DEEP LEARNING BASED SIGNAL DETECTOR

PROJECT

Integrated M. TECH

SUBMITTED BY

PRASOON RAGHUWANSHI (15MI440)
ANSHUL DHIMAN (15MI439)

To

Dr. Philemon Daniel

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY
HAMIRPUR-177005, HP (INDIA)

NOVEMBER 2019

OBJECTIVE:

This assignment presents the deep learning (DL) in the signal detection of orthogonal frequency division multiplexing (OFDM) systems. Particularly, a novel DL-based detector termed as DeepIM, which employs a deep neural network with fully connected layers to recover data bits in an OFDM system. To enhance the performance of DeepIM, the received signal and channel vectors are pre-processed. Using dataset collected by simulations, DeepIM is first trained offline to minimize the bit error rate (BER) and then the trained model is deployed for test signal detection of OFDM. Simulation results show that DeepIM can achieve a near-optimal BER with a lower runtime than existing hand-crafted detectors.

METHODOLOGY:

Consider an OFDM system with Nc sub-carriers that are split into G groups of N sub-carriers, i.e., Nc = NG. At the transmitter, the signal processing of every OFDM group is the same and independent of each other. Thus, we address only one group for simplicity. Particularly, in every transmission of each group, only K out of N sub-carriers are activated to send a total of p data bits that include p1 = K log2 M bits carried by K complex data symbols and p2 = log2(C(N,K)) bits carried by indices of active sub-carriers, i.e., p = p1+p2. Here, M is the M-ary modulation size. Notice that the mapping from p1 bits to a combination of K active indices can be implemented using either combinatorial methods or a look-up table. As a result, based on p incoming bits, the transmitted vector x = [x1,...,xN] is formed by assigning K non-zero data symbols to corresponding K active sub-carriers, i.e., xi is non-zero if sub-carrier i is active and xi = 0 otherwise, i = 1,...,N.

At the receiver, the received signal in the frequency domain is expressed by

$$y = hx + n$$

h = [h1,...,hn] is assumed to be the Rayleigh fading channels. n represents the additive white Gaussian noise (AWGN).

PROPOSED DL-BASED DETECTOR:

1. **Structure of DeepIM**

   The structure of the proposed DeepIM is depicted in Fig. 1. Similar to the current detection schemes of OFDM, the channel information is assumed to be known at the receiver. Thus, the received signal **y** and the channel **h** are considered as the coarse inputs of the DNN. To achieve a better detection performance, **y** and **h** will be pre-processed before entering the DNN model. In particular, firstly, the well-known zero-forcing (ZF) equalizer is employed to get an equalized received signal vector as follows $\bar{\mathbf{y}} = \mathbf{y} \odot \mathbf{h}^{-1}$. Intuitively, this is expected to improve the reconstruction of the *M*-ary symbols at the active sub-carriers. Secondly, the energy of the received signal is computed and then combined with $\bar{\mathbf{y}}$ to create the input of the DNN.

   As shown in Fig. 1, the real and imaginary parts of $\bar{\mathbf{y}}$ and the received energy vector $\mathbf{y}_m$ are concatenated to form the 3*N*-dimensional input vector **z.**

   The proposed DNN structure consists of two fullyconnected (FC) layers including one hidden FC layer of *Q* nodes and one FC output layer of *p* nodes as illustrated in Fig. 1. At the hidden layer, either the rectifier linear unit (Relu), $f_{\text{Relu}}(x) = \max(0,x)$, or the hyperbolic tangent (Tanh).

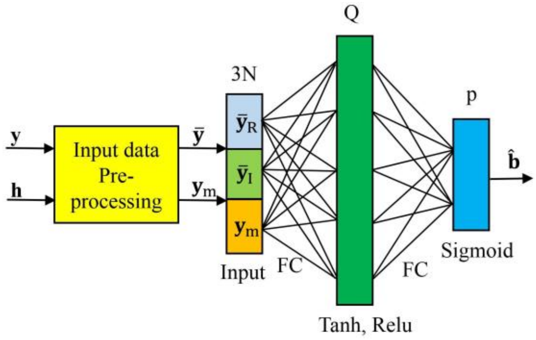

Fig. 1. The structure of the proposed DeepIM detector for OFDM.

The Sigmoid function is applied at the output layer to output the estimate of the transmitted data bits $\hat{\mathbf{b}}$. It is worth noting that the proposed DNN needs only two nonlinear layers to be sufficient for the highly accurate detection of both *M*-ary and index bits of OFDM. Specifically, let us denote by $\mathbf{W}_1$, $\mathbf{b}_1$ and $\mathbf{W}_2$, $\mathbf{b}_2$ the weights and the biases of the first and second FC layers, respectively. The output vector of the DeepIM model can be expressed as

$$\hat{\mathbf{b}} = f_{sig}\left(\mathbf{W}_2 f_{tanh/Relu}\left(\mathbf{W}_1\mathbf{z} + \mathbf{b}_1\right) + \mathbf{b}_2\right)$$

We now highlight some key insights of the proposed DNN structure. First, the input and output lengths of DeepIM are determined by system parameters such as N, K and M, while the length of the hidden layer, i.e., Q, needs to be properly selected to achieve a desired performance for each system configurations. Intuitively, when the number of transmitted bits p per OFDM group increases, we need Q large enough to guarantee a predetermined performance. Moreover, just by adjusting Q, we can attain a promising trade-off between the detection accuracy and the model complexity. Another advantage of the proposed DeepIM is that the length of the input vector z depends on N only, this interestingly makes the complexity of DeepIM less dependent on K and M. In contrast, the complexity of current detectors strongly relies on these parameters.

2. Training Procedure

Before using the proposed DeepIM detector, we need to train offline the DNN model with the data collected from simulations. Particularly, various sequences of p bits b are randomly generated to obtain a corresponding set of transmitted vectors as x. These vectors are then sent to the receiver subject to the effects of the Rayleigh fading channel and AWGN noise. Here, the channel and noise vectors are also randomly generated and changed from one-bit sequence to another, based on their known statistical models. The collected received signal and channel vectors, i.e., y and h are preprocessed as described in the previous section to attain the input dataset z whose labels are corresponding bit sequences b. Using the collected data, the DeepIM model is trained to minimize the BER, or equivalently, to minimize the difference between b and its prediction $\hat{\mathbf{b}}$. Thus, we simply adopt the mean-squared error (MSE) loss function for the training

$$\mathcal{L}\left(\mathbf{b}, \hat{\mathbf{b}}; \theta\right) = \frac{1}{p}\left\|\mathbf{b} - \hat{\mathbf{b}}\right\|^2$$

where $\theta = \{\mathbf{W}_i, \mathbf{b}_i\}_{i=1,2}$ are the weights and biases of the model. The model parameters θ can be updated for the batches randomly picked up from data samples, using the stochastic gradient descent (SGD) algorithm as follows

$$\theta^{+} := \theta - \eta \nabla \mathcal{L}\left(\mathbf{b}, \hat{\mathbf{b}}; \theta\right)$$

where η is the learning rate that defines the step size of the SGD. In our training, we adopt an advanced update algorithm based on the SGD, known as the adaptive moment estimation (Adam) optimizer, which can be easily implemented in various off-the-shelf DL platforms, such as Tensorflow and Keras.

3. **Testing Procedure**

Once trained, the DeepIM model with the optimized parameters θ is utilized for the testing the signal detection of OFDM with arbitrary received SNRs and channels of interest. More specifically, the proposed scheme can be implemented in a real-time manner to estimate the data bits over various channel fading conditions with no extra training for θ. Whenever the received signal and channel information are fed to DeepIM, it will autonomously output the estimated bits in very short computation time. Most importantly, our proposed scheme performs as good as the ML detector under channel estimation errors, though it is trained with perfect CSI, i.e., we do not need to retrain the model.

RESULTS:

In all considered experiment settings, the proposed DeepIM is trained with 10^3 epochs, each of which contains 20 batches of 10^3 data samples (also known as batch size). Hence, there are a total of $2{\times}10^4$ batches with 2×10^7 samples used for training. The learning rate η is set to 0.001, while a number of other DL parameters, such as Q is selected upon OFDM parameters in each experiment. Notice that although we consider Rayleigh fading channels in our experiments, the DeepIM detector can be applied to any other channel models.

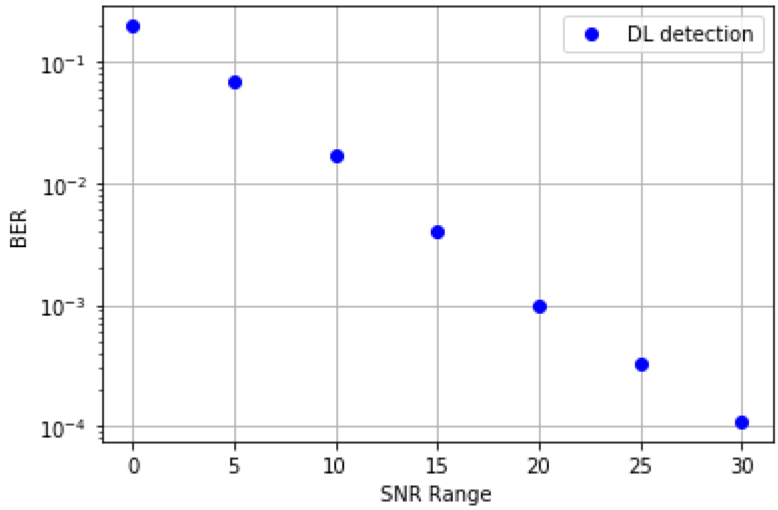

Fig. 2. BER of the proposed DeepIM under perfect CSI condition and (N,K,M) = (4,1,4).

Fig. 2 shows the BER of the proposed DeepIM, when (N,K,M) = (4,1,4) and perfect CSI is assumed, while $\bar{\gamma}_{\text{train}}$ is set to 10 dB. At the hidden layer for Q = 32, the Tanh is used. This is due to the fact that the output of the Tanh is not limited to be non-negative as the Relu, thus provides higher model capacity than the Relu. This makes the Tanh more appropriate for the DeepIM models with small Q, which may not have enough model capacity to perform the detection task.

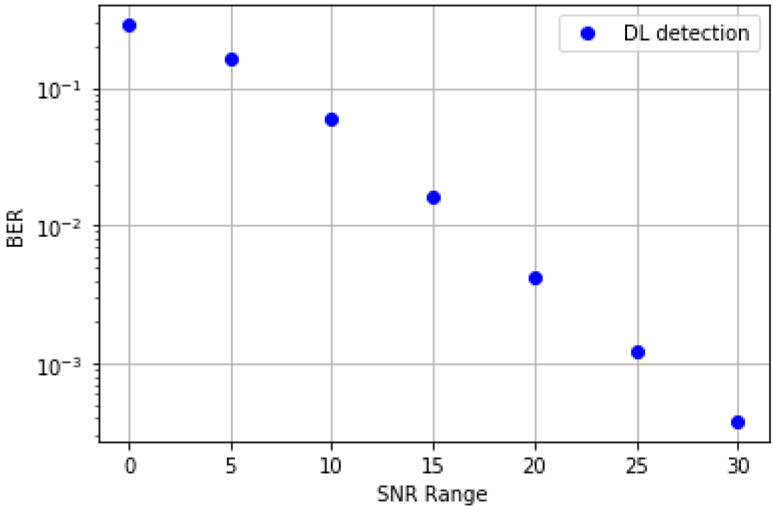

Fig. 3. BER of the proposed DeepIM under perfect CSI condition and (N,K,M) = (4,3,4).

Fig. 3 shows the BER of the proposed DeepIM, when (N,K,M) = (4,3,4) and perfect CSI is assumed, while $\bar{\gamma}_{\text{train}}$ is set to 15 dB. At the hidden layer for Q = 64. Higher data rate means larger p, or equivalently, larger number of classes involved in DeepIM. Hence, the DNN model needs to have higher model capacity than the previous setting; this observation justifies the selection of larger Q in Fig. 3 than that in Fig. 2.

As seen from two figures, DeepIM provides an attractive trade-off between performance and complexity when adjusting Q, which is not available in existing detectors.