

Project Title

SNR-Adaptive Automatic Modulation Classification using Mixture of Experts

Proposed Cluster

- AIML
- ACOM

Team Members

- Bhaskar Bhatt (079BEI013)
- Dharma Raj Thapa (079BEI017)
- Saurab Poudel (079BEI036)
- Sujan Bimali (079BEI044)

Project Overview

Automatic Modulation Classification (AMC) is a key enabling technology for cognitive radio, spectrum monitoring, and intelligent wireless receivers that must identify unknown signals in real time. Conventional deep learning-based AMC systems typically rely on a single model trained over a wide range of signal-to-noise ratios (SNRs) and channel conditions, which often leads to suboptimal performance because low-SNR and high-SNR regimes demand different feature extraction behaviours. Recent literature on deep AMC shows strong performance at specific SNR ranges but still treats the classifier as a one-size-fits-all network.

This project proposes an SNR-adaptive AMC framework based on a Mixture of Experts (MoE) architecture operating directly on complex baseband I/Q samples. The system includes an SNR estimator, a routing or gating network, and multiple specialized expert classifiers, each trained for a particular SNR regime (e.g., low, mid, and high SNR). At inference time, the estimated SNR is used to activate the most relevant experts, and their outputs are fused through soft gating or voting to produce the final modulation decision. The approach will be evaluated on simulated or public RadioML-style datasets containing common digital modulations under AWGN and fading channels, with performance reported as accuracy versus SNR and channel type. By explicitly adapting to SNR, the proposed MoE-based AMC is expected to provide better robustness and efficiency than a single global model, making it attractive for deployment in practical 5G/6G and software-defined radio systems.

Motivation

Most existing deep learning-based AMC systems use a single classifier for all SNR conditions, even though the characteristics of received signals change dramatically between very noisy and high-quality channels. This mismatch can lead to wasted model capacity at high SNR and poor discrimination at low SNR. As undergraduate students interested in advanced communication and machine learning, designing an SNR-adaptive Mixture of Experts architecture allows us to combine theoretical knowledge of SNR, modulation, and channel models with modern deep learning ideas such as expert specialization and gating networks. The goal is to build a system that is not only academically interesting but also novel enough to form the basis of a research publication in AMC and intelligent wireless receivers.

Implementation Plan

The implementation will proceed in several stages:

1. **Literature Review and Problem Definition:** Survey recent work on deep learning-based AMC, SNR-aware models, and mixture-of-experts architectures to refine the exact SNR ranges, modulations, and channel models to be considered.
2. **Dataset Generation and Preprocessing:** Use or generate complex baseband I/Q datasets (e.g., RadioML-style) with common digital modulations over AWGN and fading channels, covering a defined SNR range (e.g., -10 dB to 20 dB). Prepare train/validation/test splits stratified by SNR.
3. **SNR Estimator Design:** Implement a lightweight SNR estimation module that takes I/Q samples as input and outputs an SNR estimate or SNR-bin probability vector suitable for gating.
4. **Expert Networks Training:** Design and train separate expert CNN-based AMC models specialized for different SNR ranges (e.g., low, mid, high SNR), using appropriate subsets of the training data.
5. **Gating and Fusion Mechanism:** Implement a routing/gating network that, given the estimated SNR, selects or weights the outputs of the experts. Explore both hard routing (select one expert) and soft gating (weighted combination).
6. **Evaluation and Analysis:** Compare the proposed SNR-adaptive MoE AMC against a single global CNN baseline in terms of classification accuracy versus SNR, robustness across channels, and model complexity (parameters, inference cost).
7. **Documentation and Reporting:** Prepare code, plots, and a structured report suitable for internal evaluation and potential research paper submission.

A simplified diagrammatic representation of the system is:

Input I/Q Signal → [SNR Estimator] → [Router / Gating Network] ⇒ {Expert_{Low SNR}, Expert_{Mid SNR}, Expert_{High SNR}} → Weighted Combination → Final Modulation Prediction

Use Cases

- Cognitive radio systems that need reliable blind modulation recognition under rapidly changing SNR and channel conditions.
- Spectrum monitoring and regulatory applications where robust classification is required across a wide range of signal qualities and interference scenarios.
- Software-defined radio (SDR) testbeds for 5G/6G physical-layer research, where adaptive AMC can improve link adaptation and resource allocation.
- Defence, surveillance, and electronic warfare systems requiring flexible signal intelligence and automatic identification of unknown emitters.

Resources Required

- **Computing Resources:**
 - Workstation or laptop with multi-core CPU and at least 16 GB RAM.
 - Access to a GPU (e.g., NVIDIA GTX/RTX series) for training multiple expert networks efficiently.
 - Python environment with libraries such as NumPy, SciPy, PyTorch or TensorFlow, Matplotlib, and scikit-learn.
- **Software and Datasets:**
 - Public RadioML-style I/Q datasets or locally generated simulated datasets covering a range of modulations, SNRs, and channels.
 - Version control using Git and a code hosting platform (e.g., GitHub or GitLab) for collaborative development.
- **Hardware (Optional):**
 - Software-defined radio (SDR) hardware (e.g., USRP, RTL-SDR) for capturing real over-the-air signals to test the SNR-adaptive AMC in practical environments.