Fast Clustering of Communication Signals

Gang Ji¹, Yuxuan Zhu¹*

¹Institute of Systems Engineering, Academy of Military Science, Beijing, 100083, China
*Corresponding author’s e-mail: zhuyx16@tsinghua.org.cn

Abstract. In the modern and digital battlefield, grasping the electromagnetic power of battlefield system is the key to deciding the outcome of the war, and accurate and comprehensive electromagnetic situational awareness is of great importance. Communication protocol analysis is a prerequisite for smart jamming in non-cooperative electronic countermeasures. The design of efficient jamming strategy for synchronous or networked signals can make up for the shortcoming of insufficient jamming power, solve the problems of remote and covert jamming, greatly reduce the cost of jamming equipment and improve the survivability of battlefield. Therefore, the rapid analysis of communication signal protocol has important military application value. Traditional protocol analysis requires parameter estimation layer by layer on the communication protocol, and bitstream analysis of the protocol word can only be carried out on the premise of solving modulation pattern identification and demodulation, interweave and scrambling parameter estimation and demodulation, channel decoding parameter estimation and decoding. This analysis method has the following disadvantages: long analysis cycle, strong expert dependence and high algorithm complexity. It is difficult to meet the demand of real-time analysis of unknown and flexible electromagnetic signals. In order to solve this problem, this paper proposes an idea of fast clustering of signal protocols in the physical layer by artificial intelligence method.

1. Introduction

A typical digital communication system consists of senders, receivers and channels. The analog signal of the sender is first processed by digital-to-analog conversion and source coding. The main function of source coding is data compression to realize the transmission of information with the least code element and improve the transmission efficiency and bandwidth utilization. The source outputs the information sequence, and then carries on the channel coding processing, in which the channel coding process includes scrambling code, error correction code and interleaving, and then enters the transmitting channel through modulation. Among them, the function of scramblers lies in the random processing of information, and the main purpose of error-correcting coding and interleaving is to reduce the random error and burst error of channel.

Before the implementation of electronic jamming, it is necessary to identify the orientation, modulation mode, channel code and other information of the jamming object first, and then carry out targeted jamming. Traditional communication interference is mainly to cover the frequency band used by the other party, which requires the jammer to continuously send high-power jamming signals. And modern communication systems in order to provide reliable communication quality will basically use channel coding to control the error code generated in the transmission process, itself also has a certain anti-jamming ability. But if we can identify and acquire the characteristics of the communication protocol of the other side, then we can target them and implement the interference more efficiently. In
this way, on the one hand, the jammer can work at a relatively low average transmitting power; on the other hand, jamming based on the weakness of enemy channel coding can improve the efficiency of jamming.

However, in order to ensure high quality data transmission, many coding methods are often used in modern digital communication. It is a systematic and complicated process to clarify the parameters of each link of coding one by one. Typical channel codes include simple one-level error correction, one-level error correction with interweave, two-level error correction cascade with interweave, and more multi-level error correction cascade. It is very difficult to calculate the channel coding parameters without any prior information, which requires massive calculation and analysis.

Deep Learning models extract abstract and invariable high-level attribute features from low-level features to achieve complex nonlinear function approximation. Compared with shallow models, Deep Learning models have stronger generalization ability and richer information describing the essence of data. In view of the excellent feature extraction and learning ability of deep learning models, some researchers have applied deep learning methods to the recognition of modulation patterns [1][2][3][4] and channel coding [5][6] in recent years.

2. Deep learning methods in modulation recognition and channel coding identification

In [1], a convolutional neural network assisted Automatic modulation recognition (AMR) method for multi-antenna systems is proposed. A convolutional neural network and a classifier are trained, and it is proved that the proposed algorithm is superior to the automatic modulation recognition method assisted by Hoc (High Order Cumulants) and ANN (Artificial Neural Networks) in the experimental arithmetic averaging strategy.

In [2], In-phase and Quadrature (IQ) samples are used to train CNN, and the dropout layer is used to replace the pooling layer, so as to achieve higher identification accuracy. The author also designed a CNN based on constellation to identify the modulation modes that were difficult to distinguish in the previous CNN, such as 16QAM and 64QAM, and to classify QAM signals in the case of low SNR.

Considering that the current deep learning-based automatic modulation recognition models are all used in specialized channels rather than in generalized channels, [3] proposed a deep learning-based blind channel identification (BCI) aided Generalized Automatic Modulation Recognition (GEN AMR) method. The method is composed of two independent CNNs. The first CNN trains IQ sampling signals by distinguishing channel categories, and the second CNN trains by LOS (Line of Sight) model and NLOS (Non-Line of Sight) model.

In [4], the front-end detector is used to obtain snapshots of raw narrow-bandwidth signals, and the Convolutional Neural Network is used to automatically extract features from these snapshots to generate decision-class estimates. The experimental results on a small synthetic RF data set show the feasibility of deep neural network in the field of communication.

In [5], a blind recognition scheme for convolutional codes and Rom codes (RS, Reed-Solomon) is proposed, which uses neural networks to identify encoders from a candidate set, and evaluates the performance of the classifier in both noiseless and noiseless conditions. In the experiments of identifying terrestrial wireless and satellite communication channels, this classifier can realize high precision recognition of encoders at low signal-to-noise ratio.

In [6], the authors propose an algorithm with robustness against channel damage (such as multipath fading) using deep learning, which can identify the channel code parameters of any coding scheme (such as LDPC, convolution, Turbo, and Polar codes) without any prior knowledge, channel state estimation, and SNR estimation.

Although the deep learning method has achieved good results when applied to the recognition of modulation patterns and channel codes, the above methods all use pattern matching to identify known protocols. When faced with unknown signals, the patterns in the matching library can only be matched, and the unknown protocol signals without matching information cannot be processed. For this reason, we want to explore the differences between signals of different protocols by starting from the
characteristics of the signals themselves, so as to classify the signals of unknown protocols, and then conduct efficient and dexterous interference to communication.

3. Unsupervised clustering
Because of blind signal identification, unsupervised clustering analysis is needed. This chapter mainly introduces two kinds of unsupervised clustering methods: subspace clustering and contrastive unsupervised learning.

3.1. Subspace clustering problem
Subspace clustering problem: given a sample of the composition of matrix $X = [x_1, x_2, \cdots, x_n]^T \in R^{n \times m}, x_i \in R^m$, and known that $n$ samples belong to $k$ is the space $S_i, i = 1, \cdots, k$. Set $d_i < m, i = 1, \cdots, k$ represents the dimension of $k$ subspace. The purpose of subspace clustering is to correctly assign the $N$ samples to the corresponding subspace.

In [7], the author proposed Party (deeP subspAce clusteRing with sparsiTY prior), and summarized the general steps of the subspace clustering algorithm based on the spectral clustering algorithm: 1.2. Laplacian Eigenmap(LE) algorithm was used to extract feature vectors on the similarity matrix;3. Other clustering algorithms are used for clustering on these feature vectors. The author finds two defects of the existing algorithms: one is that they only focus on how to construct a good similarity matrix, but ignore how to extract a good expression from the similarity matrix; The current algorithm is linear in nature and can't deal with nonlinearity well. In order to solve the nonlinear problem, [8] and [9] proposed KSSC(Kernel sparse subspace clustering) and KLRR(Kernel low-rank representation), respectively. However, it is difficult to choose an appropriate Kernel for the kernel-based method in practice.[7] That is, from these two perspectives, it is considered how to better embed the similarity matrix information into the low-dimensional representation. Party uses the autoencoder to learn the expression $Z$ of data $X$. Using the idea of spectral clustering, the author uses the similarity matrix $C$ to express the global structure priori, and embedding it into the hidden layer representation $Z$. Its purpose is to hope that the expression of the hidden layer of similar data is similar to achieve the purpose of maintaining the structure. Finally, other clustering algorithms are used to cluster $Z$.

DSC-NET (Deep Subspace Clustering NET) [10], like PARTY, also uses autoencoders to assist Clustering. However, different from PARTY's purpose (learning expression), DSC-NET aims to directly learn similarity matrix $C$ through autoencoder. DSC-NET introduces a self-expressive layer in the middle of traditional autoencoders. The self-presentation layer is a linear layer with no activation functions. Its weight $W_S$ corresponds to the similarity matrix. After learning the similarity matrix, other clustering algorithms can be used for clustering.

DASC(Deep Adversarial Subspace Clustering) [11] is based on DSC-Net and GAN (Generative Adversarial Networks), which is the first successful application of GAN in unsupervised Clustering. The paper points out that if the data in the same subspace is combined linearly to get new data, the new data will still be in the subspace. On the other hand, if you have data that's in a different subspace, if you make a linear combination of it, you get data that's in a different subspace than the original data. In addition, for data in the same subspace, the discriminator (that is, the classifier) cannot tell whether it is real data or fake data, i.e., the output probability is 0.5. Therefore, under the condition that the clustering effect is good, the new data obtained by the linear combination is still in the subspace, so the discriminant cannot judge the true or false. Conversely, the more unable the discriminator is to judge true and false data, the better the clustering effect is. In this paper, DSC-NET and the positive and defective sampling layer are used as generators. After DSC-NET gets the intermediate clustering results, it is input to the positive and defective sampling layer, and then the positive and defective products are input to the discriminant. Through adversarial learning, better similarity matrix and feature expression can be obtained, and the final clustering result can be obtained.

Considering that DSC-NET did not use the results of subspace clustering and did not get an end-to-end trainable framework, [12] proposed a combination of convolution feature, self-representation model and subspace clustering to build an end-to-end trainable network. This paper proposes a new Network,
S2ConvSCN (Self-Supervised Conv. Subspace Clustering Network), which integrates convolution, self-representation model and spectral Clustering into a whole. The spectral clustering results are used to double supervise and train the S2ConvSCN network, and the clustering results are finally obtained.

3.2. Contrastive unsupervised learning

Some recent studies [17,18,19] have proposed the use of contrastive loss [16] related methods for unsupervised visual representation learning. The core idea of these methods is to build dynamic dictionaries. The keys in the dictionary are sampled from the data and are represented by the encoder network. Unsupervised learning trains the encoder to perform dictionary lookups: The encoding of a query should be similar to the encoding of its matching key, but different from that of other keys. Learning is expressed as minimizing contrast loss.

Contrastive learning is a method of constructing discrete dictionaries on high-dimensional continuous inputs such as images. The dictionary is dynamic because the keys are randomly sampled and the key encoder evolves during training. Therefore, contrastive unsupervised learning mainly consists of encoder mechanisms: end-to-end (end-to-end) and memory bank (memory bank). In [18,19], the authors adopt the end-to-end mode, using the current small batch encoding as the dictionary, so the encoding of the key is consistent. However, dictionaries plus batch processing can take up too much GPU memory and can be difficult to optimize in large batch sizes. [18,19] Increase dictionary size in multiple local locations using the pretext task. Pretext requires special network designs, such as input patching [18] and fixed acceptor domain size [19], but this may make it difficult for these networks to apply to downstream tasks.[17] The repository pattern is adopted. The repository consists of the codes of all the samples in the data set. Each small batch dictionary is randomly sampled from the repository, so it can support a large number of dictionaries, but cannot be propagated back.

MoCo (Momentum Contrast) is the unsupervised visual representation learning using Momentum Contrast that was recently proposed [13]. InfoNCE [18] was adopted as the comparative loss function in this paper. Unlike the previously mentioned end-to-end and repository mechanisms, MOCO uses a momentum mechanism, moving average, to update the encoder parameter \( \theta_k \leftarrow m \theta_k + (1 - m) \theta_q \). The key encoder is \( \theta_k \) query encoder is \( \theta_q \), \( m \in [0, 1] \) is the momentum coefficient, of which only query encoder to carry on the back propagation, and \( m \) is set as 0.999, so the dictionary encoder transformation is very smooth. In momentum mechanism, using the queue to save a few small batch recently said the characteristics of the training sample, provide k negative samples. As the training continues, the old small batch are removed from the queue and the characteristics of the new small batch said was to join the queue, guarantee the negative samples in the dictionary from the latest key encoder. Paper also points out using the BN model inhibits learning good said. The article think it might be within the group of communication between a sample (caused by BN) leaked the information, therefore the author puts forward the shuffling BN, which is more GPUs training, each GPU independent of BN samples. The key encoder, first to shuffle the deck of the small Batch sample order, This ensures that the batch statistics used to calculate the query and its correct key are from two different subsets, effectively solving the cheating problem and allowing the training to benefit from the BN.

Geoffrey Hinton's team has proposed SimCLR [15] (Simple framework for Contrastive Learning of visual Representations). Not only is it superior to all previous work, but it is also superior to the latest contrastive unsupervised learning algorithm, and its structure is much simpler: it requires neither a specialized architecture nor a special repository. SIMCLR learns the representation by maximizing consistency between different enhanced views of the same data example through contrast loss in hidden space. Specifically, the framework consists of four main parts: a random data enhancement module that can randomly transform any given data example to produce two related views of the same example, represented as \( x_i \) and \( x_j \), which we consider to be positive pairs; The neural network encoder \( f(\cdot) \) extracts the representation vector from the enhanced data;It is projected by the neural network projection head \( g(\cdot) \), which maps the representation to the space of the contrast loss; The contrast loss function defined for the contrast prediction task. SIMCLR first draws two independent enhancement
functions for each example by taking a small batch at random, and then uses two enhancement mechanisms to generate two interrelated views for each example, making the related views attractive to each other while rejecting the other examples. SimCLR experiments show that the algorithm can reduce the performance gap of linear classifier between unsupervised and pre-trained representations.

Kaiming He's team proposed the second version of MOCO, MOCO v2, by implementing two of the SIMCLR designs (neural projection head and stronger data enhancement) within the MOCO framework [14]. The MOCO framework can handle a large number of negative samples without requiring a large number of training batches, and the MOCO V2 can run on 8-GPU machines and achieve better results than the 4K to 8K batches of SIMCLR that require TPU support.

3.3. Other algorithms
Facebook AI Research Institute has proposed a deep clustering method which combines the learning of neural network parameters and the clustering assignment of acquired features [20]. The network structure proposed in this paper combines the two tasks of clustering and classification. The two tasks use the same network and share the parameters of the network. The results obtained through clustering are treated as pseudo-labels and provided to the network classifier for training and updating the parameters of the network. This mutual learning method is beneficial to the mutual promotion of the two tasks, so that each of them can get better results. The authors propose some solutions to avoid trivial solutions: when a cluster becomes empty, a non-empty cluster is selected randomly, and the centroid with small random perturbation is used as the new centroid of the empty cluster, and then the points belonging to the non-empty cluster are reallocated to the two result clusters; If the input data are not uniformly distributed, the network parameter θ learned will specifically distinguish them. Parameter θ will cause the network to have only the same output. The solution is to resample the input data to make the distribution uniform, or to use false labels.

In [21], the author applied deep clustering to Person re-identification and proposed a bottom-up clustering (BUC) method to jointly optimize the relationship between CNN and untagged samples.

4. Conclusion
In recent years, artificial intelligence methods processing communication signals, a lot of achievements, on the automatic modulation recognition and channel parameter estimation has achieved good results, but these methods are using the mode matching method to known protocol, which can identify when faced with the unknown signal only to match the style of library match, cannot handle does not match the unknown protocol for information signal. Therefore, it is very important to use unsupervised clustering method to extract features and quickly cluster them.

References
[1] Wang J, Wang Y, Li W, et al. Automatic Modulation Classification Method for Multiple Antenna System Based on Convolutional Neural Network[J]. 2020.
[2] Wang Y, Liu M, Yang J, et al. Data-driven deep learning for automatic modulation recognition in cognitive radios[J]. IEEE Transactions on Vehicular Technology, 2019, 68(4): 4074-4077.
[3] Gu H, Wang Y, Hong S, et al. Blind channel identification aided generalized automatic modulation recognition based on deep learning[J]. IEEE Access, 2019, 7: 110722-110729.
[4] Karra K, Kuzdeba S, Petersen J. Modulation recognition using hierarchical deep neural networks[C]//2017 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN). IEEE, 2017: 1-3.
[5] Gautam N, Lall B. Blind Channel Coding Identification of Convolutional encoder and Reed-Solomon encoder using Neural Networks[C]//2020 National Conference on Communications (NCC). IEEE, 2020: 1-6.
[6] Dehdashtian S, Hashemi M, Salehkaleybar S. Deep-Learning Based Blind Recognition of Channel Code Parameters over Candidate Sets under AWGN and Multi-Path Fading Conditions[J]. arXiv preprint arXiv:2009.07774, 2020.
[7] Peng X, Xiao S, Feng J, et al. Deep Subspace Clustering with Sparsity Prior[C]//IJCAI. 2016: 1925-1931.

[8] Patel V M, Vidal R. Kernel sparse subspace clustering[C]//2014 IEEE international conference on image processing (ICIP). IEEE, 2014: 2849-2853.

[9] Xiao S, Tan M, Xu D, et al. Robust kernel low-rank representation[J]. IEEE transactions on neural networks and learning systems, 2015, 27(11): 2268-2281.

[10] Ji P, Zhang T, Li H, et al. Deep subspace clustering networks[J]. Advances in neural information processing systems, 2017, 30: 24-33.

[11] Zhou P, Hou Y, Feng J. Deep adversarial subspace clustering[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 1596-1604.

[12] Zhang J, Li C G, You C, et al. Self-supervised convolutional subspace clustering network[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 5473-5482.

[13] He K, Fan H, Wu Y, et al. Momentum contrast for unsupervised visual representation learning[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 9729-9738.

[14] Chen X, Fan H, Girshick R, et al. Improved baselines with momentum contrastive learning[J]. arXiv preprint arXiv:2003.04297, 2020.

[15] Chen T, Kornblith S, Norouzi M, et al. A simple framework for contrastive learning of visual representations[J]. arXiv preprint arXiv:2002.05709, 2020.

[16] Hadsell R, Chopra S, LeCun Y. Dimensionality reduction by learning an invariant mapping[C]//2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). IEEE, 2006, 2: 1735-1742.

[17] Wu Z, Xiong Y, Yu S X, et al. Unsupervised feature learning via non-parametric instance discrimination[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 3733-3742.

[18] Oord A, Li Y, Vinyals O. Representation learning with contrastive predictive coding[J]. arXiv preprint arXiv:1807.03748, 2018.

[19] Hjelm R D, Fedorov A, Lavoie-Marchildon S, et al. Learning deep representations by mutual information estimation and maximization[J]. arXiv preprint arXiv:1808.06670, 2018.

[20] Caron M, Bojanowski P, Joulin A, et al. Deep clustering for unsupervised learning of visual features[C]//Proceedings of the European Conference on Computer Vision (ECCV). 2018: 132-149.

[21] Lin Y, Dong X, Zheng L, et al. A bottom-up clustering approach to unsupervised person re-identification[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2019, 33: 8738-8745.