Implementation: End to End Deep Learning Architectures for Automatic Modulation Recognition

Timothy Do

February 8th, 2024

1 Introduction

Wireless communication signals go through an end-to-end pipeline for a receiver to reliably receive data from a transmitter. The high-level processes from the input data are channel coding, modulating at some carrier frequency, pulse shaping, transmitting through a channel, then demodulating and decoding the output data. Mathematically, it is modeled as

$$y(t) = x(t) * h(t) + n(t) \tag{1}$$

where the received signal y is the modulated pulse-shaped input signal x convolved with the channel impulse response h in addition to Additive White Gaussian Noise (AWGN) n. Features of wireless signals (e.g., modulation type, channel path loss/fading realizations, channel codes) can be very insightful to know at the receiver to effectively decode the data. The problem lies when these features aren't known at the receiver, thus making reliable estimation and detection methods necessary for learning them.

Automatic Modulation Recognition (AMR) is the process of blindly identifying the type of modulation scheme of a transmitted signal from the receiver end. Its applications range from dynamic spectrum access in cognitive radio to intercepting malicious communications in the military. A high-level block diagram in Fig. 1 incorporates AMR as a decision block to the demodulator to correctly recover the signal.

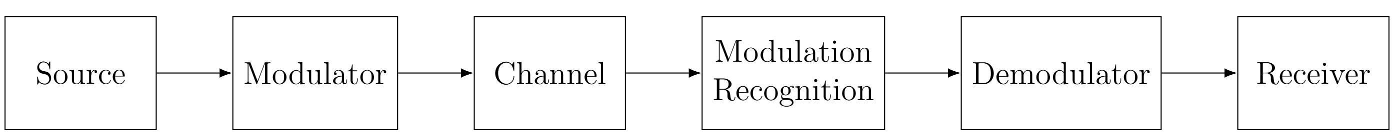

Figure 1: Typical Wireless Commuincations Block Diagram with AMR

This implementation project plans to investigate the task of AMR by classifying with a variety of deep neural network models from end to end. In the field of computer vision, neural network architectures such as CNN (Convolutional Neural Network) from AlexNet [1] yield high classification accuracy in image classification tasks and time-efficiency for training with GPU accelerators.

2 Initial Literature Review

2.1 Datasets

The RadioML dataset is a repository for generating synthetic data for various radio machine learning tasks [2]. It was created using GNURadio [3], a program with various models for modulators, channel coders, demodulators, and dynamic channel impairments like fading. Random channel impairments with SNRs ranged from -20 to 20 dB are applied to each vector. For v2016.10a, there are 11 different modulation classes, which include a mixture of digital and analog modulations. Each vector of RadioML is stored as 128 I/Q (in-phase/quadrature) complex samples (i.e., 2-Wide Columns) with a modulation class and SNR label. The most recent version v2018.01a contains both synthetic and real over-the air data with 24 different modulation class vectors as 1024 I/Q complex samples.

The HisarMod dataset is a repository that aims at being more applicable in real-world scenarios than the RadioML dataset [4]. For v2019.1, it was created using MATLAB v2017a with the Communications Toolbox with SNRs ranging from -20 dB to 18 dB aligning with the RadioML dataset. There is also a uniform distribution of channel fading realizations between Rayleigh, Rician (k=3), and Nakagami models applied throughout the dataset. It has 26 different modulation classes, each with vectors as 1024 I/Q complex samples with a modulation class and SNR label.

2.2 Radio Deep Learning Architectures

Foundational research efforts have shown the effectiveness of CNNs [5] in the task of modulation recognition. Using the RadioML v2016.10a dataset, the authors achieved an overall 87.4% accuracy rate across all SNRs. A follow-up work by the same group used a hybrid architecture known as CLDNN (Convolutional Long Term Short Memory) [6]. It is composed of a 3-layered CNN with concatenation of its first layer fed as the input to the LSTM (Long Term Short Memory) which outperforms the previous CNN. Finally, they trained and tested the architectures on real over-the air data (i.e., RadioML v2018.01a) and with 10 dB SNR, the overall classification rate for all modulation schemes was 95.6% [7].

Another work investigates the performance of LSTM (Long Term Short Memory) models when performing the task of AMR against CNNs [8]. Using the RadioML v2016.10a dataset, they found that in high (10 dB) SNR regimes the LSTM outperformed the CNN with a classification accuracy of 94%, but falls short on low (-2 dB) SNR regimes.

In the industry, researchers at BAE Systems proposed a hierarchical neural network (HNN) with frequency feature extraction [9]. It utilizes multiple classifiers in series to first segment the modulation into two broad classes (i.e., Analog/Digital), then based on the first classifier utilizes another classifier for more specific types of modulations. With the RadioML v2016.10a dataset, they assessed that the overall model performs well over high (18 dB) SNRs at over 90%. However, they couldn't assess the sub-classifier of modulation order effectively because of the lack of training samples, partitioned only from m-order modulations.

3 Implementation Plan & Learning Outcomes

The high-level implementation plan is to train a variety of deep learning models all under PyTorch and the same datasets (i.e., RadioML/HisarMod) to evaluate their AMR accuracies relative to each other. PyTorch [10] is a modern deep learning framework that abstracts high-level neural network layers and provides the backbone to many modern transformer and diffusion models. Additionally, dataset combination/augmentation for increasing the number of training samples will be explored, as shown effective in [11] for image classification.

To successfully execute the implementation of training accurate deep learning models to classify modulation schemes of wireless communication signals, a subset of milestones has been curated to progress smoothly throughout the rest of the quarter by the time of the deadline of the project report. They are as follows:

1. Visualize the features of RadioML/HisarMod datasets and import them in Python.
2. Abstract the deep learning architectures and define their class functions in PyTorch.
3. Train the neural networks on the RadioML/HisarMod dataset based on features with k-fold cross validation to shuffle the training/validation/testing subsets.
4. Experiment with dataset augmentation of the datasets (if dimensionally compatible).
5. Evaluate each deep learning architecture along the metrics of classification accuracy upon varying one dataset parameter at a time (e.g., SNR, Number of training samples).
6. Tabulate results, plotting various figures for different architectures (e.g., Classification Accuracy vs. SNR) and write up the project report.

By following the milestones listed above to the eventual completion of the project and its report, the following learning outcomes will be achieved:

1. Understand how synthetic transmit-receive radio signals datasets are generated to conform to the principles of wireless communications with GNURadio/MATLAB.
2. Incorporate time-efficient inferencing AMR neural networks using GPU accelerators.
3. Assess which deep learning architectures under different SNR regimes yields the best AMR accuracies and if data combination/augmentation improves upon it.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. 25th Int. Conf. Neural Inf. Proc. Syst.*, ser. NIPS'12, vol. 1. Red Hook, NY, USA: Curran Associates Inc., Dec. 2012, pp. 1097—1105.

[2] T. J. O'Shea and N. West, "Radio machine learning dataset generation with gnu radio," *Proc. 6th GNU Radio. Conf.*, Sep. 2016.

[3] E. Blossom, "Gnu radio: tools for exploring the radio frequency spectrum," *Linux J.*, vol. 2004, p. 4, Jan. 2004.

[4] K. Tekbiyik, A. R. Ekti, A. Gorcin, G. K. Kurt, and C. Kececi, "Robust and fast automatic modulation classification with cnn under multipath fading channels," in *Proc. IEEE Veh. Technol. Conf.* IEEE, May 2020. [Online]. Available: http://dx.doi.org/10.1109/VTC2020-Spring48590.2020.9128408

[5] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *Engineering Applications of Neural Networks*, C. Jayne and L. Iliadis, Eds. Cham: Springer International Publishing, Aug. 2016, pp. 213–226.

[6] N. E. West and T. O'Shea, "Deep architectures for modulation recognition," in *Proc. IEEE Int. Symp. Dyn. Spectr. Acc. Netw.*, Mar. 2017, pp. 1–6.

[7] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 168–179, Jan. 2018.

[8] S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, and S. Pollin, "Deep learning models for wireless signal classification with distributed low-cost spectrum sensors," *IEEE Trans. on Cogn. Commun. Netw.*, vol. 4, no. 3, pp. 433–445, Sep. 2018.

[9] K. Karra, S. Kuzdeba, and J. Petersen, "Modulation recognition using hierarchical deep neural networks," in *Proc. IEEE Int. Symp. Dyn. Spectr. Acc. Netw.*, Mar. 2017, pp. 1–3.

[10] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," Dec. 2019.

[11] A. Mikołajczyk and M. Grochowski, "Data augmentation for improving deep learning in image classification problem," in *Proc. Intl. Interdiscip. PhD Wkshp.*, May 2018, pp. 117–122.