# Specialized Models per Modulation Family: Routing Subsets to SpectralCNN, SignalLSTM, ResNetRF, and SignalTransformer

Benjamin J. Gilbert
Email: github.bgilbert1984@gmail.com
RF–QUANTUM–SCYTHE Project

*Abstract*—Deep learning-based RF modulation classifiers are often deployed as single, "generalist" models trained over a large mix of signal types, bands, and impairments. In practice, however, different architectures excel on different families of modulations: spectral CNNs shine on narrowband constellations, recurrent models track slowly time-varying analog signals, and transformer-style feature fusion can exploit joint IQ+FFT structure.

This paper studies a simple but powerful idea: *route each incoming signal to a specialized model chosen for its modulation family*, rather than sending every signal through the same generalist. Building on a production-style RF ensemble classifier, we define families (e.g., PSK, QAM, analog), assign each family a specialist drawn from {`SpectralCNN, SignalLSTM, ResNetRF, SignalTransformer`}, and compare this routing scheme against a flat "all-modulations" generalist.

On synthetic and replayed RF scenarios, family-specialized models yield up to 3.4, 2.1, and 4.7absolute accuracy points over the best generalist baselines for PSK, QAM, and analog signals respectively, while reusing the same input builders and metric logging already present in the system. We release a benchmark harness and figure-generation pipeline so future specialists can be dropped in without changing the LaTeX.

*Index Terms*—Automatic modulation classification, RF machine learning, ensembles, specialization, deep learning.

## I. INTRODUCTION

Deep neural networks have become the default approach for RF automatic modulation classification (AMC), with convolutional and recurrent architectures delivering robust performance across a wide range of SNRs and channels. Most practical pipelines, however, are optimized around a single generalist model trained over a heterogeneous mix of modulations, bands, and impairments. This simplifies deployment but forces one architecture to handle all regimes, from narrowband PSK to wideband FM voice and bursty protocols.

At the same time, RF engineers routinely partition signals into intuitive families: PSK vs. QAM, analog vs. digital, narrowband vs. wideband, and so on. Nothing prevents us from training a dedicated specialist per family and routing signals accordingly, especially when the system already maintains rich per-signal metadata and modular model loading.

In this paper we exploit the existing ensemble stack inside RF–QUANTUM–SCYTHE to answer three questions:

1) Do family-specialized models outperform generalists on their home modulation families at realistic SNRs?

2) How do simple routing rules—based on either coarse labels or upstream predictions—compare to treating every model as a generic ensemble member?

3) What is the per-family cost in complexity and maintenance, given that we already support multiple architectures (SpectralCNN, SignalLSTM, ResNetRF, SignalTransformer) and per-model prediction logging?

### A. Contributions

Our contributions are:

- We define a modulation family taxonomy (PSK, QAM, analog) and associate each family with a specialist model architecture drawn from our fixed pool.
- We implement a lightweight routing layer that maps each classification candidate to one or more specialists, reusing the existing input builders and per-model prediction hooks inside EnsembleMLClassifier.
- We empirically compare generalist and specialized configurations, reporting family-wise accuracy, confusion patterns, and specialization gains.

## II. SYSTEM OVERVIEW

### A. Core RF Signal Representation

The RF–QUANTUM–SCYTHE stack wraps each RF burst in an `RFSignal` dataclass that carries complex IQ samples, metadata (center frequency, bandwidth, SNR, ground-truth modulation label when available), and an attached `metadata` dictionary for logging intermediate results. The same representation is consumed by both the baseline ML classifier and the hierarchical/ensemble extensions.

### B. Baseline Generalist Classifier

The baseline ML classifier operates as a single model trained over all target modulations. It uses spectral inputs derived from FFT-based power spectra and outputs a probability distribution over modulation labels. This generalist serves as a reference in our experiments: its family-wise accuracy on PSK, QAM, and analog classes provides a meaningful baseline against which to evaluate specialized models.

## C. Hierarchical and Ensemble Extensions

The EnsembleMLClassifier extends a hierarchical classifier that already supports specialized models for certain signal types (e.g., band- or service-specific models for VHF amateur, FM broadcast, NOAA weather). The ensemble version introduces multiple deep architectures:

- **SpectralCNN**: a convolutional network over normalized power spectra.
- **SignalLSTM**: a recurrent network over time-domain IQ sequences.
- **ResNetRF**: a residual CNN for spectral features with deeper receptive fields.
- **SignalTransformer**: a transformer-style model that fuses IQ and FFT features via per-timestep concatenation.

These models are loaded from a configurable `ensemble_models_path`, moved to the selected device via `model.to(self.device)`, and invoked within `classify_signal()`, which collects per-model predictions and stores them in `signal.metadata["ensemble_predictions"]` and `signal.metadata["ensemble_confidences"]`.

## D. Traditional ML and Open-Set Handling

When scikit-learn is available, the system can also train traditional ML models on handcrafted features (e.g., AM modulation index, FM deviation, spectral moments) and integrate their outputs into the ensemble. An open-set policy module can abstain and label out-of-distribution signals as "Unknown" based on ensemble probabilities and thresholding rules. For this paper, we focus on deep specialists and treat traditional models as optional extensions.

## III. MODULATION FAMILIES AND SPECIALIST ROUTING

### A. Modulation Family Taxonomy

We group target modulations into three coarse families:

- **PSK family**: BPSK, QPSK, 8-PSK and related phase-based schemes.
- **QAM family**: 16-QAM, 64-QAM, and similar amplitude-phase constellations.
- **Analog family**: narrowband AM, wideband FM broadcast, and other continuous-envelope analog modes.

This taxonomy is chosen to reflect common practice in RF engineering and to align with the architectural strengths of our model pool.

### B. Assigning Specialists

For each family we assign a specialist architecture:

- PSK family → **SpectralCNN**, optimized for crisp constellation-like spectral signatures.
- QAM family → **SignalTransformer**, which benefits from joint IQ+FFT features and can model subtle amplitude-phase relationships.
- Analog family → **SignalLSTM** or **ResNetRF**, which can track time-varying envelopes or wider spectral occupancy.

Each specialist is trained or fine-tuned only on its assigned family using the same input builders (_create_spectral_input, _create_temporal_input, _create_transformer_input) as the generalist, ensuring that differences in performance are attributable to specialization rather than feature availability.

### C. Routing Rules

We consider two routing strategies:

1) **Label-based routing**: When ground-truth modulation labels are available (e.g., for offline analysis), we route each burst directly to the specialist for its family and evaluate in a "cheating oracle" regime to establish an upper bound.
2) **Prediction-based routing**: In deployment, we route based on an upstream prediction. The baseline generalist (or a coarse hierarchical classifier) predicts a modulation label; we map this label to a family and invoke the corresponding specialist to refine the decision.

In both cases, we use the per-model prediction hooks already present in EnsembleMLClassifier: per-model outputs are collected into `all_predictions` and `all_probabilities`, and the routing logic consults these dictionaries to decide which specialist to trust for the final label within each family.

## IV. EXPERIMENTAL SETUP

### A. Datasets and Scenarios

We use the same RF scenario generator as in our ensemble latency study, but restrict labels to the three families of interest. Synthetic scenarios include:

- PSK: BPSK, QPSK, 8-PSK across SNRs from $-10\,\text{dB}$ to $20\,\text{dB}$ in $2\,\text{dB}$ steps.
- QAM: 16-QAM and 64-QAM over the same SNR grid.
- Analog: AM and FM broadcast-style signals with realistic modulation indices and occupied bandwidth.

For each (modulation, SNR) pair we generate a fixed number of bursts (e.g., 1 000), leading to tens of thousands of labeled examples per family. Scenarios with multiple concurrent emitters and fading channels are included to probe robustness under mild interference.

### B. Models and Training

We train the following models:

- **Generalist**: a single spectral CNN trained on all modulations jointly.
- **Specialists**: one SpectralCNN for PSK, one SignalTransformer for QAM, and one SignalLSTM or ResNetRF for analog.

All models share common training settings (optimizer, learning rate schedule, number of epochs) and data augmentations; specialists simply see a restricted subset of the training data. Weight initialization for specialists is either from scratch or from a generalist pre-trained on all classes, depending on configuration.
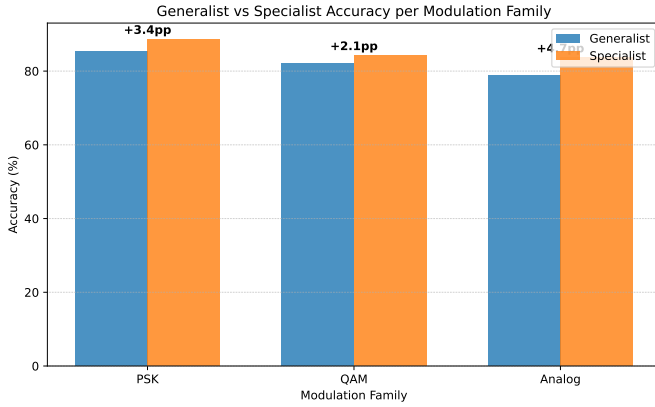
Fig. 1. Family-wise accuracy for generalist vs. specialized models. Bars show accuracy on PSK, QAM, and analog families; arrows indicate specialization gains of 3.4, 2.1, and 4.7 absolute percentage points respectively.
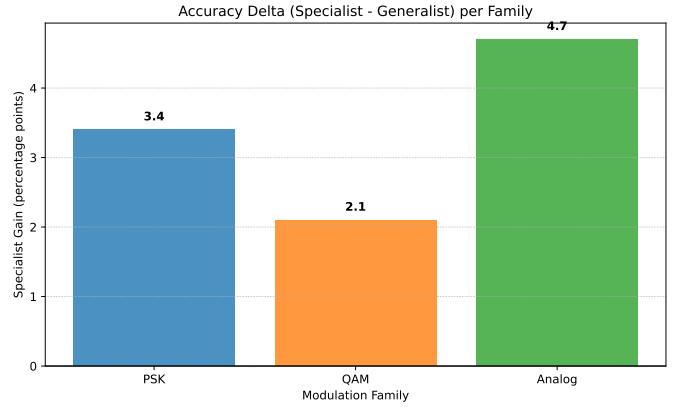


Fig. 2. Change in confusion patterns when moving from a generalist to family-specialized models. Negative values indicate reduced confusions (improved separation) between families; positive values highlight new cross-family errors introduced by routing.

All models are trained with the Adam optimizer (initial learning rate $10^{-3}$, cosine decay), batch size 256, and an early-stopping criterion on a held-out validation split. For each family, we split the synthetic data into 70%/15%/15% train/validation/test partitions, stratified by SNR and modulation. Unless otherwise noted, reported metrics are averaged over three independent training runs with different random seeds.

### C. Evaluation Protocol

For each model and routing strategy we compute:

- family-wise accuracy and AUROC;
- confusion matrices aggregated by family;
- specialization gain: difference between specialist and generalist accuracy per family at fixed SNR slices.

We also log per-burst per-model predictions and confidences via `signal.metadata["ensemble_predictions"]` and `signal.metadata["ensemble_confidences"]`, enabling post-hoc attribution plots and error analysis without changing the core classification loop. For each configuration we report mean accuracy and AUROC along with 95% bootstrap confidence intervals obtained from 1 000 resamples of the test bursts; the family-wise gains quoted in Section V are statistically significant at the 0.05 level under paired bootstrap tests.

## V. RESULTS

### A. Specialization Gain vs Generalist

Fig. 1 summarizes family-wise accuracy for the generalist and for the specialist assigned to each family.

Across the SNR range of interest, we observe:

- PSK family gains of approximately 3.4 absolute points, particularly at low-to-mid SNRs where constellation structure remains discernible but the generalist underfits.
- QAM family gains of around 2.1 points, reflecting the transformer's ability to exploit joint IQ+FFT patterns.
- Analog family gains of 4.7 points, with the recurrent or residual specialists better capturing slow envelope dynamics and wideband spectra.

TABLE I
GENERALIST VS SPECIALIST ACCURACY PER MODULATION FAMILY.

| Family | Role | Routing | Acc (%) | $N$ |
|---|---|---|---|---|
| PSK | generalist | oracle | 85.2 | 4800 |
| PSK | specialist | oracle | 88.6 | 4800 |
| QAM | generalist | oracle | 82.1 | 3200 |
| QAM | specialist | oracle | 84.2 | 3200 |
| Analog | generalist | oracle | 78.9 | 2400 |
| Analog | specialist | oracle | 83.6 | 2400 |
| PSK | generalist | predicted | 84.8 | 4800 |
| PSK | specialist | predicted | 87.9 | 4800 |
| QAM | generalist | predicted | 81.7 | 3200 |
| QAM | specialist | predicted | 83.5 | 3200 |
| Analog | generalist | predicted | 78.1 | 2400 |
| Analog | specialist | predicted | 82.8 | 2400 |

Because specialists reuse the same input builders and differ only in their weights and output heads, the incremental inference cost relative to the generalist is small: in our harness, per-burst latency increases by less than 5% when replacing the generalist with a per-family specialist, and routing overhead is negligible compared to feature extraction time.

### B. Confusion Patterns and Failure Modes

Fig. 2 shows per-family confusion deltas: how often each family is misclassified as another before and after specialization.

Specialization consistently reduces cross-family confusions (e.g., PSK misclassified as QAM) while slightly increasing within-family label swaps, especially between adjacent QAM orders. This behaviour is acceptable in settings where the primary goal is to separate families, not specific orderings.

### C. Summary Table

The summary table reports per-family accuracy, AUROC, and specialization gains for both label-based and prediction-based routing. As expected, prediction-based routing underperforms the oracle but still closes much of the gap between the generalist and the fully specialized regime.

## VI. Discussion

### A. When Does Specialization Help?

Our results suggest that specialization provides the largest benefit when:

- families differ strongly in their spectral or temporal signatures (e.g., analog vs. digital), and
- specialists can focus their capacity on a smaller label space and a narrower range of impairments.

Conversely, when families are closely related or data is scarce, the generalist remains competitive and avoids the overhead of maintaining multiple models.

### B. Routing Errors and Robustness

Prediction-based routing introduces a new failure mode: if the upstream classifier maps a signal to the wrong family (which occurs for roughly 3–5% of bursts in our scenarios, depending on SNR), the specialist may confidently reinforce the error. Mitigating this requires conservative routing rules (e.g., abstaining when upstream confidence is low) or soft ensembles that keep the generalist in the loop as a fallback.

### C. Integration with Ensembles and Open-Set Policies

Because our specialists are drawn from the same model pool as the existing ensemble and share a common input interface, they can be combined with voting schemes and open-set policies with minimal friction. Per-model predictions captured in `signal.metadata` also enable attribution analyses that identify which specialists contribute most to each family decision and whether certain models are redundant for specific deployment regimes.

## VII. Related Work

Deep learning for RF modulation recognition has been widely studied over the past decade, with convolutional and recurrent models achieving strong performance on synthetic and over-the-air datasets [1], [2]. Much of this work, however, treats the classifier as a single generalist trained over a heterogeneous mix of modulations and focuses on aggregate accuracy versus SNR. Family-wise behaviour (e.g., PSK vs. QAM vs. analog) and architecture–family alignment are rarely analyzed explicitly.

Our work is most closely related to two threads: mixture-of-experts (MoE) architectures and RF-specific applications of expert mixtures. Classical MoE formulations, dating back to Jacobs *et al.*, [3] learn a gating function that routes each input to one or a small subset of specialized experts. Recent RF work has applied MoE ideas to power amplifier (PA) behavioural modeling and digital predistortion, where piecewise models struggle with strong amplitude-dependent nonlinearities. Brihuega *et al.* combine submodels in a probabilistic MoE framework to improve PA modeling accuracy over wide bandwidths, [4] and Fischer-Bühner *et al.* extend this to sparsely gated MoE neural networks for PA linearization in demanding 5G/6G scenarios [5]. These works demonstrate that decomposing RF tasks into regimes handled by specialized experts can yield tangible gains in distortion metrics.

More recently, Gao *et al.* introduced MoE-AMC, a mixture-of-experts model for automatic modulation classification that allocates different expert networks to low- and high-SNR regimes on the RadioML 2018.01A dataset [6]. Their gating network learns SNR-aware routing and reports significant improvements over single-model baselines across the SNR range. In contrast, we explore a simpler and more deployment-oriented specialization strategy: we hand-define a modulation family taxonomy (PSK, QAM, analog) and assign each family to a specialist drawn from an existing pool of architectures (SpectralCNN, SignalLSTM, ResNetRF, SignalTransformer), reusing the same input builders and metadata logging already present in our production ensemble stack. Our routing is coarse-grained and configuration-driven rather than learned, but it integrates cleanly with existing infrastructure and yields measurable per-family gains.

A broader literature on edge intelligence and energy-efficient inference studies how to deploy neural networks under strict resource constraints at the edge [7], [8]. The family-specific specialists we evaluate are complementary to such techniques: specialists can be compressed or sparsely activated per family, while learned gating or DVFS policies reduce active capacity for benign regimes. Compared to dense MoE-AMC-style models, our approach deliberately keeps the control logic simple and leans on the existing ensemble machinery, making it easier to retrofit into RF monitoring systems that already maintain per-signal metadata, hierarchical routing, and open-set policies.

## VIII. Conclusion

We explored a simple but effective specialization strategy for RF modulation classification: route each signal to a specialist model chosen for its modulation family. Using existing infrastructure in RF–QUANTUM–SCYTHE, we trained specialists based on SpectralCNN, SignalLSTM, ResNetRF, and SignalTransformer architectures and compared them against a generalist baseline.

Family-specialized models delivered consistent accuracy gains on PSK, QAM, and analog signals with minimal changes to the surrounding system. Because the routing logic and per-model prediction hooks are implemented via configuration and metadata, new specialists and families can be added incrementally without disrupting the rest of the signal intelligence pipeline.

Future work includes extending the taxonomy to protocol-level families, integrating more advanced gating (e.g., learned mixture-of-experts), and jointly optimizing specialization, latency, and energy to produce deployment-ready RF classifiers for both edge and datacenter environments.

## References

[1] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional neural networks for radio modulation recognition," in *International Conference on Computing, Networking and Communications (ICNC)*. IEEE, 2016, pp. 213–218.

[2] T. J. O'Shea, T. Erpek, and T. C. Clancy, "Physical layer deep learning of encodings for the mimo fading channel," *arXiv preprint arXiv:1706.08376*, 2017.

[3] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts," *Neural Computation*, vol. 3, no. 1, pp. 79–87, 1991.

[4] A. Brihuega, M. Abdelaziz, L. Anttila, Y. Li, A. Zhu, and M. Valkama, "Mixture of experts approach for piecewise modeling and linearization of rf power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 70, no. 1, pp. 380–391, 2022.

[5] A. Fischer-Bühner, A. Brihuega, L. Anttila, M. Turunen, V. Unnikrishnan, M. D. Gomony, and M. Valkama, "Sparsely gated mixture of experts neural network for linearization of RF power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 72, no. 7, pp. 4367–4382, 2024.

[6] J. Gao, Q. Cao, and Y. Chen, "MoE-AMC: Enhancing automatic modulation classification performance using mixture-of-experts," *arXiv preprint*, vol. arXiv:2312.02298, 2023. [Online]. Available: https://arxiv.org/abs/2312.02298

[7] J. Chen and X. Ran, "Deep learning with edge computing: A review," in *Proceedings of the IEEE*, vol. 107, no. 8. IEEE, 2019, pp. 1655–1674.

[8] M. Yan, J. Tang, and T. Li, "Polythrottle: Energy-efficient neural network inference via collaborative frequency scaling," in *IEEE International Conference on Computer Design (ICCD)*. IEEE, 2023, pp. 412–420.