A Deep Convolutional Learning Method for Blind Recognition of Channel Codes

Feng Tian¹, Jiao Wang² and Jianqing Li²*

¹Chengdu Aircraft Industry (Group) Co., Ltd, Chengdu, China
²School of Electronic Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China
Email: buaatf@126.com; misology@foxmail.com; lijq@uestc.edu.cn

Abstract. Blind identification of channel codes is a significant part in the field of non-cooperative signal processing. It plays a significant role in intelligent communication, information interception, information confrontation and so on. To solve issues of high dependence of manual extraction of expert features, low robustness, difficulty of deployment from traditional methods for blind recognition of channel codes, this paper presents a novel way for blind recognition of channel codes by integrating deep learning technology. For the extraction of features from signals, deep convolutional neural network containing 4 convolutional layers is designed. For model training, database of convolutional codes is generated. Experimental results show that this method can achieve 98%+ recognition accuracy with SNR not lower than 4 dB, which proves the effectiveness of the method.

1. Introduction

Channel coding is widely applied to digital communication systems, and error-correcting code is the most important encoding method of it. The blind recognition technology of channel code is reverse-identifying of coding parameters, which has great potential for applications in communication fields [1]. As modern digital communication systems develop, blind identification of channel coding has been widely used in electronic countermeasure, intelligence analysis, and adaptive modulation coding (AMC) [2]. The central idea is the receiver attempts to identify the coding parameters from the channel received data blindly. With the development of cognitive communication, blind recognition of channel coding becomes an essential function to future intelligent communication system, which has great theoretical significance and application value for the research of this technology.

Blind recognition of channel codes is to restore the original data of received coding sequence by recognizing the encoding parameters of the original information. Then decode the coding sequence via certain algorithm. At present, the blind recognition of channel codes mainly focuses on the identification of the coding parameters of convolutional codes and block codes. For example, Refs. [3, 4] use the soft information of the receiving sequence to calculate the comprehensive posterior probability (SPP) of each encoding candidate, and use SPP as the recognized feature vector to quickly identify the encoding method used by the sending end. Further, Ref. [5] proposes an optimization algorithm that using the average likelihood difference (LD) to replace log-likelihood ratio as the feature vector. The channel coding recognition algorithm above needs to calculate the recognition feature quantity of each possible coding candidate before channel decoding, and the recognition feature quantity is complex in form and contains a lot of exponential and multiplication operations. On the one hand, the computational complexity of identifying feature quantity is high and the extra delay...
introduced is large. On the other hand, the calculation of the recognition features needs a lot of extra resource consumption, which is not idea for the scenario with limited hardware resources.

In recent years, both in industrial production and scientific research, deep learning has made remarkable achievements, and has been successfully applied in image processing [6, 7], speech recognition, video processing, biomedicine and other fields [8]. Inspired by these achievements, researchers have attempted to use deep learning technology to deal with communication signal processing and successes have been obtained for modulation classification [9], signal classification, channel estimation and decoding [10]. Deep learning offers a novel way for us to enrich the field of error-correcting coding.

A deep convolutional neural network (CNN)-based method for the blind recognition of channel code parameters is presented in this paper, which reduces the complexity of feature extraction and can adapt to more complicated communication environments. First, for maximum retention of the original data of the signal, the one-dimensional signal to be recognized is used as the input data of the model in this method. Next one-dimensional convolutional neural network is built for channel coding blind recognition according to the characteristics of one-dimensional signal. This method can achieve high recognition accuracy in blind recognition of channel codes, at the same time, it can reduce the high dependence of traditional channel coding blind recognition methods on artificial feature extraction and prior knowledge, and can be directly used for multiple types of error correction code parameter recognition.

2. System Structure
In this section, we propose the fundamental system structure of transceivers of channel encoder and decoder, this process is shown in figure 1. The channel coding blind recognition discussed in this paper (the dotted box portion in figure 1) is located at the receiving end, after demodulation and before decoding. This paper focuses on studying the convolutional codes, but the proposed method can be easily applied to other types of channel codes. Set the information sequence before channel coding as \(b = [b_1, b_2, \ldots, b_i]\), and the code sequence \(c\) is obtained after encoding in code rate \(k/n\). The code sequence \(c\) is then mapped to the sign vector \(s\) by binary phase shift keying (BPSK) modulation. This work considers an additive white gaussian noise (AWGN) channel. After passing through AWGN channel, the receiving sequence at the receiving end is \(r = s + w\), where \(w\) is the noise sequence. We use the log-likelihood ratio value instead of the channel value. The output sequence of soft information after soft demodulation is \(l = [l_1, l_2, \ldots, l_n]\). The soft information \(l_i\) of the \(i\)th bit is expressed as the posterior logarithmic likelihood ratio of the coding bit \(c_i\), that is

\[
l_i = \ln\left(\frac{p(c_i = 0|r)}{p(c_i = 1|r)}\right), i = 1, 2, \ldots, n
\]

Figure 1. Structure of a general communication link.

3. The Proposed Approach
3.1. Deep Learning
The concept of deep learning is based on the research of artificial neural network, multilayer perceptron with multiple hidden layers is a class of artificial neural network. The convolutional neural
network is a kind of feedforward neural networks containing convolutional layers and deep structure. It has the advantages of local connection, weight sharing and pooling operation, which vastly simplifies the training complexity of the model. CNN has a strong feature extraction capability and does not need to carry out complex data preprocessing process, which is very suitable for one-dimensional signal recognition.

The size of the receptive field of the model is decided by the number of the convolutional layers and the size of the convolution kernel. When performing convolution operation, each feature map uses the same convolution kernel, which traverses the data at a specific step size. In general, the slide step size is set to be less than the size of the convolution kernel, and the product of traversal is added with the offset parameters, and then the sum is fed into the activation function. Finally, the feature maps of the output layer are passed to the next layer. The convolution operation of a neuron is operated as follows:

$$z = f\left(\sum w_j \times x_j + b\right)$$

where $x$ is the input sample, $w$ is the weight coefficient, $b$ is the bias vector, $f$ is the activation function, and $z$ is a value in the next feature map.

### 3.2. Network Architecture

The convolutional neural network is a kind of feedforward neural networks containing convolutional layers and deep structure. Neural networks consist of many linked neurons. Each layer of the network is composed of many neurons, and the output of each layer is the nonlinear function of the weighted sum of the lower neurons.

The process of the proposed deep neural network architecture is shown in table 1. It consists of 4 convolutional layers, 2 max pooling layers, 1 global average pooling layer, 1 dropout layer and 3 dense layers. Particularly, the first convolutional layer contains 64 convolution kernels of size 1×3. The second convolutional layer contains 128 convolution kernels with the size of 1×3. The following layer is a pooling layer with pool size of 2. Then, a convolutional network is connected behind the layer, and output 128 feature maps which become the input of the second pooling layer. The fourth convolutional layer contains 256 filters with the size of 1×3. In order to introduce nonlinear features into the system and enhance the complexity of the network, the rectified linear unit (ReLU) activation function is utilized after each convolutional layer, and the gradient vanishing phenomenon can be prevented. Two dense layers of 128 and 64 neurons were added after the fourth convolutional layer. In addition, a dropout layer is introduced after the first dense layer to randomly hide some neurons with a probability of $P = 0.4$ [11], so as to decrease the computational complexity of model training, prevent model from over-learning the features in the data, and increase the generalization capability of the model. Finally, the output layer outputs the classification label using the softmax multi-classifier, and the confidence of each coding parameter on K types is calculated as follows:

$$\hat{y}_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}, i = 1, 2, ..., K$$

### 3.3. Neural Network Training

Keras used for model training is a popular high-level application programming interface (API) built in python which makes it simpler and more efficient. The Keras framework provides complete library functions for model building, training and testing. The network layer, loss function, gradient optimizer, regularization method and so on are all individual modules, which can be used to construct the network model conveniently, and are suitable for cutting-edge research work.

To evaluate the blind recognition performance of channel coding, the error gradients of the predicted label values ($p$) and true class labels ($t$) are represented by the minimum cross entropy loss function ($L$):
Table 1. CNN network structure.

| Type of layer          | Output dimensions | Parameters |
|------------------------|-------------------|------------|
| Input layer            | $1 \times 5000$   |            |
| Convolutional layer    | $64 \times 5000$  | $1 \times 3$ |
| Convolutional layer    | $128 \times 5000$ | $1 \times 3$ |
| Max Pooling            | $128 \times 2500$ | $1 \times 2$ |
| Convolutional layer    | $128 \times 2500$ | $1 \times 3$ |
| Max Pooling            | $128 \times 1250$ | $1 \times 2$ |
| Convolutional layer    | $256 \times 1250$ | $1 \times 3$ |
| Global Average Pooling | 256               |            |
| Dense                  | 256               |            |
| Dropout                | 256               | 0.4        |
| Dense                  | 128               |            |
| Softmax                | 7                 |            |

$$L(t, p) = - \frac{1}{N} \sum_{i=1}^{N} t_i \log(p_i) + (1 - t_i) \log(1 - p_i)$$  \hspace{1cm} (4)

Adam [12] adaptive learning rate optimization algorithm was adopted for gradient descent, and its initial default learning rate was $lr = 0.002$.

4. Experimental Results

4.1. Dataset

To simulate the complex communication environment, an encoding recognition dataset is generated to train and test the recognition accuracy of the presented approach. Matlab software is used to generate 5 convolutional codes with the code rate of $1/2$, and the 5 generating matrices of the convolutional codes are $C[5, 7]$, $C[15, 17]$, $C[23, 35]$, $C[53, 75]$, $C[133, 171]$, $C[247, 371]$ and $C[561, 753]$, respectively. Each generation matrix generates 20,000 frames of signal samples of length of 5000. At the sending end, the generated random signal source is encoded by the convolutional encoder, and then the encoded data is modulated via BPSK, and then fed into the AWGN channel for transmission. The Signal-to-noise ratio (SNR) of a single sample is a random value of -10dB~20dB. At the receiving end, the received communication signal is demodulated accordingly, and the soft sequence after demodulation constitutes the sample used in this work. Then the samples are divided into training samples and verification samples according to the ratio of 8:2.

4.2. Results of Experiments

The method of deep learning is utilized for blind identification of the coded dataset, which does not require any preprocessing and manual feature extraction of the dataset. To assess the accuracy of the classifier under different SNR, a new dataset is generated as a test set. Each signal-to-noise ratio under each convolutional code parameter generates 1000 samples respectively to constitute the test set. The identification results of different convolutional code parameters are shown in figure 2.

The recognition results of the proposed approach for convolutional code parameters is shown in figure 2. As you can see from the figure that along with the increase of SNR, the recognition performance of the model is gradually improving. In particular, the identification performance of all convolutional code parameters is greatly improved when the SNR is higher than 0 dB. In addition, when the SNR gradually increases, the longer the constraint length is, the slower the accuracy rate will increase. However, when the SNR is greater than 5 dB, all the convolutional code parameters reach the recognition accuracy of 90%+. The test results show that the presented approach is effective.
In order to analyze the misidentification relationship between different parameters, we draw several representative confusion matrices respectively, with signal-to-noise ratio of -2 dB, 2 dB and 4 dB, as shown in figure 3. In the figure, the vertical coordinate is the real label of the signal, and the horizontal coordinate is the recognition result of the model. The blue result on the diagonal is the accuracy of correctly identifying the class.

As you can see from figure 3 that the convolutional code with shorter constraint length has higher recognition accuracy at low SNR. When the SNR is -2 dB, only C[5, 7] can achieve the 89% of recognition accuracy while other types are lower than 50%. When the SNR increased to 2 dB, the recognition accuracy of all types was significantly improved, and the recognition accuracy of the four convolutional codes with the shortest constraint length reaches 97%. After increasing the SNR to 4 dB, the recognition accuracy of all convolutional code parameters is higher than 98%. It can also be seen from the above analysis results that when the constraint length of convolutional code increases, the structure of channel coding becomes more complex, and higher SNR is needed to achieve the same recognition accuracy.

5. Conclusion
This work proposes a blind recognition of channel codes approach based on deep convolutional neural network. First, we construct a deep neural network including 4 convolutional layers, 2 pooling layers, and 3 dense layers to realize end-to-end recognition of coded data. Next, we use Matlab to generate a channel coded dataset for the evaluation of our algorithm. Simulation results show that this method is very suitable for blind recognition of channel coding, it also works well under low SNR circumstances.
References

[1] Huang L, Chen W, Chen E and Chen H 2017 Blind recognition of k/n rate convolutional encoders from noisy observation Journal of Systems Engineering and Electronics 28 (2) 235-243.

[2] Goldsmith A J and Chua S G 1998 Adaptive coded modulation for fading channels IEEE Transactions on Communications 46 (5) 595-602.

[3] Moosavi R and Larsson E G 2014 Fast blind recognition of channel codes IEEE Transactions on Communications 62 (5) 1393-1405.

[4] Moosavi R and Larsson E G 2011 A fast scheme for blind identification of channel codes IEEE Global Communications Conference pp 1-5.

[5] Yu P, Peng H and Li J 2016 On blind recognition of channel codes within a candidate set IEEE Communications Letters 20 (4) 736-739.

[6] Szegedy et al 2015 Going deeper with convolutions Computer Vision and Pattern Recognition 1-9.

[7] He K, Zhang X, Ren S and Sun J 2016 Deep residual learning for image recognition Computer Vision and Pattern Recognition 770-778.

[8] LeCun Y, Bengio Y and Hinton G 2015 Deep learning Nature 521 (7553) 436-444.

[9] OShea T J, Corgan J and Clancy T C 2016 Convolutional radio modulation recognition networks International Conference on Engineering Applications of Neural Networks pp 213-226.

[10] Gruber T, Cammerer S, Hoydis J and Brink S T 2017 On deep learning-based channel decoding Annual Conference on Information Sciences and Systems (CISS) pp 1-6.

[11] Srivastava N, Hinton G E, Krizhevsky A, Sutskever I and Salakhudinov R 2014 Dropout: A simple way to prevent neural networks from overfitting Journal of Machine Learning Research 15 (1) 1929-1958.

[12] Kingma D P and Ba J 2015 Adam: A method for stochastic optimization International Conference for Learning Representations 1-15.