail; Section IV shows the simulation results, and conclusions are finally presented in Section V.

II. PROBLEM FORMULATION

A. REPRESENTATION OF PRI

The PRI sequence p_n consists of some instant PRI values which can be described as [34]:

$$p_n = \{p_1, p_2, \ldots, p_i, \ldots, p_{N-1}\}, \tag{1}$$

where N is the number of intercepted pulses.

Since the PRI sequence is the first difference of TOA sequence, by observing the TOA sequence, p_i can be calculated by $p_i = t_{i+1} - t_i$, where t_i is the ith intercepted pulse.

Each digitized PRI value p_i can be represented by a one-hot vector $\mathbf{g}_{pri}$, whose unique non-zero element of **1** indicating the value of the digitized features. For example, if the PRI is upper-bounded by $5000\mu s$ and digitized with a unit of $5\mu s$, then the one-hot representation of $6.7\mu s$ is $[0, 1, 0, 0, \cdots, 0]^T \in \mathbb{R}^{1001\times1}$. One-hot features can be processed more easily by machine learning techniques than their numerical counterparts [34].

B. EMBEDDING OF THE PRI SEQUENCES

One-hot features can well represent the PRI values while the elements are only zeros and ones, which are too sparse and could make the learning process unstable. Therefore, embedding ideas [35] are introduced to compress the one-hot expression and stabilize the learning process.

An embedding turns positive integers into dense vectors of fixed size. In the context of neural networks, embeddings are low-dimensional, learned continuous vector representations of discrete variables. Embeddings are useful because they can reduce the dimension of categorical variables and meaningfully represent categories in the transformed space [35].

In recent years, embedding technique has proven to perform well in a variety of areas especially in NLP tasks [35]–[37].

According to the embedding technique, the one-hot PRI features can be converted as follows,

$$\mathbf{e}_{pri} = \mathbf{E}^{(pri)} \mathbf{g}_{pri}, \qquad (2)$$

where $\mathbf{g}_{pri} \in \mathbb{R}^{1 \times l}$ is one-hot PRI vector, $\mathbf{E}^{(pri)} \in \mathbb{R}^{L \times 1}$ is embedding matrix for the feature with $l \ll L$, $\mathbf{e}_{pri} \in \mathbb{R}^{L \times l}$ is embedded vector.

Embedding matrix acts like look-up table. When a one-hot feature is given, a column of the matrix is selected according to the position of the non-zero vector element to represent the feature. The embedded features are then fed to the neural network as a series of inputs. Well designed neural networks are required to extract inner patterns within successive pulses, so as to identify different emitters, and be aware of pulse contexts to distinguish pulses from outliers.

The parameter matrix $\mathbf{E}^{(pri)} \in \mathbb{R}^{L \times 1}$ of the embeddings are initialized properly and trained together with other parameters during the training phase of the neural network via supervised learning, and the detailed training progress will be discussed in Section III.

C. PRI MODULATION MODES

There are commonly six basic types of PRI modulation, i.e., constant PRI, staggered PRI, sliding PRI, jittered PRI, Dwell and Switch (D&S) PRI and wobulated PRI. These six types of PRI sequences can be described as follows [11], [15].

Constant PRI: A constant PRI sequence always has a fixed value of k. $p_n = k, \forall n \in \mathbb{Z}+$ for some real number $k > 0$.

Staggered PRI: There are M fixed values appear cyclically in order in a stagger PRI sequence. $p_n = y_i$ where each $y_i > 0$ is a pre-defined value and $i = (n \bmod M)$.

Sliding PRI: The values of sliding PRI sequences always monotonously increase (or decrease) to the maximum (or the minimum) value, and then suddenly go down (or up) to the minimum (or the maximum) value. $p_n = PRI_0 + \delta * (n \bmod M)$ where $PRI_0 > 0$ is the initial PRI value of a slide period, $\delta \in \mathbb{R}$ is some value indicating the rate of change in PRI during a slide period and M is the number of pulses in each slide window.

Jittered PRI: The values of jittered PRI sequences jittered around a certain constant value, and the range of jitter is typically between 1% and 30%, which is random but typically follows a Gaussian distribution.

Wobulated PRI: The values of wobulated PRI sequence always have a shape similar to the sinusoidal function, and also appear periodically. $p_n = PRI_0 + A \sin(\omega * x_n + \varphi)$ where $PRI_0 > 0$ is the value that PRI will oscillate around. A is the amplitude of the modulation, ω is the frequency of the sine function, φ is the phase of the sine function, and x_n is some value proportional to n defining the sampling resolution.

Dwell and Switch (D&S) PRI: Like the staggered PRI, there are several stable values in the D&S PRI sequence, but the difference is that the values may remain at the same value $y_i > 0, \forall i$ for several pulses $x_i - x_{i-1}$ before switching to another value $y_{i+1} > 0$ in the D&S PRI.

D. LOST AND SPURIOUS PULSES

The lost and spurious pulses of the pulse streams make it difficult to recognize PRI modulation. Lost pulses are happens because the ESM system does not detect every pulse emitted by a radar [38], [39], especially when the antenna is facing away from the radiation sources. Besides missing pulses, there are also many spurious pulses in received signals because other emitters are transmitting signals simultaneously. Advanced signal transmitters and crowded electromagnetic environments are the main causes of lost and spurious pulses, which always occur during the reception of radar pulses. In addition, errors made in the deinterleaving step will result in lost and spurious pulses.

Let $p_n = \{p_1, p_2, \ldots, p_{N-1}\}$ denote the original PRI sequence, the received TOA sequence can be described as $T_n = \{t_1, t_1 + p_1, \ldots, t_1 + p_1 + \ldots + p_{N-1}\}$, where N is the total number of received pulses.

When there are j pulses are lost from the ith pulse to $(i + j - 1)$th pulse in the received TOA sequence T_n, then the new PRI sequence p'_n can be represent as:

$$p'_n = \begin{cases} p_n, & n = 1, 2, \ldots, i-1 \\ p_i + \ldots + p_{i+j-1}, & n = i \\ p_{n+j}, & n = i+1, \ldots, N-j-1. \end{cases} \qquad (3)$$

Besides, if there are j pulses added to the sequence following the ith pulse, which have the first-order difference as $g_k = \{g_0, g_1, \ldots, g_j\}$, then the new PRI sequence p''_n can be described as:

$$p''_n = \begin{cases} p_n, & n = 1, 2, \ldots, i-1 \\ g_{n-i}, & n = i, \ldots, i+j \\ p_{n-j}, & n = i+j+1, \ldots, N+j-1, \end{cases} \qquad (4)$$

where j is the number of inserted pulses. g_0 and g_j are the differences between the original pulse and the spurious pulse. g_1 to g_{j-1} is the difference between the spurious pulses themselves.
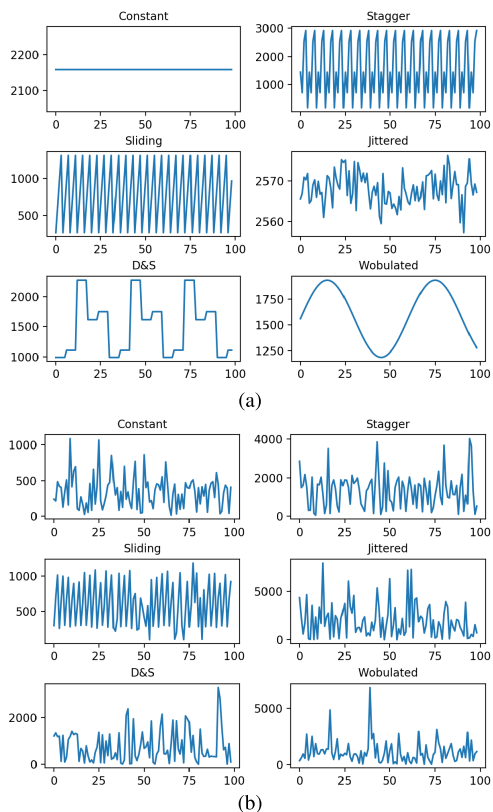
FIGURE 1. Six curves of PRI modulation modes in (a) ideal environment; (b) real EW environment.

In addition, measurement error happens when receiving the signals, which typically follows a Gaussian distribution with the mean value is the value of PRI sequence.

In an ideal EW environment without lost or spurious pulses, the PRI modulation mode seems to be easily identified by distinguishing regular patterns. However, when noise is added to the curves, it is difficult to tell which modulation mode the PRI sequence belongs to.

Six kinds of PRI modulation curves in an ideal environment without noise are shown in Fig. 1a. The same modulation modes in the real EW environment with 70% of lost pulses and 50% of spurious pulses are shown in Fig. 1b. The figure shows that it is no longer easy to recognize these PRI modulation modes using traditional statistical methods.

Considering the characteristics of the above PRI sequences, we introduce an attention-based RNN structure to solve the PRI modulation recognition problem. This is because i) RNN is suitable for processing time-series data; ii) RNN can take sequential input which can perform on-time processing; iii) The attention mechanism can well solve the problem of high ratios of lost and spurious pulses.

III. RNN BASED MODULATION RECOGNITION MODEL
In this section, we present the idea of classifying PRI sequences using an attention-based RNN model. A sketch of the proposed attention-based RNN structure is shown in Fig. 2. PRI sequence $p_n = \{p_1, p_2, \ldots, p_{N-1}\}$ is first
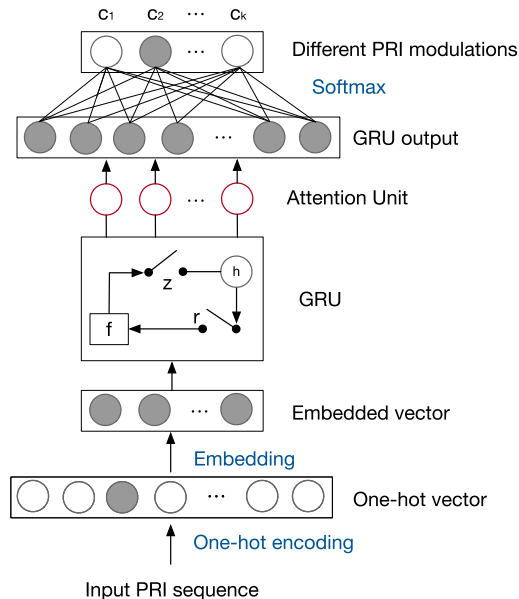

FIGURE 2. The framework of the PRI modulation recognition process.
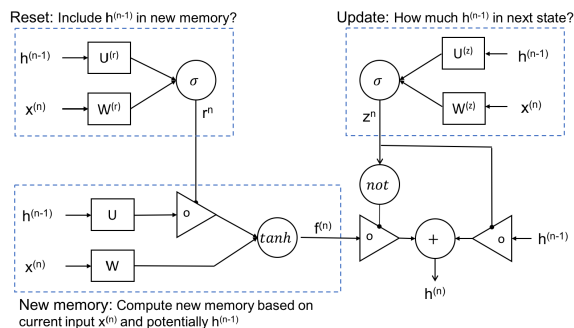

FIGURE 3. Detailed procedures of GRU.

represented by a one-hot vector, and then embedded into the lower-dimensional features according to Eq. (2). The embedded features are then fed into the attention-based GRU model. This model extracts sequential patterns contained in the PRI sequence, and stores them in the state of the GRU, which is denoted by $\mathbf{h}_n$, and is also considered the output of the model. The parameters of this process need to be tuned during the training phase [27]. Finally, a fully-connected layer is appended to the model to map its state vectors to a probability distribution along with different classes, with softmax acting as the activation function.

Detailed discussions of the structure and the stages of training and validation of the proposed model will be provided in the following sections.

A. RNN FOR MODULATION RECOGNITION
In order to take long PRI series into consideration, we use GRU [27] to promote the performance of RNN. Detailed procedures of GRU are shown in Fig. 3 and Eqs. (5)-(8).

$$\mathbf{z}_n = \sigma(\mathbf{W}^{(u)}\mathbf{x}_n + \mathbf{U}^{(u)}\mathbf{h}_{n-1} + \mathbf{b}^{(u)}), \qquad (5)$$

$$\mathbf{r}_n = \sigma(\mathbf{W}^{(r)}\mathbf{x}_n + \mathbf{U}^{(r)}\mathbf{h}_{n-1} + \mathbf{b}^{(r)}), \qquad (6)$$

$$\mathbf{f}_n = \tanh(\mathbf{W}\mathbf{x}_n + \mathbf{r}_n \odot \mathbf{U}\mathbf{h}_{n-1} + \mathbf{b}), \qquad (7)$$

$$\mathbf{h}_n = \mathbf{z}_n \odot \mathbf{h}_{n-1} + (1 - \mathbf{z}_n) \odot \mathbf{f}_n. \qquad (8)$$

There are four main vectors in the GRU structure, they are the update gate $\mathbf{z}_n$, the reset gate $\mathbf{r}_n$, the new memory $\mathbf{f}_n$ and the hidden state vector $\mathbf{h}_n$ at time instant n. Other parameters include the input $\mathbf{x}_n$, the bias vector $\mathbf{b}$ and the weight matrices $\mathbf{W}$ and $\mathbf{U}$. In Eq.(5), the update gate $\mathbf{z}_n$ decides how much of $\mathbf{h}_{n-1}$ will pass to the next state. The reset gate identifies how important $\mathbf{h}_{n-1}$ is to the new memory $\mathbf{f}_n$ in Eq.(6). These two vectors are obtained via a logistic sigmoid function $\sigma(\cdot)$ by the input $\mathbf{x}_n$ and the hidden state $\mathbf{h}_{n-1}$ transformed linearly with respect to the matrices $\mathbf{W}$ and $\mathbf{U}$, and then added with a bias vector $\mathbf{b}$. In Eq.(7), the memory vectors $\mathbf{f}_n$ are also obtained by $\mathbf{W}$, $\mathbf{U}$ and $\mathbf{b}$ but via a hyperbolic tangent function $\tanh(\cdot)$. The hidden states of GRU $\mathbf{h}_n$ are finally updated by the other three vectors mentioned above in Eq.(8). $\mathbf{1}$ is an all-one vector in the equation, and $\odot$ stands for element-wise multiplication.

There is a fully connected layer following the GRU structure to obtain the probability distribution over the PRI class. Assuming that the K PRI modulation modes is to be recognized, the probability $\hat{\mathbf{p}}$ of the input PRI sequences can be described as follows:

$$\hat{\mathbf{p}} = s(\mathbf{W}^{(o)}\mathbf{h}_n + \mathbf{b}^{(o)}), \qquad (9)$$

where $\mathbf{W}^{(o)}$ are weight matrices, $\mathbf{b}^{(o)}$ are bias vectors, and $s(\cdot)$ is the softmax function.

Each element of the output vector $\hat{\mathbf{p}}$ represents the probability that the PRI sequence belongs to a certain class. The class with the greatest probability is then selected as the classification result.

B. ATTENTION MECHANISM

Intuitively, the periodicity of the PRI sequence makes the PRI values repeatedly appear, so the PRI values within the complete period can be used to determine its type. In addition, due to the spurious and lost pulses in the PRI sequence, only part of the PRI values belong to the original PRI sequence, which can be used for the recoginition. Other wrong PRI values cannot contribute to the calculation of class labels. However, GRU lacks the ability to adaptively focus on certain areas or locations, so it may result in redundancy or lost information during learning. Therefore, an attention mechanism has been introduced to help the RNN focus on the pulses that contribute to the recognition.

The attention mechanism in neural networks is based on the human attentional visual mechanism [40]. Human visual attention has been well studied. Although there are different models, they all come down to being able to focus on a certain area of an image with "high resolution" while perceiving the surrounding with "low resolution". Then they adjust the focal point over time. As with PRI modulation, it provides a larger weight value for certain pulses of the entire sequence and ignores those pulses with incorrect values.
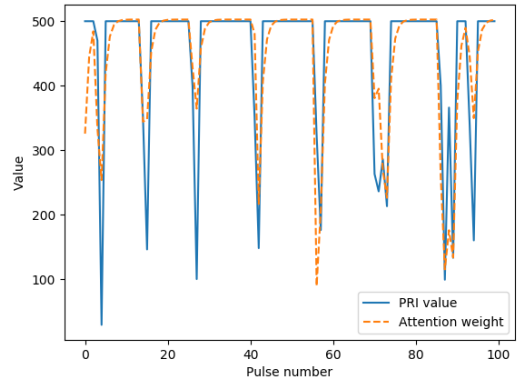

FIGURE 4. The PRI values and corresponding attention weights.

In our proposed RNN structure, the attention mechanism can well learn the alignment between the input PRI sequence and the output PRI modulation mode. The attention mechanism links to the related parts of the input PRI sequence and then assigns a higher weight to the corresponding pulses of the input $\mathbf{x}_n$ [41], [42].

The attention gate receives the sequence information x_n and hidden state at the last moment h_{n-1} to learn a weight matrix $\mathbf{A}_n$ that can express the importance of these information.

$$\alpha_n = \tanh(\mathbf{h}_{n-1}, \mathbf{x}_n). \qquad (10)$$

$$\mathbf{A}_n = \frac{\exp(\alpha_n)}{\sum_n \exp(\alpha_n)}, \qquad (11)$$

$$\hat{\mathbf{h}}_n = \tanh(\mathbf{A}_n[\mathbf{x}_n, \mathbf{h}_{n-1}]), \qquad (12)$$

where $\mathbf{x}_t$ is the input vector at time n, and $\mathbf{h}_{n-1}$ is the hidden state at last moment. In the above formula, the attention mechanism can be considered to construct a fixed length of the embedded value $\hat{\mathbf{h}}_n$ of the input sequence by calculating an adaptive weighted average of the state sequences $\mathbf{h}_{n-1}$ and input $\mathbf{x}_n$.

In order to know the meaning of the attention mechanism, a constant PRI is taken as an example. PRI sequences with constant PRI and other PRI modulation modes are imported into the proposed RNN structure for training. After the RNN converges, a constant PRI sequence with several spurious pulses and their respective attention weights α_i are drawn in Fig. 4. Since the value of the attention weight is between $[-1, 1]$, we take the absolute value of attention weight, and multiply it by a proper positive number to make the regularity more obvious. Here, the closer the absolute value is to 1, the more important the corresponding part is. Vice versa, if the absolute value is to 0, the corresponding value of PRI is not important.

The blue line is the curve of the constant PRI sequence. If there were no spurious or missing pulses, the line would remain constant at the value of 500, which means that the falling parts of the curve are where the spurious pulses are located (the lost pulses are not considered here). The orange line represents the corresponding attention weights for the

PRI values. The greater the attention weight is, the greater the contribution of the corresponding PRI value to the recognition results. Since spurious pulses are not conducive to recognition results, the corresponding attention weights should be small. Vice versa, the attention weights of the normal value of PRI at 500 should be large. The results are shown in Fig. 4, where the falling parts of the orange line exactly match the blue line.

C. TRAINING OF PROPOSED RNN MODEL

Since several weight matrices and bias vectors are introduced into the attention-based RNN model, they should be calculated by supervised learning. To achieve this, the tagged PRI sequences are fed into the RNN structure. The predicted class of the PRI sequence is calculatd through the RNN model, and then compared to the previously given labels to obtain a loss through the loss function. The loss indicates the difference between the estimated and target class labels. Then, the weight matrices are calculated by minimizing the loss through the back-propagation process. After multiple back-propagations, the trained GRU can converge to perform a satisfactory classification.

The parameters should be initialized before the training phase. In our proposed RNN model, the hidden state $\mathbf{h}_0$ is initialized to $\mathbf{0}$, while all the weight matrices and bias vectors are initialized randomly. After that, PRI sequence samples are imported to the proposed model. The PRI sequences are first digitized and converted to one-hot vectors to accommodate the machine learning algorithm. The one-hot vectors are then embedded to form the lower-dimensional input vectors $\mathbf{x}_n$ of the RNN model. The RNN model processes the input vectors according to Eq.(5)-(8) and finally outputs the state vector $\mathbf{h}_n$. When processing the last sample of the PRI sequence, putting the final state vector into the fully-connected layer to obtain a probability distribution estimate $\hat{\mathbf{p}} = [\hat{\mathbf{p}}_1, \ldots, \hat{\mathbf{p}}_K]^T$ on the K classes.

The target label of the PRI sequence is $\mathbf{p} = [0, \ldots, 0, 1, 0, \ldots, 0]^T \in \mathbb{R}^{K \times 1}$, and the only non-zero element in $\mathbf{p}$ represents the class number. The loss function used in our RNN model follows the binary cross entropy criterion [43] to calculate the differences between the expected values and the true labels:

$$loss = -\sum_{k=1}^{K}[p_k log(\hat{p_k}) + (1 - p_k)log(1 - \hat{p_k})]. \quad (13)$$

From Eq. (13) we can inform that, the greater the loss, the more $\hat{\mathbf{p}}$ deviates from $\mathbf{p}$. Our goal is to make the loss as small as possible. When the average loss of a batch of data remains stable and the recognition accuracy in the test dataset is satisfying, the model is considered as well-trained.

The back-propagation process [43] is always used in neural networks to minimize the loss value. It calculates the derivations between the estimated values and the target values, and then modifies the weight matrices to converge the neural network parameters. The back-propagation process can be

described as follows:

$$\alpha_{new} = \alpha_{old} - \eta \frac{\partial loss}{\partial \alpha}, \quad (14)$$

where α represents any one of the tunable parameters, η is a self-defined positive learning rate which is smaller than $\mathbf{1}$. Many literature have studied the details of back-propagation of RNN [44], [45]. The most widely used machine learning platforms, such as Pytorch [46] and Tensorflow [47], provide callable functions to perform the back-propagation process automatically.

Parameter settings and dataset descriptions used to train and test the RNN classifier are introduced in Section IV. Lost and spurious pulses are considered in the dataset to better simulate the actual EW environment.

IV. SIMULATIONS

In this section, we perform simulations to demonstrate the performance of attention-based RNN recognition for six PRI modulation modes. The PRI sequences considered in this paper have a high noise ratio, so it is difficult to distinguish them using statistical features. Therefore, other neural networks including deep neural network and convolutional neural networks are considered to be base-line methods for performance comparison.

A. SIMULATION SETTINGS

1) PARAMETERS OF PRI SEQUENCES

Six kinds of PRI sequences are simulated in this section. All parameters are simulated based on conditions that may be encountered in a real EW environment.

The parameters for each PRI modulation are chosen such that the PRI values will always at a particular range that is most likely to be seen in the real environment. For each PRI sequence, the pulse length N is set to 300. 40, 000 samples are simulated separately for each PRI modulation mode, which means that a total of 240, 000 PRI sequences are simulated. Specifically, 80% of 240, 000 streams are generated to train the proposed RNN structure, 10% for test set and the other 10% of the streams are used as the validation set. The test set are simulated with different levels of lost pulse ratio or spurious pulse ratio, and each level has 1, 000 sequences. Detailed parameter settings for each PRI modulation mode are described below.

Constant PRI: Constant PRI is defined by $PRI = random(min\_val, max\_val)$, which means ideal constant PRI sequence will always maintain the same value between the maximum value max_val and minimum value min_val. In the simulation, min_val is set to 100 and max_val is set to 3, 000.

Jittered PRI: The values of jittered PRI sequence jittered around a certain constant value $PRI_0 = random(min\_val, max\_val)$, and the range of jitter is $deviation = random(min\_dev, max\_dev)$. So the values for jittered PRI sequence are generated by sampling $PRI = PRI_0 * (1 - deviation), PRI_0 * (1 + deviation)$. Table 1 shows the values of those parameters.

TABLE 1. Parameters of simulated jittered PRI.

Parameters	Value	Description
min_val	100	Mean value of PRI
max_val	3000	
min_dev	3%	Deviation from mean PRI value
max_dev	30%	

TABLE 2. Parameters of simulated stagger PRI.

Parameters	Value	Description
min_val	100	Mean value of PRI
max_val	3000	
min_len	3	Number of values for each window
max_len	10	

TABLE 3. Parameters of simulated sliding PRI.

Parameters	Value	Description
min_start	300	Start value of sliding PRI window
max_start	500	
min_k	300	Gradient of the sliding PRI sequence
max_k	700	
min_len	3	Number of pulses in each sliding window
max_len	10	

TABLE 4. Parameters of simulated wobulated PRI.

Parameters	Value	Description
min_val	1000	Mean value of PRI
max_val	2000	
min_A	100	Deviation from mean PRI value
max_A	500	
min_ω	3	Frequency of the sinusoidal function
max_ω	12	
min_φ	0	Phase of the sinusoidal function
max_φ	2	

Stagger PRI: There are $M = random(min\_len, max\_len)$ fixed values appear cyclically in order in a stagger PRI sequence. For each element of one stagger period, the value is $PRI = random(min\_val, max\_val)$. Table 2 shows the values of those parameters.

Sliding PRI: Sliding PRI uses a start value $PRI_0 = random(min\_start, max\_start)$ and a gradient $k = random(min\_k, max\_k)$ to simulate the sequence. The PRI values of one sliding period can be calculated by $PRI = PRI_0 + k * n, n = 1, 2, \cdots, N$ where $N = random(min\_len, max\_len)$ is the number of PRI value. Table 3 shows the values of those parameters.

Wobulated PRI: Wobulated PRI is defined by a sinusoidal function $PRI = PRI_0 + A\sin(\omega * n + \varphi), n = 1, 2, \cdots$. $PRI_0 = random(min\_val, max\_val)$ is the mean value of wobulated PRI. $A = random(min\_A, max\_A)$ is the deviation from mean PRI value. $\omega = random(min\_\omega, max\_\omega)$ and $\varphi = random(min\_\varphi, max\_\varphi)$ the frequency and phase of sinusoidal function respectively. The detailed parameter settings are list in table 4.

TABLE 5. Parameters of simulated D&S PRI.

Parameters	Value	Description
min_val	100	Mean value of PRI
max_val	3000	
min_D	3	Number of switches per window
max_D	10	
min_S	5	Number of pulses before switching
max_S	20	

D& S PRI: Dwell and switch PRI sequence consists of several PRI values $PRI = random(min\_val, max\_val)$. For one D&S PRI window, there are $D = random(min\_D, max\_D)$ different PRI values, and each of them will remain for $S = random(min\_S, max\_S)$ times. This will define a window, this window is then repeated until the total number of PRI values reach $N = 300$. Table 5 shows the values of those parameters.

2) ERRORS IN EW ENVIRONMENT

There are three kinds of noise that are considered during the training and the validation phases, namely the lost pulses, the spurious pulses and the measurement error [48]. Each sequence is assumed to drop a certain probability $\rho_m = \frac{\sum_i a_i}{\sum_i a_i + b_i}$ of pulses, where a_i is the number of lost pulses in the ith period, and b_i represents the remaining pulses in the ith period. Spurious pulses are added between two adjacent pulses with their number subjecting to a Poisson distribution with a mean of $\rho_n(1 - \rho_m)$. In addition, measurement errors are added to the sequences following a Gaussian distribution. In this way, the ratio of noise number to pulse number is guaranteed to be ρ_n in average in the streams. During the training of RNN, ρ_m is set from 0 to 0.7, and ρ_n is set from 0 to 0.5. The standard deviation of the measurement error is set from 0 to 0.1. Different values of ρ_m and ρ_n are selected for test to show the robustness of the trained networks.

Due to the overlap values and high ratio of lost and spurious pulses, these PRI modulations are difficult to be categorized directly by conventional statistical methods.

3) PARAMETERS OF PROPOSED RNN STRUCTURE

The attention-based RNN model is trained to recognize PRI sequences in the six categories mentioned previously. The entire structure consists of four basic parts, namely embedding, attention, GRU and fully connected layers. For each training or validation sample, the detailed parameter settings for each component are described as follows.

Embedding: The length of the input PRI sequence is 300, and it is then transformed into a one-hot vector with shape of 300×10000. So the input size of the embedding layer is 300×10000 Embedding size is set to 128. After embedding, the input size becomes 300×128.

Attention mechanism: The attention weight is 300, so the input size of the RNN after attention is still 300×128.

TABLE 6. Different proportions of lost and spurious pulses for the test set.

	Proportion (%)
Lost pulses	0, 20, 50, 70
Spurious pulses	0, 10, 20, 30, 40, 50
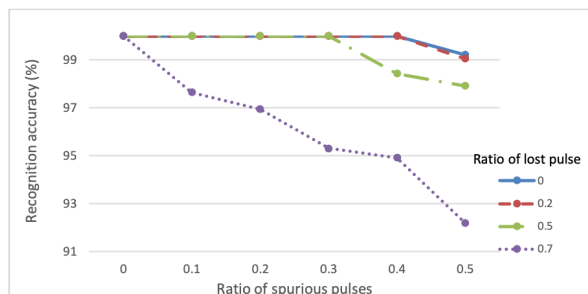
FIGURE 5. Recognition accuracy with different ratios of lost and spurious pulses.

GRU: There are three layers of the GRU we proposed. The hidden size of the GRU is set to 256, so the output size of the GRU is 300×256.

Fully connected layer: Six PRI modulation modes are to be classified, and the final output of the entire structure is 1×6.

The attention-based RNN structure is trained on the Pytorch platform with a batch size of 64 and a learning rate of $\eta = 0.001$. Each of the batch is selected randomly from the corresponding dataset, and 3, 750 batches in total are used to train the network.

B. RESULTS

During the training phase, all possible scenarios with a wide range of parameters of the noise-contaminated PRI modulation modes are fed into the proposed RNN model. The model is trained till converged. In the test, four different ratios of lost pulses and six ratios of spurious pulses for each PRI modulation mode are put into the well-trained model separately to understand how the proposed RNN model is performed in different environments. The selected values for lost and spurious pulse ratios are listed in Table 6. The classification performance is evaluated in terms of the detection probabilities, that is, the ratio of correct classification number to the total number of samples.

Fig. 5 shows the detection probabilities for all six classes with different noise environments. Each line indicates how the recognition accuracy varies with increasing spurious pulse ratio at the same lost pulse ratio. It can be concluded from Fig. 5 that the recognition accuracy is higher than 92% even in the worst case with 70% lost pulses and 50% spurious pulses (the pulse curves can be seen in Fig. 1b). The reason for the performance degradation is that the lost and spurious pulses cause huge damages to the original regularized PRI patterns, and make them indistinguishable from streams in other classes.
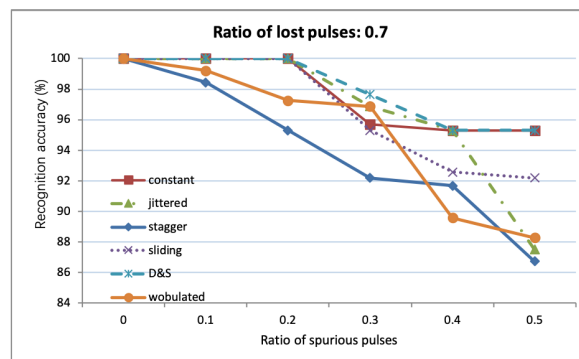
FIGURE 6. Six curves of PRI modulation modes in simulated EW environment.

In addition, to understand the recognition performance of different PRI modulation modes, several separate tests are performed under different conditions for different PRI modes. First of all, the lost pulse ratio is set to 0 and the spurious pulse ratio ranges from 0 to 50%. Only when the spurious pulse ratio is 50%, the recognition accuracy of the stagger PRI is 93.75%, and the accuracy of all other cases is 100%. Then the lost pulse ratio is set to 70% and the spurious pulse ratio is still ranging from 0 to 50%. The results are shown in Fig. 6. The results indicate that different modulation modes have different responses to lost and spurious pulse ratios. Stagger PRI modulation is easily identified as D&S PRI when there is a high ratio of spurious pulses. Constant and D&S PRI sequences are minimally affected by lost and spurious pulses. Even in the worst case, the accuracy of these two PRI modulation modes is still over 95%. However, wobulated, stagger and jittered PRI sequences are most affected by spurious pulses. It can achieve an accuracy of nearly 100% when the spurious pulses ratio is 0, while the accuracy is greatly reduced to around 86% with 50% spurious pulses ratio. This is because these three kinds of PRI sequences have a faster change in one cycle than the other three PRI modulation modes, so the lost and spurious pulses will make the PRI modes more difficult to be identified.

The other two neural network structures are compared to illustrate the superiority of the proposed RNN structure, namely the fully-connected (FC) neural network and the convolutional neural network (CNN). Attention mechanism is not used on FC or CNN. All three neural networks have the same input and number of layers. The difference is that the main layers are fully connected layers, convolutional layers or recurrent layers. The results are shown in Fig. 7. Fig. 7a shows the recognition accuracy with spurious pulse ratio of 50%, and Fig. 7b is the performance of lost pulse ratio of 70%. It can be seen from the results that with the increase of the lost pulses ratio or the spurious pulses ratio, the proposed RNN model can obtain higher recognition accuracy than the other two models.

The attention mechanism used in GRU scans the entire input PRI sequence and selects the most relevant region to extract the corresponding features of the PRI sequences.
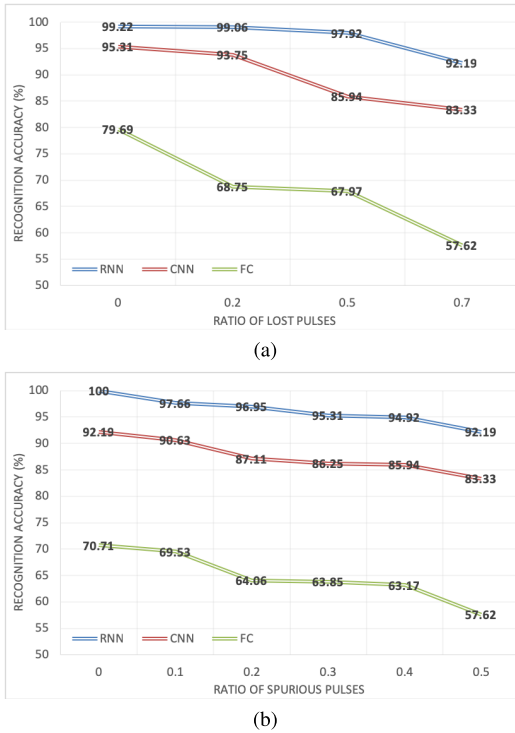
FIGURE 7. Recognition accuracy of different neural network structures with the increase of ratio of (a) lost pulses; (b) spurious pulses.
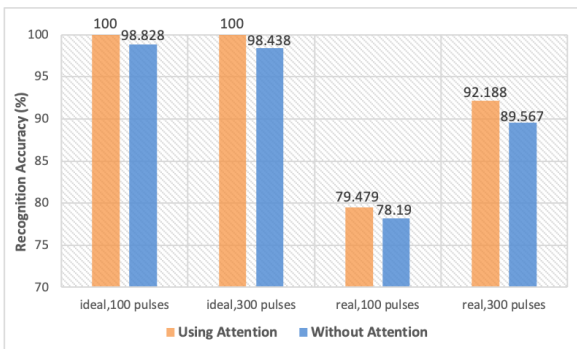

FIGURE 8. Recognition accuracy of PRI sequences in different environment with different length of pulses.

The use of attention mechanism can improve recognition performance, and the results of identifying all six PRI modulation modes under different conditions are shown in Fig. 8. Four different scenarios are simulated, an ideal environment and a real environment with a pulse length of 100 and 300. The results indicate that the attention mechanism can improve performance in different situations.

In an ideal environment, RNN is sufficient to identify those PRI modulations, but still the attention mechanism can promote the performance. This is because the attention mechanism adds a layer of neurons to the original RNN, which enhances the computing power of the network. In the environment with errors, the attention mechanism can increase recognition accuracy when the RNN cannot accurately classify the PRI modulation mode. This is because the PRI

pattern is hard to extract due to the high ratio of lost and spurious pulses. Meanwhile, when the number of pulses is small, the number of repetition periods of the PRI sequence is also small, which makes feature extraction more difficult. Furthermore, it helps the RNN to focus on the correct pulses and ignore the noise pulses. The attention probability denotes the alignment between the target class of the PRI and a local region of the input PRI sequence. It can also be considered as a regularization parameter for the GRU, as the attention helps to diminish the gradient of back-propagation from the classification process. Therefore, the entire process can use an attention mechanism to link to the relevant part of the input PRI sequence, and then assign a higher weight to the corresponding features.

Accordingly, our proposed attention-based RNN model not only classifies those PRI modulation modes in an ideal environment but is also robust to environments with high ratios of lost and spurious pulses.

C. ANALYSIS AND DISCUSSION
1) PROCESSING SEQUENTIAL DATA
RNN is the state of the art algorithm for sequential data. This is because it is the first algorithm that remembers its input, due to an internal memory, which makes it perfectly suited for machine learning problems involving sequential data. It is one of the algorithms behind the scenes of the amazing achievements of deep learning over the past few years. Because of their internal memory, RNNs are able to remember important information about the inputs they received, which allows them to process sequential data very accurately.

The PRI sequence is time-series data, which are just a series of data points listed in time order. The PRI pattern is learned from the regularity among pulses in a PRI sequence, which is how we recognize the PRI modulation mode. In this way, RNN is very suitable for PRI modulation recognition.

2) MEMORY-CENTERED
RNN intends to use the connections through a sequence of nodes to perform machine learning tasks associated with memory and clustering. GRU helps to adjust the neural network input weights to solve the vanishing gradient problem, which is a common issue with RNN. As a refinement of the general RNN structure, GRU has an update gate and a reset gate. Using these two vectors, the model optimizes the output by controlling the flow of information through the model. Like other types of recurrent network models, models with GRUs can retain information over time – which is one of the easiest ways to describe these types of techniques is that they are a ''memory-centered'' neural network type. In contrast, other types of neural networks without GRUs typically cannot retain information.

Since the PRI sequence is a long time sequence data and the pattern may require more than a few pulses to identify, the ''memory-centered'' neural network type is well suited for PRI modulation recognition.

V. CONCLUSION

In this paper, an attention-based RNN model is proposed for PRI modulation modes recognition. The model is trained through supervised learning and tested in various environments. The simulation results show that the model is able to mine and extract local patterns of streams of different feature types, and the mined patterns can be easily used to recognize the modulation mode of PRI sequences. Meanwhile, the proposed method solves the classification problems in complex environments with high ratio of lost and spurious pulses by utilizing attention mechanism to ignore the noise parts. Future work may include experiments on real data. Parameter estimation of the PRI sequences is also a direction.

REFERENCES

[1] H. Arslan, Ed., "Cognitive radio, software defined radio, and adaptive wireless systems," in *Signals and Communication Technology*. Amsterdam, The Netherlands: Springer, 2007.

[2] Y. Wang, M. Liu, J. Yang, and G. Gui, "Data-driven deep learning for automatic modulation recognition in cognitive radios," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 4074–4077, Apr. 2019.

[3] D. C. Robbins, R. K. Sarin, E. J. Horvitz, and E. B. Cutrell, "Advanced navigation techniques for portable devices," U.S. Patent 7 327 349, Feb. 5, 2008.

[4] Y. Jia, S. Li, Y. Qin, and R. Cheng, "Error analysis and compensation of MEMS rotation modulation inertial navigation system," *IEEE Sensors J.*, vol. 18, no. 5, pp. 2023–2030, Mar. 2018.

[5] Z. F. Syed, P. Aggarwal, X. Niu, and N. El-Sheimy, "Civilian vehicle navigation: Required alignment of the inertial sensors for acceptable navigation accuracies," *IEEE Trans. Veh. Technol.*, vol. 57, no. 6, pp. 3402–3412, Nov. 2008.

[6] J. Eaves and E. Reedy, *Principles of Modern Radar*. Springer, 2012.

[7] P. Kumari, J. Choi, N. Gonzalez-Prelcic, and R. W. Heath, Jr., "IEEE 802.11ad-based radar: An approach to joint vehicular communication-radar system," *IEEE Trans. Veh. Technol.*, vol. 67, no. 4, pp. 3012–3027, Apr. 2018.

[8] K. Nishiguchi and M. Kobayashi, "Improved algorithm for estimating pulse repetition intervals," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 36, no. 2, pp. 407–421, Apr. 2000.

[9] K. Gençol, A. Kara, and N. At, "Improvements on deinterleaving of radar pulses in dynamically varying signal environments," *Digit. Signal Process.*, vol. 69, pp. 86–93, Oct. 2017.

[10] K. Gençol, N. At, and A. Kara, "A wavelet-based feature set for recognizing pulse repetition interval modulation patterns," *TURKISH J. Electr. Eng. Comput. Sci.*, vol. 24, pp. 3078–3090, 2016.

[11] H. Mardia, "New techniques for the deinterleaving of repetitive sequences," *IEE Proc. F-Radar Signal Process.)*, vol. 136, no. 4, pp. 149–154, Aug. 1989.

[12] P. S. Ray, "A novel pulse TOA analysis technique for radar identification," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 34, no. 3, pp. 716–721, Jul. 1998.

[13] R. G. Wiley, *ELINT: The Interception and Analysis of Radar Signals*. Norwood, MA, USA: Artech House, 2006.

[14] L. Cheng, W. Wei, and W. Xuesong, "A method for extracting radar words of multi-function radar at data level," in *Proc. IET Int. Radar Conf.* Edison, NJ, USA: IET, 2013, pp. 1–5.

[15] G. P. Noone, "A neural approach to automatic pulse repetition interval modulation recognition," in *Proc. Inf., Decis. Control. Data Inf. Fusion Symp., Signal Process. Commun. Symp. Decis. Control Symp.*, Feb. 1999, pp. 213–218.

[16] J.-P. Kauppi and K. S. Martikainen, "An efficient set of features for pulse repetition interval modulation recognition," in *Proc. IET Int. Conf. Radar Syst.* Edison, NJ, USA: IET, 2007, pp. 1–5.

[17] A. Mahdavi and A. M. Pezeshk, "A robust method for PRI modulation recognition," in *Proc. IEEE 10th Int. Conf. SIGNAL Process. Proc.*, Oct. 2010, pp. 1873–1876.

[18] Y. Liu and Q. Zhang, "An improved algorithm for PRI modulation recognition," in *Proc. IEEE Int. Conf. Signal Process., Commun. Comput. (ICSPCC)*, Oct. 2017, pp. 1–5.

[19] J.-P. Kauppi, K. Martikainen, and U. Ruotsalainen, "Hierarchical classification of dynamically varying radar pulse repetition interval modulation patterns," *Neural Netw.*, vol. 23, no. 10, pp. 1226–1237, Dec. 2010.

[20] Z. Shi, H. Wu, W. Shen, S. Cheng, and Y. Chen, "Feature extraction for complicated radar PRI modulation modes based on auto-correlation function," in *Proc. IEEE Adv. Inf. Manage., Communicates, Electron. Autom. Control Conf. (IMCEC)*, Oct. 2016, pp. 1617–1620.

[21] X. Li, Z. Huang, F. Wang, X. Wang, and T. Liu, "Toward convolutional neural networks on pulse repetition interval modulation recognition," *IEEE Commun. Lett.*, vol. 22, no. 11, pp. 2286–2289, Nov. 2018.

[22] E. Norgren, "Pulse repetition interval modulation classification using machine learning," Ph.D. dissertation, KTH Royal Inst. Technol., Stockholm, Sweden, 2019.

[23] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," 2015, *arXiv:1506.00019*. [Online]. Available: http://arxiv.org/abs/1506.00019

[24] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2048–2057.

[25] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[26] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," in *Proc. Int. Conf. Artif. Neural Netw. ICANN*, Edinburgh, Scotland, vol. 2, 1999, pp. 850–855.

[27] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," 2014, *arXiv:1406.1078*. [Online]. Available: http://arxiv.org/abs/1406.1078

[28] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *CoRR*, vol. abs/1409.0473, 2014.

[29] A. M. Rush, S. Chopra, and J. Weston, "A neural attention model for abstractive sentence summarization," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Lisbon, Portugal, Sep. 2015, pp. 379–389.

[30] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2016, pp. 1480–1489.

[31] C. dos Santos, M. Tan, B. Xiang, and B. Zhou, "Attentive pooling networks," 2016, *arXiv:1602.03609*. [Online]. Available: http://arxiv.org/abs/1602.03609

[32] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3104–3112.

[33] B. Zhang, D. Xiong, and J. Su, "A GRU-gated attention model for neural machine translation," 2017, *arXiv:1704.08430*. [Online]. Available: https://arxiv.org/abs/1704.08430

[34] Z.-M. Liu and P. S. Yu, "Classification, denoising, and deinterleaving of pulse streams with recurrent neural networks," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 55, no. 4, pp. 1624–1639, Aug. 2019.

[35] O. Levy and Y. Goldberg, "Neural word embedding as implicit matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2177–2185.

[36] D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, and B. Qin, "Learning sentiment-specific word embedding for Twitter sentiment classification," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2014, pp. 1555–1565.

[37] Y. Goldberg and O. Levy, "word2vec explained: Deriving Mikolov et al.'s negative-sampling word-embedding method," 2014, *arXiv:1402.3722*. [Online]. Available: https://arxiv.org/abs/1402.3722

[38] M. Bagheri and M. H. Sedaaghi, "A new method for detecting jittered PRI in histogram-based methods," *Turkish J. Electr. Eng. Comput. Sci.*, vol. 26, no. 3, pp. 1214–1224, 2018.

[39] R. G. Licursi de Mello and F. Rangel de Sousa, "Precise techniques to detect superimposed radar pulses on ESM systems," *IET Radar, Sonar Navigat.*, vol. 12, no. 7, pp. 735–741, Jul. 2018.

[40] A. Byers and J. T. Serences, "Enhanced attentional gain as a mechanism for generalized perceptual learning in human visual cortex," *J. Neurophysiol.*, vol. 112, no. 5, pp. 1217–1227, Sep. 2014.

[41] F. Ma, R. Chitta, J. Zhou, Q. You, T. Sun, and J. Gao, "Dipole: Diagnosis prediction in healthcare via attention-based bidirectional recurrent neural networks," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2017, pp. 1903–1911.

[42] S. Seo, J. Huang, H. Yang, and Y. Liu, "Interpretable convolutional neural networks with dual local and global attention for review rating prediction," in *Proc. 11th ACM Conf. Recommender Syst. (RecSys)*, 2017, pp. 297–305.

[43] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[44] B. A. Pearlmutter, "Gradient calculations for dynamic recurrent neural networks: A survey," *IEEE Trans. Neural Netw.*, vol. 6, no. 5, pp. 1212–1228, Sep. 1995.

[45] J. Martens and I. Sutskever, "Learning recurrent neural networks with hessian-free optimization," in *Proc. 28th Int. Conf. Int. Conf. Mach. Learn.*, 2011, pp. 1033–1040.

[46] N. Ketkar, *Deep Learning With Python*. New York, NY, USA: Apress, Apr. 2017.

[47] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," 2016, *arXiv:1603.04467*. [Online]. Available: http://arxiv.org/abs/1603.04467

[48] W. Yu, W. Su, H. Gu, J. Yang, and X. Lu, "Weak maneuvering target detection in random pulse repetition interval radar," *Signal Process.*, vol. 171, Jun. 2020, Art. no. 107520.

ZHANGMENG LIU received the Ph.D. degree in statistical signal processing from the National University of Defense Technology (NUDT), China, in 2012. He is currently an Associate Professor with NUDT working on the interdiscipline of electronics engineering and computer science, especially electronic data mining.

XUEQIONG LI received the B.S. degree from the College of Transportation, Southeast University, Nanjing, China, in 2013, and the M.S. degree from the College of Electronic Science and Engineering, National University of Defense Technology, Changsha, China, in 2015, where she is currently pursuing the Ph.D. degree. Her current research interests include signal processing and machine learning.

ZHITAO HUANG received the B.S. and Ph.D. degrees in information and communication engineering from the College of Electronic Science and Engineering, National University of Defense Technology, Changsha, China, in 1998 and 2003, respectively. He is currently a Professor with the College of Electronic Science and Engineering, National University of Defense Technology. His research interests include radar and communication signal processing and array signal processing.

• • •