Robust Modulation Classification Under Uncertain Noise Condition Using Recurrent Neural Network

Shisheng Hu†, Yiyang Pei*, Paul Pu Liang‡, and Ying-Chang Liang†, Fellow, IEEE
† University of Electronic Science and Technology of China (UESTC), Chengdu, P. R. China
‡ Singapore Institute of Technology, 10 Dover Drive, Singapore 138683
Email: hss_ss@163.com, Yiyang.Pei@singaporetech.edu.sg, pliang@cs.cmu.edu, liangyc@ieee.org

Abstract—Modulation classification using deep neural networks has recently received increasing attention due to its capability in learning rich features of data. In this paper, we propose a low-complexity blind data-driven modulation classifier. Our classifier operates robustly over Rayleigh fading channels under uncertain noise conditions modeled using a mixture of three types of noise, namely, white Gaussian noise, white non-Gaussian noise and correlated non-Gaussian noise. The proposed classifier consists of several layers of recurrent neural networks (RNN) which is well-suited for learning representations from time-correlated data. The classifier is trained using the labeled raw signal samples generated under different noise conditions. Simulation results show that the performance of our proposed classifier approaches that of maximum likelihood classifiers with perfect channel knowledge and outperforms existing expectation maximum (EM) and expectation conditional maximum (ECM) classifiers which iteratively estimate channel and noise parameters.

I. INTRODUCTION

Blind modulation classification (BMC) is a promising technique to identify the modulation scheme used in a communication system with limited or no prior knowledge. It is a fundamental step before signal detection in cognitive radio networks where a cognitive receiver may not have the knowledge of the operating parameters of the primary system [1]–[3].

Conventionally, there are generally two types of BMC methods, namely, likelihood-based (LB) approach and feature-based (FB) approach. LB classifiers [4] compute and compare the likelihood functions of the received signals with different modulation scheme hypotheses which are known as the optimal classifiers in Bayes’ sense, when perfect knowledge of the channel and noise parameters are available. However, in practice, the relevant parameters have to be estimated, thus high computational complexity may be introduced, and furthermore, their performance may be limited by the accuracy of the estimated parameters. FB classifiers focus on extracting certain features of the received signals such as higher-order statistics (HOS) and use various approaches for decision making [5], [6]. They are generally with lower computational complexity but also lead to suboptimal performances.

The above conventional classifiers face a number of challenges, especially when extended to a new channel model. First, they usually assume well-defined signal and noise, the models and hence need to be re-designed each time when there is a change in the model since mismatch in the model may cause severe degradation to the resulting performance. The redevelopment of a new classifier catered for a new signal model is likely to take longer time. Secondly, signal and channel models capturing more practical concerns usually introduce more unknown parameters and this leads to classifiers with increasing computational complexity. Last but not least, tractable analytical models may not exist, e.g., for underwater communication [7] and molecular communication [8], which limits the application of the above classifiers in such cases.

Due to the recent advancement in machine learning, various data-driven approaches have been proposed [9]–[13]. These classifiers are trained with labeled signal features or samples using various machine learning algorithms. They do not necessarily need an analytical signal model and hence they can be extended easily to a new scenario as long as data samples can be collected. In addition, once they are trained offline, which may take some time, they can be implemented with a lower complexity in real time. Among the data-driven classifiers, deep neural networks based classifiers have attracted considerable attention recently due to its capability of learning complex relationship without the explicit need for feature extraction. For example, in [11]–[13], the authors have used raw representation of signal samples to train classifiers based on convolutional neural networks, recurrent neural network (RNN) and k-sparse autoencoder, respectively.

Existing data-driven classifiers have been designed assuming white Gaussian noise [11]–[13]. However, various studies have shown that the noise experienced in most communication channels is non-Gaussian with various natural and man-made sources of noise [14]. Furthermore, the additive noise has also been found to be correlated in time, which may arise due to the channel or may be introduced by narrowband filtering of uncorrelated noise [15]. Hence, the existence of different types of noise in communication channels can lead to the uncertainty of channel noise conditions for modulation classification.

In this paper, we propose a low-complexity blind data-driven modulation classifier which operates robustly under uncertain noise condition modeled using a mixture of three types of noise, namely, white Gaussian noise, white non-Gaussian noise and correlated non-Gaussian noise. The proposed classifier consists of several layers of RNN with long short-term
memory (LSTM) units which is well-suited in learning representations from time-correlated data, in this case, the signal samples modulated by the same modulation scheme across all time steps. We train the proposed classifier using labeled received in-phase and quadrature (IQ) components of signal samples generated under different noise conditions. Simulation results show that the performance of our proposed classifier approaches that of LB classifiers with perfect channel knowledge and outperforms those expectation maximization (EM) and expectation conditional maximization (ECM) classifiers which iteratively estimate channel and noise [14]–[16].

The rest of this paper is organized as follows. Section II describes the signal model and problem formulation. In Section III, the recurrent neural network and its variant LSTM neural network are briefly introduced. The structure of our proposed deep neural network is described in Section IV. Simulation results in terms of classification accuracy vs. SNR are shown in Section V. Finally, concluding remarks are given in Section VI.

II. SYSTEM MODEL AND PROBLEM FORMULATION

A. System Model

By assuming signals are sampled synchronously from rectangular pulse which satisfies the Nyquist criterion, the $n$-th received complex signal sample can be expressed as

$$ r(n) = hs(n) + w(n), $$

for $n = 1, 2, ..., N$, where $s(n)$ is the transmitted symbol drawn randomly and equiprobably from an unknown modulation scheme $M_i$, $h$ is the channel coefficient remaining constant in each observation period but varying randomly from one period to another, and $w(n)$ denotes the additive noise.

Existing data-driven modulation classifiers mostly focus on white Gaussian noise. However, noise is shown to be non-Gaussian and even time-correlated in realistic communication channels, and studies of modulation classification under those noise conditions using traditional methods are investigated [14], [15]. Thus, in this paper, in addition to white Gaussian noise, we also consider two types of non-Gaussian noise, namely, white non-Gaussian noise and time-correlated non-Gaussian noise.

1) White Non-Gaussian Noise: The probability density function (pdf) of the white non-Gaussian noise $w(n)$ is assumed to follow a Gaussian Mixture Model with $M$ components [17]:

$$ f(w(n)) = \sum_{m=0}^{M-1} \frac{\lambda_m}{\pi \sigma_m^2} \exp \left( -\frac{|w(n)|^2}{\sigma_m^2} \right), $$

where $\lambda_m$ is the probability that $w(n)$ is generated from the $m$-th Gaussian component, $0 < \lambda_m < 1$ and $\sum_{m=0}^{M-1} \lambda_m = 1$; and $\sigma_m^2$ denotes the variance of the $m$-th Gaussian component.

2) Time-correlated Non-Gaussian Noise: Time-correlated non-Gaussian noise can be obtained by passing a sequence of independent and identically distributed non-Gaussian white noise $\{e(n)\}_{n=1}^{N}$ generated from (2) through an autoregressive AR($P$) filter with coefficients $\{a(i)\}_{i=1}^{P}$ [15]

$$ w(n) = -\sum_{i=1}^{P} a(i)w(n-i) + e(n). $$

B. Modulation Classification

Denote the set of modulation schemes as $\mathcal{M} = \{M_i, i = 1, 2, \ldots, L\}$, where $L$ represents the total number of candidate modulation schemes considered. In this paper, we are interested in determining the modulation scheme $M_i \in \mathcal{M}$ of the received signals based on a sequence of $N$ received samples $r = \{r(1), r(2), \ldots, r(N)\}$.

By letting $P(M_i|r)$ be the a posterior probability of $M_i$ given the received signal sequence $r$, the modulation classification problem can be formulated as a posterior probability learning problem and solved by using the maximum a posterior (MAP) criterion

$$ \hat{M}_i = \arg \max_{M_i \in \mathcal{M}} P(M_i|r). $$

By using Bayes’ theorem, the a posterior probability can be represented as:

$$ P(M_i|r) = \frac{P(r|M_i)P(M_i)}{P(r)}, $$

where $P(r|M_i)$ is the likelihood of the received sequence $r$ given the modulation scheme $M_i$; $P(M_i)$ is the prior probability of the modulation scheme $M_i$; and $P(r)$ is the marginal likelihood of the received sequence $r$ which does not depend on $M_i$. Assuming all candidate modulation schemes are equiprobable, the MAP classifier is then reduced to the maximum likelihood (ML) classifier

$$ \hat{M}_i = \arg \max_{M_i \in \mathcal{M}} P(r|M_i), $$

which is commonly adopted in literature for modulation classification as the optimal classifier. However, as mentioned in Section I, the calculation of the above likelihood requires the complete knowledge of channel parameters and noise variance, which is usually not known at the receiver. Hence, the performance given by the ML classifier can only be used as the upper bound to the actual performance of modulation classification. EM based algorithms have been proposed to iteratively estimate channel parameters and noise variance [14]–[16]. However, such algorithms are of high complexity for real-time classification. In addition, it is very sensitive to the selection of initial values for iteration and is easily stuck in a local optimum.

C. Modulation Classification using a Deep Neural Network

In this paper, we use a deep neural network to learn the a posterior probability distribution $P(M_i|r)$, for $i = 1, \ldots, L$. The main structure of our deep neural network is shown in
To measure the performance of the a posteriori probability vector generated from the modulation scheme, the mapping function $f$ is used to map the signal samples $r$ to $d$, and $d$ is further sent into a softmax layer to conclude the a posteriori probability vector represented by $y$.

Fig. 1. First, a multi-layer neural network is used to map the $N$-dimensional received signal sequence $r$ to an $L$-dimensional vector $d$ as follows:

$$f : r \in \mathbb{C}^N \rightarrow d \in \mathbb{R}^L. \quad (7)$$

where the mapping function $f$ represents the type and structure of the neural network, which is characterized by a set of parameters $\theta$. Then, the vector $d$ is passed to the softmax activation function and produces an $L$-dimensional output vector $y$:

$$y = \frac{\exp(d)}{\sum_{j=1}^K \exp(d_j)}. \quad (8)$$

The $i$-th element $y_i$ of the vector $y$ can be considered as an estimation of the a posteriori probability associated with modulation scheme $M_i$, i.e., $y_i \approx P(M_i|r)$.

The true modulation scheme $z$ that $r$ is generated from is encoded as a one-hot vector $1_i \in \mathbb{R}^L$ i.e., an $L$-dimensional vector with the $i$-th element $z_i$ is equal to one if $r$ is generated from the modulation scheme $M_i$ and zero otherwise. To measure the performance of the a posteriori probability learning algorithm, we use the following cross-entropy loss function:

$$E(f) = -\sum_{i=1}^L z_i \log (y_i). \quad (9)$$

Our objective is to determine a suitable choice of the mapping function $f$ and the associated parameters $\theta$ to minimize the above loss function. One popular approach to find the set of parameters is to use the gradient descent algorithm to iteratively update them so that the loss function is minimized.

III. RECURRENT NEURAL NETWORK

In this section, we will introduce conventional recurrent neural network (RNN) [18] and the gated RNN with long short-term memory (LSTM) units, the latter of which are commonly known as LSTM neural network [19] and will be used in our proposed neural network.

A. Conventional Recurrent Neural Network (RNN)

Unlike fully-connected neural network where units of one hidden layer are connected to the next hidden layer, units of one hidden layer in RNN are recurrently connected to itself [20], as shown in Fig.2. Consider an RNN with $T$ input time steps $\{x_t\}_{t=1}^T$ and $T$ output time steps $\{y_t\}_{t=1}^T$. The distinctive feature of an RNN is that the output of current time step $y_t$ is decided by both the input of the current time step $x_t$ and the memory of the past time steps, described by $h_t$. This property enables RNNs to be conditioned on representations learnt from inputs of previous time steps and to dynamically change the output of the current time step. By sharing weights across time steps, RNNs have much fewer parameters and can be optimized more efficiently. Thus, RNNs are well-suited to process time series data where learning temporal dependencies are crucial. RNNs have been extensively used for language sequence modeling [21], [22], and multimodal time series modeling [23], [24].

Mathematically, the memory $h_t$ of the current time step captures the previous information up to and including time step $t$. $h_t$ is updated based on both the input of the current time step $x_t$ and the memory of past time steps denoted as $h_{t-1}$ as follows

$$h_t = \tanh(W x_t + U h_{t-1} + b_h), \quad (10)$$

where $W$, $U$ represent the weight matrices from input to memory and from memory to memory, respectively; $b_h$ is the bias and $\tanh(\cdot)$ is a nonlinear activation function used to enable these neural network to universally approximate arbitrary functions. Then the output of the current time step can be updated as follows

$$y_t = V h_t + b_y, \quad (11)$$

where $V$ is the weight matrix from memory to output and $b_y$ is the bias. Note that the parameters $W, U, V, b_h, b_y$ are shared parameters across all time steps.
Fig. 3: Structure of an LSTM unit. $\sigma$ is the sigmoid activation function, $tanh$ is the hyperbolic tangent activation function. $\oplus$ denotes element-wise addition, $\otimes$ denotes element-wise multiplication. $i_t$, $f_t$, $o_t$ are the input, forget and output gates respectively; $W_i, U_i, W_f, U_f, W_o, U_o$ and the weight matrices for transforming between input, memory and output space respectively.

B. LSTM Neural Network

Conventional RNN suffers from vanishing gradient problem during the gradient decent updates. Hence, RNN is not suitable to learn the long-term dependencies. Gated RNN has been introduced to overcome such limitation.

LSTM neural network is one kind of gated RNN with LSTM units. Its structure is shown in Fig.3, and the core idea behind LSTM unit is that besides the outer unit recurrence, it has LSTM cells that have an internal recurrence (self-loop). Each cell has the same inputs and outputs as an ordinary recurrent neural network, but introduces more parameters and a system of several gates that controls the flow of information. Three gates in an LSTM unit are defined as:

$$
i_t = \sigma(W_i h_{t-1} + U_i x_t + b_i),$$

$$f_t = \sigma(W_f h_{t-1} + U_f x_t + b_f),$$

$$o_t = \sigma(W_o h_{t-1} + U_o x_t + b_o),$$

where $\sigma(\cdot)$ is the sigmoid activation function to obtain a value between 0 and 1; $i_t$, $f_t$, $o_t$, are the input gate, the forget gate, the output gate, respectively; $W_i, U_i, W_f, U_f, W_o, U_o$ and $b_i, b_f, b_o$ are the corresponding weight matrices and bias, respectively. The above gates allow LSTM cells to learn to store and access information over long distance of units, thereby avoiding the vanishing gradient problem.

In the self-loop, the cell memory $c_t$ is put towards the next unit with self-loop weight $f_t$ to decide to remove or add information to the next cell memory:

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tanh(W_c h_{t-1} + U_c x_t + b_c).$$

The output of current time step $h_t$ is decided by the current cell memory $c_t$ and the output gate $o_t$:

$$h_t = o_t \cdot \tanh(c_t).$$

C. Temporal Attention Layer

Different strategies have been used to summarize the final output of an LSTM layer for different tasks. The most common method involves using the output at the final time step $h_T$ as a summary of all temporal information. In this paper, we go a step further and use a temporal attention mechanism [25] over the outputs of all time steps $h_1, ..., h_T$.

Attention weights $\{\alpha_t\}_{t=1}^T$ are learnt by passing the outputs $\{h_t\}_{t=1}^T$ through a shared time-distributed neural network layer with a softmax activation. This allows us to interpret the attention weights $\{\alpha_t\}_{t=1}^T$ as a probability distribution over time steps that indicate the importance of each time step toward modulation classification. The final summarized output is a weighted sum of outputs of all time steps using these attention weights:

$$h = \sum_{t=1}^{T} \alpha_t \cdot h_t. \quad (15)$$

Motivated by the superior performance of RNN and its variant LSTM neural network in many tasks, we propose to use a deep LSTM neural network to learn the a posterior distribution for the considered modulation classification problem.

IV. PROPOSED DEEP RECURRENT NEURAL NETWORK

We propose a seven-layer neural network for the modulation classification problem with the structure shown in Fig 4. This proposed deep neural network consists of three stacked-LSTM layers which are followed by four fully-connected (FC) layers. For the first two layers of the stacked LSTM, the output of the previous layer is treated as the input to the current layer. This allows temporal representations to be extracted in multiple stages in a hierarchical manner. The third and final LSTM layer utilizes a temporal attention mechanism (15) over outputs of all time steps to derive the final output.

Approaching the modulation classification problem using temporal models such as LSTM neural network is beneficial for three reasons. Firstly, as analyzed in Section III, LSTM network is capable of learning the temporal dependencies from time series data. Especially, consider solving the modulation classification problem when signal samples are highly correlated in the time domain over time-correlated non-Gaussian channels. LSTM neural network can learn more effectively these underlying temporal representations as compared to non-temporal models. The latter neural network layers can then learn the a posterior probability distribution (5) from the summarized temporal information. In addition, since the signal samples cross all time steps are modulated by the same modulation scheme, LSTM network is also capable of classifying signal samples generated over channels without time correlation.

Secondly, for previous classifiers that are based on traditional neural network, each input node can only receive a one-dimensional input. As a result, IQ components of one signal sample are concatenated and sent to two nodes of input layer [13]. However, based on the fact that in traditional LB and
Signal Samples

![Diagram of the proposed deep neural network]

**Fig. 4: Structure of the proposed deep neural network consisting of three stacked-LSTM.** The final LSTM layer uses a temporal attention layer over outputs of all time steps to summarize temporal information. Four fully-connected (FC) layers are then used to learn the \( a \text{ posteriori} \) probability.

We hypothesize that combining IQ components of one signal sample to one unit improves the isolation of different signal samples in the input layer. This will give the neural network better chances to learn the relationships across all input signal samples. Using an LSTM network allows us to combine IQ components of signal samples into two-dimensional vectors as the input to an LSTM cell at each time step. This represents a more natural method of dealing with complex numbers in time series data.

Thirdly, LSTM neural network uses significantly fewer parameters as compared to conventional fully-connected neural networks. This is because LSTM network shares weight matrices and biases across time steps so the number of parameters is not affected by the number of time steps. In contrast, conventional neural network effectively has different weights matrices per time step and the number of parameters grows linearly with the number of time steps.

After three stacked-LSTM layers, four FC layers are then used to improve the generalization power of the proposed neural network. To derive the \( a \text{ posteriori} \) probability of the modulation schemes, the softmax activation function as in (8) is used after the last FC layer.

The hyperparameters of this proposed neural network are optimized and presented in Table I. As for the training data, the IQ components of the signal samples are chosen as the features to construct the inputs to the proposed deep neural network. The true modulation schemes of the input signal samples represented using one-hot vectors are used as the labels for the inputs. Moreover, cross-entropy (9) is chosen as the loss function. The Adaptive Moment Estimation (Adam) optimizer is used to update the neural network parameters [26]. After training, the classification decision is made following the MAP criterion given in (4).

### Table I: Parameters Used in Proposed Deep Neural Network

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of LSTM layers</td>
<td>3</td>
</tr>
<tr>
<td>Number of time steps of each LSTM layer</td>
<td>128</td>
</tr>
<tr>
<td>Dimension of input to each time step of the first LSTM layer</td>
<td>2</td>
</tr>
<tr>
<td>Dimension of output of each time step of the LSTM layers</td>
<td>{10, 10, 10}</td>
</tr>
<tr>
<td>Dimension of the output of the FC layers</td>
<td>{16, 16, 16, 4}</td>
</tr>
<tr>
<td>Activation function of the first three fully-connected layers</td>
<td>ReLU</td>
</tr>
<tr>
<td>Batch size</td>
<td>128</td>
</tr>
<tr>
<td>Learning rate</td>
<td>0.0005</td>
</tr>
<tr>
<td>Training epoch</td>
<td>450</td>
</tr>
</tbody>
</table>

**V. Simulation Results**

In this section, the performance of the proposed deep neural network is evaluated using simulations. The proposed deep neural network is trained and tested using the deep neural network library Keras. The modulation candidates considered in the paper are \( \mathcal{M} = \{\text{BPSK}, \text{QPSK}, 8\text{-PSK}, 16\text{-QAM}\} \). It is assumed that the receiver determines the modulation scheme of the received signals using \( N = 128 \) samples. We consider Rayleigh fading channels with the magnitude of the channel coefficient \( h \) assumed to be Rayleigh distributed, with \( E[|h|^2] = 1 \), and its phase is uniformly distributed in \( [0, 2\pi] \). We consider three types of noise, namely, white Gaussian noise, white non-Gaussian noise, and time-correlated non-Gaussian noise.

For each type of noise, 6000 realizations of received signals are generated according to the signal model defined in Section II for each modulation scheme and SNR value. Further, IQ components of the signal samples generated with all types of noise and their true modulation schemes as labels are mixed to construct the entire training data. After training, the classification accuracy of the proposed classifier is evaluated using the test data generated in a way same as the training data.

**A. White Gaussian Noise**

We first compare the classification accuracy of our proposed classifier with some existing classifiers under white Gaussian noise \( w(\cdot) \sim \mathcal{CN}(0, \sigma_0^2) \) at different SNR values in Fig. 5. The SNR of white Gaussian noise is defined as \( E[|h|^2]/\sigma_0^2 \) assuming the transmitted symbols have unit power. The performance of ML classifier shows the upper bound for all classifiers since it assumes perfect knowledge of the channel and noise variance. It is observed that the performance of our proposed classifier converges to the ML classifier when...
SNR is greater than 10dB, which can be seen as a result of good capability of learning the \textit{a posteriori} probabilities as analyzed in section II. It is further observed that our proposed classifier outperforms the classifier using EM algorithm [16], which iteratively estimates the channel and the noise variance. Although the performance of EM is very close to our proposed classifier especially at median SNR, the EM classifier incurs much higher computational complexity since the number of iterations needed can be very large.

\subsection*{B. White Non-Gaussian Noise}

For the case of Rayleigh fading white non-Gaussian channels, the noise is generated from a two-term Gaussian mixture model, which captures both thermal noise (with proportion \( \lambda_0 \) and variance \( \sigma_0^2 \)) and the impulse noise (with proportion \( \lambda_1 \) and variance \( \sigma_1^2 \)). The values of \( \lambda_0 \) and \( \sigma_1^2/\sigma_0^2 \) are set to 0.9 and 100, respectively. As in [14], we define SNR as the ratio of the averaged received signal power and the variance of the thermal noise component, i.e., \( E[|h|^2]/\sigma_0^2 \).

In Fig. 6, we compare the classification accuracy of our proposed classifier with some existing classifiers under different SNR values. Same as the case for Gaussian noise, the ML classifier that assumes perfect knowledge of channel coefficient, noise power and noise proportion is observed to have the best performance. In addition, it is observed that our proposed classifier outperforms the EM-based classifier in [14] (using the same number of received samples \( N = 128 \)) with a larger performance gap as compared to the case of white Gaussian noise. This is due to the fact that more parameters need to be estimated for EM-based classifier for the case of non-Gaussian noise and the limited number of \( N = 128 \) received signals samples may not be enough to obtain a good estimation. Even though the classification accuracy of EM-based classifiers can be improved by increasing the number of received signal samples (e.g., \( N = 256 \)), this will cause even longer time for modulation classification and increased computational complexity for iterative estimation. Furthermore, we also compare with the ML classifier which assumes Gaussian noise, i.e., it regards the noise variance as \( \lambda_0\sigma_0^2 + \lambda_1\sigma_1^2 \) and the parameters \( h, \{\lambda_0, \lambda_1\} \) and \( \{\sigma_0^2, \sigma_1^2\} \) are also assumed to be known [14]. Such a classifier suffers a poor performance due to the mismatch of the noise model.

\subsection*{C. Time-Correlated Non-Gaussian Noise}

In Rayleigh fading time-correlated non-Gaussian channels, the noise is generated from AR(1) process with \( a[1] = -0.75 \) and \( e[n] \) as a two-term Gaussian mixture model with \( \lambda_0 = 0.9, \sigma_1^2/\sigma_0^2 = 100 \).
In Fig.7, we compare the performance of the proposed classifiers under time-correlated non-Gaussian noise with the ML classifier assuming perfect knowledge of all unknown parameters [15], the ECM classifier which is an EM-based algorithm that iteratively estimates the unknown parameters [15], and the ML classifier assuming white non-Gaussian noise. It is observed that the performance of our proposed classifier is closest to the ideal ML classifier. It is further observed that the performance of our proposed classifier is comparable to the case of white non-Gaussian noise whereas the ECM suffers from larger performance degradation. This is because the performance of ECM algorithms is limited by the increasing number of unknown parameters to be estimated in this case. It’s worth noting that at high SNR regimes, the ECM classifier is even worse than the ML classifier assuming a mismatched noise model. This is because at high SNR values, the effect of noise becomes very small. For ML classifier assuming white non-Gaussian noise, this results in an insignificant mismatch effect. However, for ECM classifier, the estimation of a smaller noise value becomes very difficult and may easily get stuck in a local optimum. Overall, the performance of our proposed classifier is more robust and the complexity is not affected with a change of different noise models.

VI. CONCLUSION

In this paper, we have proposed a low-complexity blind data-driven modulation classifier which operates robustly over Rayleigh fading channels under uncertain noise condition modeled using a mixture of three types of noise, namely, white Gaussian noise, white non-Gaussian noise and correlated non-Gaussian noise. The proposed classifier has used several layers of recurrent neural network (RNN) which is well-suited for learning representations from time-correlated data. The classifier has been trained using the labeled raw signal samples generated under different noise conditions. Simulation results have shown that the performance of our proposed classifier approaches that of maximum likelihood classifiers with perfect channel knowledge and outperforms existing expectation maximum (EM) and expectation conditional maximum (ECM) classifiers which iteratively estimate channel and noise.

ACKNOWLEDGEMENT

This work was supported by the National Natural Science Foundation of China under Grants 61571100, 61631005, and 61628103.

REFERENCES


