Towards scalable zero-shot modulation recognition

Wei Xiong, Petko Bogdanov, Mariya Zheleva
Department of Computer Science, University at Albany – SUNY
{*wxiong, pbogdanov, mzheleva*}*@albany.edu*

Abstract*—With the advent of Dynamic Spectrum Access networks, practical modulation recognition (ModRec) has become an important problem with critical applications to spectrum enforcement and resource allocation. Existing ModRec frameworks hinge on exhaustive classifier training and require labeled observations for all modulations before recognition can be performed. As a result, a general ModRec system based on fully supervised classification quickly becomes intractable due to the emergence of new proprietary waveforms and the need to collect labeled training data for them and retrain the system. This underpins the need to perform modulation classification of previously unobserved modulations. Thus, the question emerges *can learned classifiers be transferred to "unobserved" modulations?

In this paper, we investigate the utility of zero-shot learning to address the above challenge in the context of ModRec. We design ModRec-0, which employs unseen modulations' constellation diagrams within a zero-shot framework to transfer learned classifiers from observed modulations. We propose several important properties to enable this transfer, such as the number of constellation points, phase and amplitude levels. Our system is flexible in that additional such properties as well as a variety of classification features can be incorporated. We evaluate our framework on synthetic and real over-the-air traces and investigate the necessary conditions such as SNR levels and observed training classes for successful transfer. We demonstrate that ModRec-0 achieves over 85% average accuracy across multiple modulation classes both in synthetic and real-world traces.

I. Introduction

The goal of modulation recognition (ModRec) is to determine the modulation of a passively sensed input signal based on pre-trained classifier or a likelihood model [1]. ModRec has emerged as an important component of future wireless networks with applications in both civil and military contexts. Civil ModRec applications include resource allocation and spectrum enforcement, while military applications range from identifying adversary transmitting units to efficient jamming [2]. ModRec approaches can be grouped in two families: likelihood based (LB) and feature based (FB) [1]. LB methods evaluate the likelihood of candidate modulations given observations, while FB approaches extract features from observations and train classifiers which are employed for recognition. One drawback of LB methods is their high computational complexity [1]. FB approaches are more scalable, drawing on the tremendous advances in machine learning, but require labeled training data for all modulations of interest.

The existing literature on FB contributes important advances on improving the employed classifier models [3]–[5] as well as on feature engineering [6]–[9]. Common to all of them is the critical dependency of their accuracy on the diversity and number of training instances [10]. In many application scenarios, however, the collection of training instances for every modulation class can be prohibitive or impractical altogether, as some emerging modulations (e.g. Quadrature Spatial Modulation [11]) are seldom widely available, and thus, hard to train for. Another challenging scenario is the recognition of customized versions of popular modulations such as phase rotations and shape variance [12]. Methods from the transfer learning literature, such as zero-shot learning, bring promise for robust ModRec with limited supervision. Recent work from image processing proposed generalized representations enabling between-class attribute transfer [13] and sharing of features between classes and generalized attributes [14]. Beyond new representations, various classifiers have been considered for the zero-shot learning: from traditional ones [15] to novel deep learning architectures [16].

Our goal in this paper is to investigate the feasibility of feature-based modulation classification with limited supervision. Our key insight is that modulations often share properties depending on how they manipulate the phase, frequency and amplitude of the carrier. For example, all modulations from the phase-shift keying family manipulate the phase but not the amplitude and frequency. We design ModRec-0, which exploits shared properties across modulations in support of ModRec with limited supervision. We differentiate between *observed* and *unobserved* modulations. "Observed" are modulations for which we have labeled training data, while "unobserved" are modulations for which we do not have training data, but need to classify. Our zero-shot framework learns from observed classes a generalized feature representation which can then be employed for classification of unobserved modulations [17], [18]. To this end ModRec-0 utilizes a semantic representation space of the modulation domain and a linear classifier model. Our semantic embedding space summarizes key "theoretical" modulation attributes such as amplitude and phase levels, and is flexible in that it could be extended with other attributes of interest. We evaluate ModRec-0's performance with simulated and over-the-air transmissions.

The contributions of our work are as follows:

- **Novelty:** Our work is the first to conceptualize semantic representations of modulations and evaluate the utility of zero-shot learning for feature-based ModRec. We show the feasibility of ModRec of previously unobserved signals.
- **Generality:** While we instantiate the semantic attribute matrix—a key component of zero-shot transfer learning—with specific attributes, it can be easily extended with additional ones. Similarly, alternative classification approaches and features can be adopted in the task.

- **Applicability:** We study the strengths and limitations of our approach using both simulation and over-the-air traces and illustrate its applicability in both simulated as well as real-world wireless channel conditions. We map out the feasibility and future challenges towards a zero-shot ModRec deployment.

II. Problem definition and notation

Next we present the general pipeline of feature-based (FB) ModRec and typical features extracted from in-phase and quadrature components (IQ) formatted signals.

ModRec as supervised learning. Supervised FB ModRec employs a training dataset $\{X, y\}$, where $X \in \mathbb{R}^{d \times n}$ is a set of d-dimensional instances each representing a sequence of measured IQ samples and $y \in \{c_0, c_1 \dots c_k\}^{n \times 1}$ is a class vector encoding type of modulation giving rise to each instance. The goal is then to learn a classifier which maps *unobserved* instances to one of the *observed* classes $f(x) \rightarrow \{c_0, c_1 \dots c_k\}$. In the case of zero-shot learning, discussed in the next section, we will seek to further be able to "reuse" this function to predict a set of classes unobserved during training, i.e., $f_0(x) \rightarrow \{c_{k+1} \dots c_{k+k'}\}$.

Mapping IQ samples to features. Feature engineering is an important step in FB ModRec [1]. Let $(r_1, r_2, ..r_m)$ denote a sequence of m consecutive IQ samples sensed via a passive sensor, where $r_i \in \mathbb{C}$ is the instantaneous complex signal sampled at time i. Existing literature typically assumes a signal model with a channel effect of additive Gaussian white noise (AWGN), which can be represented as:

$$r(n) = s(n) + w(n), \tag{1}$$

where $w(n) \sim \Phi(0, \sigma^2)$ is a zero-mean Gaussian noise term. The feature engineering process in FB ModRec maps the input IQ signal to a d-dimensional feature space: $g : \mathbb{C}^m \rightarrow \mathbb{R}^d$. The feature representations of all obtained instances are then employed for training the classifier model.

High order cumulants which summarize statistical properties of observed IQ samples are among the most commonly used features [6], [7], [19]. More recent work has also considered order statistics [8] and patterns in the local order of IQ samples [9]. Note that we focus on cumulants as features for the evaluation in this work, however, a thorough consideration of other features is warranted and we plan to pursue this as part of our future work on extending the current analysis.

III. ModRec-0: Zero-shot learning for ModRec

In this section, we outline our proposed approach. First we describe our zero-shot pipeline and methodology. Then we detail the ModRec semantic attributes, an imperative component employed in the pipeline.

A zero-shot pipeline for ModRec. A key difference in the zero-shot setting in comparison to the classical fully-supervised case is that we need to learn a discriminator $f_0(x) \rightarrow \{c_{k+1} \dots c_{k+k'}\}$ for *unobserved* classes. Intuitively, the idea is to learn an association V between features of training classes and semantic attributes S available for both training $\{c_1 \dots c_k\}$ and testing $\{c_{k+1} \dots c_{k+k'}\}$ classes.

Similar to the fully-supervised case, the training input in zero-shot learning is a dataset $\{X, Y\}$, of N training instances, i.e., samples from modulations for which we know the ground truth modulation. Note that instead of a class indicator vector, we learn to predict the association of each instance with one or more of the training classes encoded in matrix $Y \in \{0, 1\}^{N \times k}$. We adopt the linear classifier framework originally proposed by Romera-Paredes et al. [20] which can also be kernelized to perform non-linear classification. The general linear classification scheme learns a discriminator function $f : X \rightarrow Y$ by minimizing a regularized empirical risk:

$$\min_W \; L(f(X, W), Y) + \Omega(W), \tag{2}$$

where $L(f(X, W), Y)$ quantifies the training error (or empirical misclassification) and $\Omega(W)$ imposes a desired shape on the classifier(s). Within this general framework, one can express a linear Support Vector Machine (SVM) objective by choosing a hinge loss for the shape of $L(X^T W, Y)$ and the Frobenius norm which imposes small margin for the shape of $\Omega(W)$. With an SVM objective, the columns of W are separating hyperplanes for each of the classes.

The novel challenge with zero-shot learning is that the training $\{c_0, c_1 \dots c_k\}$ and testing $\{c_{k+1} \dots c_{k+k'}\}$ classes are disjoint. Therefore the trained classifiers in W need to be transferred to classes for which we have not observed training instances. The zero shot approach to achieve this is to employ an intermediate representation space, which acts as a bridge between knowledge of *unobserved* and *observed* classes.

Lampert et al. [13] call such intermediary representations *attributes* (not to be confused with the features representing instances). Within this framework all classes (both *observed* and *unobserved*) are represented by a finite set of attributes S, each having a real-world interpretation [21]. We will discuss the design of attributes for modulations in the following section and will assume that we have a mapping from all classes to attributes in a matrix $S \in \mathcal{R}^{k' \times m}$, where k' is the total number of classes and m is the number of semantic attributes.

Given the attribute representation of classes S the general goal in transfer learning is to use the semantic attribute space to transfer the classifiers for training classes to those for testing [15]. Romera-Perendes and colleagues [20] demonstrated a simple yet effective learning process whose key idea is to learn predictors V for the semantic embedding of classes S as opposed to directly to the class matrix Y. By letting $W = VS$ in Eq. (2) for the SVM case one obtains the following objective:

$$\min_V \; L(X^T VS, Y) + \Omega(V), \text{where} \tag{3}$$

$$\Omega(V) = \gamma ||VS||_F^2 + \lambda ||X^T V||_F^2 + \beta ||V||_F^2. \tag{4}$$

Note that the first two ridge regularization terms shrink the feature space and perform semantic projection on the hyperplanes in V, while the third term follows the classical margin minimization criterion in SVM. All regularization are convex and so is the overall objective in Eq. (3). In addition there is an efficient closed-form solution for V due to [20]. Regularization weights γ, λ and β can be set based on cross-validation within the training set.

In order to predict testing classes, we first project a new instance x into attribute space via the discriminators learned in V, thus obtaining a semantic space representation of the instance $x^T V$. This representation is then compared with all training class semantic representations S' and the class of highest agreement (inner product value in semantic space) is predicted similar to the process in SVM:

$$\underset{i \in c_{k+1} \cdots c_{k'}}{\operatorname{argmax}} \ x^T V S'_i \tag{5}$$

where S'_i is the semantic representation vector (attribute values) of the i-th training class.

The right pane of Figure 1 provides an intuitive visualization of the training and prediction processes in `ModRec-0`. The input to our framework (left pane) is comprised of a semantic attribute matrix for all modulations, both *observed* (i.e. training) and *unobserved* (i.e. testing); and a set of labeled instances for all training modulations. In the training phase (top right pane), we take the labeled training instances and their respective attributes and seek to learn the corresponding classifier V. We then transfer the learned classifier to the prediction stage. Upon receiving a previously *unobserved* instance, we use the learnt classifier along with the attribute matrix for the testing classes to find the most likely modulation type corresponding to the measured instance.

The accuracy of our framework depends on the design of the semantic attributes embedding S, which has to facilitate maximal knowledge transfer. In what follows, we propose one possible such embedding S, however, the framework is flexible in that S can be further extended within the same framework. An important practical advantage of the methodology is that adding a *new* target modulation class does not require re-training of the model V. Instead, one needs to add a corresponding attribute representation to S' in order to accommodate prediction for this new class.

Design of modulation attributes embedding S. Semantic attributes capture the intrinsic characteristics of a class. For example, recent work from the image segmentation literature [13] uses visually-observable features of a class such as "is yellow" and "has a long neck", whereas others [22] use subjectively-perceived features such as "is cute". Various approaches to attribute design have been employed ranging from fully expert designed [23] to fully data-driven [24]. In the image segmentation literature, semantic attributes are typically acquired by manual labeling (e.g. a Mechanical Turk task). Similarly, in the modulation recognition domain, attributes will draw on expert input.

Using expert knowledge, we design our semantic attributes, all of which draw on the modulation's constellation shape and are illustrated in the left pane of Figure 1:

1) *Number of unique amplitude levels A.* As the name suggests, we count the unique amplitude levels per modulation. For BPSK, QPSK and 8-PSK this would be 1, whereas 16-, 64- and 128-QAM have 3, 9 and 16 distinct levels, respectively.
2) *Number of unique phase levels ϕ.* We define a phase level as the phase at which a given constellation point falls on a Polar coordinate system. For example, BPSK has two constellation points, which are typically at 0° and 180°, whereas QPSK has four constellation symbols at 0°, 90°, 180° and 270°.
3) *Normalized pairwise distance of constellation points.* These attributes encode the spread of a constellation. We obtain them by first calculating the pairwise distance between all points in a constellation. We then note the minimum (d_{min}), median (d_{med}) and maximum (d_{max}) observed pairwise distances. We derive two distance attributes per modulation class, namely the normalized minimum distance $D_{min} = d_{min}/d_{max}$ and the normalized median distance $D_{med} = d_{med}/d_{max}$.
4) *Constellation overlap at rotation $\theta°$.* A modulation's constellation can be thought of as a set of points in a Cartesian coordinate system with an origin $\mathcal{O}(0,0)$. We can create a rotated version of the original constellation by keeping the origin fixed, while rotating the x and y axes at an angle θ. Thus, for this attribute we take all unique rotations of θ_i° for which the derived constellation shape at θ_i° rotation completely overlaps with the original constellation shape at rotation 0°. For example, for BPSK the original and derived constellations overlap at 180°, whereas for QPSK this overlap happens at 90°. We implement this by varying θ from 0° to 180° in increments of 5° and taking only the rotation angles for which the derived constellation fully overlaps with the original.

All attribute values are bounded between 0 and 1 [20]. This is satisfied for our attributes as they either take a binary value (either 0 or 1) or real value normalized between 0 and 1. Specifically, A, ϕ and θ take binary values (e.g. "unique amplitude level of 3 or different"), whereas D_{min} and D_{med} take continuous values normalized between 0 and 1.

We derive the attribute matrix as follows. Assume that for an attribute such as the number of amplitude levels, all modulations yield a collection of measurement denoted as $A = \{A_1, A_2, \dots A_M\}$, where M equal the number of modulations. From collection A we calculate unique elements as the set $B = \{B_1, B_2, \dots B_K\}$, $K \leq M$. For each modulation, if an attribute (say A_m) is equal to an element (say B_k), the k-th column in the attribute matrix would be encoded as 1 and otherwise as 0. This process is adopted for the representation of all modulations. Finally, we concatenate columns from each attribute category (i.e. all A's, all ϕ's, all θ's, D_{min} and D_{med}) to form S. To keep the matrix compact, we do the following post-processing: (1) omit columns of constant values and (2) retain only one from a group of duplicate columns. Fig. 2 illustrates the compacted attribute matrix annotated for the six modulation classes used in our experimentation.

The attribute values for each modulation can be obtained from the theoretical constellation diagrams and, thus, do not require any empirical observation of the corresponding signals. Thus, employing the above pipeline, we can easily add a new modulation to the attribute matrix as long as we have its constellation projection onto a Polar coordinate system.

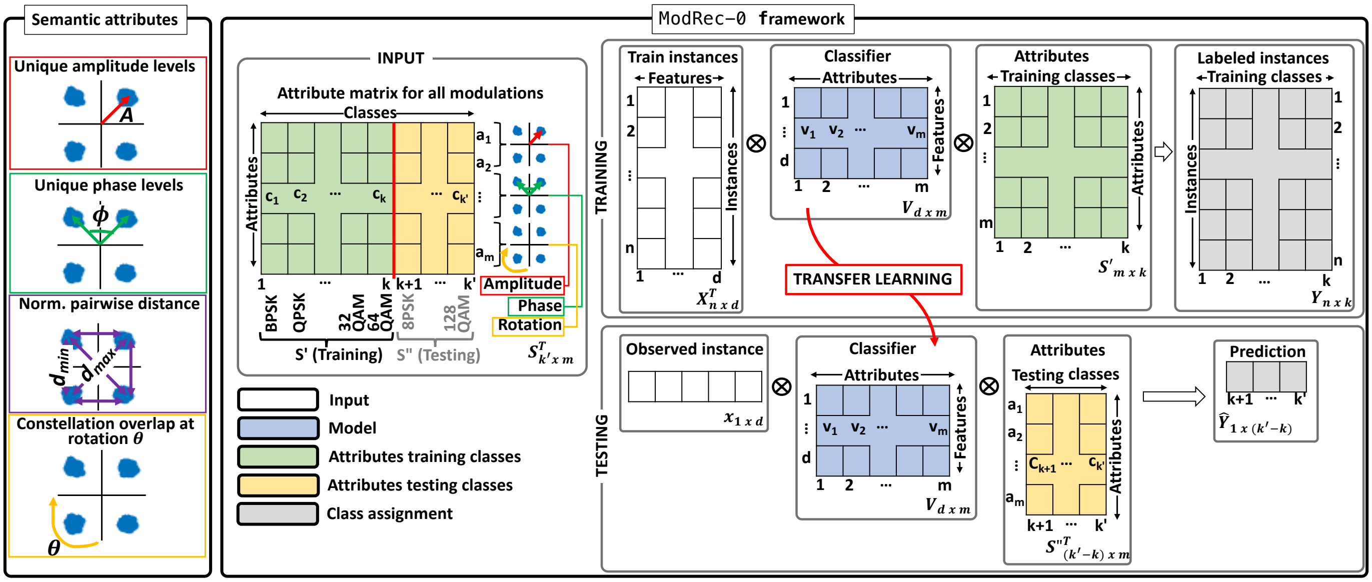


Fig. 1: ModRec-0's zero-shot transfer learning pipeline. The framework takes as an input a matrix of semantic attributes for both *observed* (i.e. training) and *unobserved* (i.e. testing) classes. In the training process, the framework learns a classifier model V, which associates features of the training classes with the semantic attributes for both training and testing classes. V is then transferred to the testing stage for run-time classification of previously *unobserved* modulations.

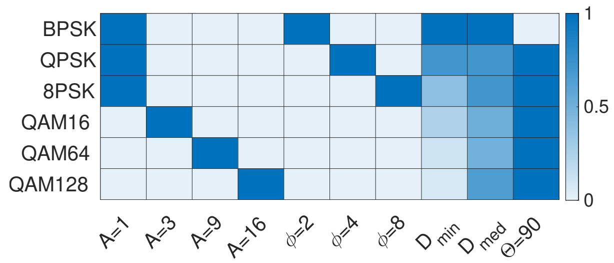


Fig. 2: Semantic attributes for six modulations. Each row indicates an individual class while each column indicates whether a class is characterized by a given attribute.

IV. Evaluation

We now evaluate the performance of ModRec-0 in both realistic simulation as well as on over-the-air traces from [10].

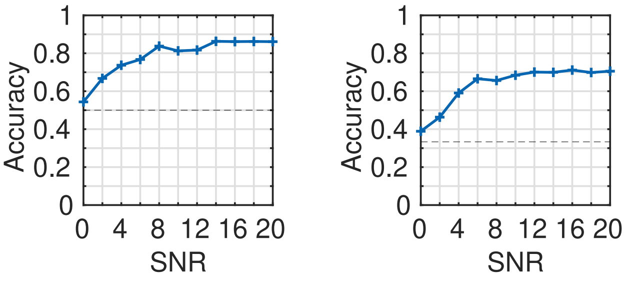
Experimental setup. We implement the zero shot learning framework stemmed from [20]. Our synthetic dataset is generated with the MATLAB Communications Toolbox and includes six modulations: three from the PSK family (BPSK, QPSK, 8-PSK) and three from the QAM family (16-QAM, 64-QAM and 128-QAM). Training and testing instances of each modulation contain 512 samples each. We train on 3000 instances of each modulation and test on 3000. For each instance we compute the features as detailed in §III before performing the classification. We also use an over-the-air dataset from [10], which was recorded in an indoors interference-free environment and features four modulation classes: BPSK, QPSK, 16-QAM and 64-QAM. The transmitter and sensor are positioned in line-of-sight and use USRP B200 SDRs as radio front ends. The dataset contains extracted cumulant features for every class (i.e. no raw IQ samples were provided). We use fourth-order and sixth-order cumulants as the feature representations. In interest of space we omit details in feature extraction and point the interested reader to [19]. For each experiment, we report the average accuracy across all partition combinations. We define accuracy as the ratio of correctly-predicted instances over the total number of test instances.

In the training phase we use a data-driven approach to find the optimal parameters λ, β and γ (see (4)). Specifically, we perform a grid search on these parameters, increasing each in the range $[0.001, 0.01, 0.1, 1, 10, 100, 1000]$. For this parameter estimation, we held out a 40% portion of the training dataset for parameter validation, using accuracy as the validation criteria.

A. ModRec-0 on synthetic data

Our experiments with synthetic data allow for tight control over the SNR, features and the testing/training combinations.

1) Overall accuracy: We begin by presenting results for the overall accuracy of ModRec-0 on synthetic data. We explore two scenarios: 4/2 where training is performed on four classes and the prediction on two and, 3/3 where training takes three classes and prediction takes three classes. We note that the first scenario gains 15 unique train/test combinations, whereas the second scenario gains 20 unique combinations. Fig. 3 presents our results for average accuracy over all training/testing combinations as a function of increasing SNR (0-20dBm in increments of 2dBm). The dashed line indicates a random guess. For both 4/2 and 3/3, the prediction accuracy is well above a random guess. For low SNR regimes, up to 8dBm, the performance suffers deterioration, however, for SNR higher than 8dBm, we are able to maintain high and persistent performance of up to 0.89. Case 4/2 (Fig. 3a) gains superior performance in comparison with case 3/3 (Fig. 3b) across all SNR regimes. Intuitively, the reduced amount of training data in 3/3 compared to 4/2 takes a toll on the classification performance. In the following evaluation, we set forth to understand what training/testing combinations



(a) Train on 4; Predict 2 classes

(b) Train on 3; Predict 3 classes

Fig. 3: Average accuracy over all combinations with increasing SNR.

TABLE I: Accuracy break down for 4/2 (top) and 3/3 (bottom). All cases that do not capture the in-family variation are plagued with poor performance.

#	Training	Testing	2dBm	10dBm	20dBm
1	4/8PSK, 64/128QAM	2PSK, 16QAM	0.98	1.00	1.00
2	4/8PSK, 16/128QAM	2PSK, 64QAM	0.99	1.00	1.00
3	4/8PSK, 16/64QAM	2PSK, 128QAM	0.98	1.00	1.00
4	2/8PSK, 64/128QAM	4PSK, 16QAM	0.78	0.99	1.00
5	2/8PSK, 16/128QAM	4PSK, 64QAM	0.82	1.00	1.00
6	2/8PSK, 16/64QAM	4PSK, 128QAM	0.48	0.79	0.99
7	2/4PSK, 64/128QAM	8PSK, 16QAM	0.60	0.96	0.99
8	2/4PSK, 16/128QAM	8PSK, 64QAM	0.67	1.00	1.00
9	2/4PSK, 16/64QAM	8PSK, 128QAM	0.71	0.95	0.93
10	8PSK, 16/64/128QAM	2/4PSK	0.50	1.00	1.00
11	4PSK, 16/64/128QAM	2/8PSK	0.50	0.50	1.00
12	2PSK, 16/64/128QAM	4/8PSK	0.50	0.50	0.50
13	2/4/8PSK, 128QAM	16/64QAM	0.50	0.50	0.50
14	2/4/8PSK, 64QAM	16/128QAM	0.50	0.50	0.50
15	2/4/8PSK, 16QAM	64/128QAM	0.50	0.50	0.50

#	Training	Testing	2dBm	10dBm	20dBm
1	8PSK, 64/128QAM	2/4PSK, 16QAM	0.51	0.98	1.00
2	8PSK, 16/128QAM	2/4PSK, 64QAM	0.54	1.00	1.00
3	8PSK, 16/64QAM	2/4PSK, 128QAM	0.37	0.89	1.00
4	4PSK, 64/128QAM	2/8PSK, 16QAM	0.45	0.98	1.00
5	4PSK, 16/128QAM	2/8PSK, 64QAM	0.50	1.00	1.00
6	4PSK, 16/64QAM	2/8PSK, 128QAM	0.58	0.67	0.78
7	4/8PSK, 128QAM	2PSK, 16/64QAM	0.65	0.67	0.67
8	4/8PSK, 64QAM	2PSK, 16/128QAM	0.66	0.67	0.67
9	4/8PSK, 16QAM	2PSK, 64/128QAM	0.66	0.67	0.67
10	2PSK, 64/128QAM	4/8PSK, 16QAM	0.37	0.37	0.35
11	2PSK, 16/128QAM	4/8PSK, 64QAM	0.35	0.60	0.65
12	2PSK, 16/64QAM	4/8PSK, 128QAM	0.30	0.65	0.66
13	2/8PSK, 128QAM	4PSK, 16/64QAM	0.50	0.66	0.65
14	2/8PSK, 64QAM	4PSK, 16/128QAM	0.45	0.67	0.67
15	2/8PSK, 16QAM	4PSK, 64/128QAM	0.42	0.67	0.67
16	2/4PSK, 128QAM	8PSK, 16/64QAM	0.33	0.32	0.37
17	2/4PSK, 64QAM	8PSK, 16/128QAM	0.45	0.66	0.67
18	2/4PSK, 16QAM	8PSK, 64/128QAM	0.50	0.58	0.65
19	16/64/128QAM	2/4/8PSK	0.33	0.67	0.67
20	2/4/8PSK	16/64/128QAM	0.33	0.33	0.33

are most susceptible to performance deterioration in order to inform best practices in classifier training for zero-shot modulation recognition.

2) Performance breakdown across combinations: Tbl. I presents the breakdown of classification performance for each training/testing combination. The top table presents results for the 4/2 case, whereas the bottom for the 3/3 case. The accuracy value is averaged across all test classes (i.e. if the test combination was 2-PSK and 16-QAM, the reported accuracy is averaged over the prediction accuracy for each class). We report accuracy for three SNR levels: 2, 10 and 20dBm.

First, we focus on the 4/2 case (Tbl. I top). A random guess in this scenario would gain an accuracy of 0.5. Combinations 1-9 maintain high performance, including for the very challenging SNR regime of 2dBm. Combinations 10-15 gain a random guess (i.e. poor classification performance) across all SNR regimes. To gain more insight, we look at the representation of each modulation family (i.e. PSK and QAM) in the training and testing stages of the algorithm. The main difference between combinations 1-9 and 10-15 is that in 1-9 we train on two classes from each modulation family and test on one class from each family, whereas in combinations 10-15 we train on one class per family and test on two classes of that family. In other words, combinations 1-9 capture the *variance within a modulation family* in training, whereas combinations 10-15 do not capture that variance. Our results for the 3/3 case further explore this issue. We note that none of the training combinations here captures the variance of both modulation families simultaneously. As a result, the average accuracy is driven up by the classification performance of the family for which the variance is captured in training (e.g. cases 1-6 gain high performance, since the classifier is trained on two QAM classes and predicts one QAM class). The performance is susceptible to low-SNR regimes.

The key insight from these observations is that *training in a way that do not capture the in-family variance severely deteriorates the classification performance.* Thus, our recommendation is that *training has to include at least two classes per family, and potentially more for complex modulations*, in support of zero-shot modulation recognition.

3) Effects of the training mix on performance: We further explore the effects of training on ModRec-0, focusing on whether training captures the in-family variance (i.e. within PSK and QAM), cross-family variance (i.e. across PSK and QAM) or both. We attempt to classify two modulations from the QAM family (16-QAM and 64-QAM), while training on three different datasets: (i) one containing labeled data for 4-PSK and 32-QAM, (ii) another containing 4-PSK, 32-QAM and 128-QAM and (iii) a third containing 32-QAM and 128-QAM. In the first case, training captures the cross-family variance between PSK and QAM modulations, but does not capture the in-family variance for QAM. In the second case, we capture both the in-family and the cross-family variance. Finally, in the third case, we only capture the in-family variance for QAM, but not the cross-family variance. Before we delve in our results, we note that we deliberately chose the most challenging classification setting (i.e. predicting high-order QAM modulations), as this case illuminates the benefits of cross-family training. Lower order modulations (i.e. in the PSK family) are successfully classified as long as the in-family variation is captured. Tbl. II presents our results. Combination 1 (training for cross-family variance) gains an accuracy of 0.81. Combination 2, where training captures both the in-family and the cross-family variance, boosts performance to 0.89. Finally, excluding 4-PSK from the training mixture significantly impacts the classification performance, as indicated by the results from combination 3, which are close to a random guess.

The key insight from this experiment is that beyond capturing the in-family variance in training, *incorporating the cross-family variance leads to significant performance*

TABLE II: Effects of the training classes mixture on the classification performance.

	Train	Test	Accuracy
1	4 PSK, 32 QAM	16/64 QAM	0.8047
2	4 PSK, 32/128 QAM	16/64 QAM	0.8845
3	32/128 QAM	16/64 QAM	0.5007

TABLE III: Effects of SNR on classification performance. Training dataset contains {4 PSK, 32/128 QAM}; Testing dataset contains {16/64 QAM}.

Train ↓ / Test →	SNR 2dBm	SNR 10dBm	SNR 20dBm
SNR 2dBm	**0.5403**	0.5000	0.5000
SNR 10dBm	0.5363	**0.5000**	0.5000
SNR 20dBm	0.4972	0.5057	**0.8845**
Mixed SNR	0.5073	0.5000	0.5000

gains, especially in cases where we are looking to classify complex waveforms (e.g. high-order QAM modulations). Our recommendation is that *training should strive to incorporate at least two classes per family from multiple families.*

4) Effects of SNR on performance: The signal-to-noise ratio can significantly impact the constellation shape of a signal and in turn, the classification performance. We are, thus, interested in understanding how does the relationship of training and testing SNR affect classification performance. Ultimately, we seek to inform whether zero-shot modulation classification should be SNR-aware or not. For this experiment, we choose a challenging classification case with a favorable training combination, namely seeking to classify 16-QAM and 64-QAM while training on 4-PSK, 32-QAM and 128-QAM. We set the SNR of the testing and training data to 2, 10 and 20dBm and explore all training/testing combination across these SNR regimes. Tbl. III presents our results. Horizontally, we have our test settings at 2, 10 and 20dBm from left to right. Vertically, we have our training settings at 2, 10, 20dBm and a mixed setting that utilizes samples from all three SNR levels. We train on 3000 samples at each setting. Along the diagonal (highlighted) are all the SNR-aware cases, whereby the testing and the training were performed on the same SNR level. Across all test SNRs, the best performance is gained upon SNR-aware training. In addition, providing a mixed-SNR training dataset does not aid the classification performance.

The key insight from this experiment is that zero-shot modulation recognition is sensitive to the training SNR. We postulate that the transfer of knowledge from one SNR to another is not successful, due to the impact of SNR on the modulation constellation shape. This is particularly relevant for high-order modulation classification. Thus, our recommendation for zero-shot modulation classification is that it should employ channel estimation techniques [25] to pinpoint the SNR of collected signals, prior to attempting their classification. In addition, further exploration of attributes' sensitivity to SNR will inform SNR-blind ModRec with limited supervision.

B. ModRec-0 on over-the-air traces

In this section, we seek to evaluate the performance of zero-shot modulation recognition on real over-the-air traces by employing a dataset from [10]. This dataset was collected with USRP B200 radios across various SNR settings and contains four modulation classes: BPSK, QPSK, 16-QAM and 64-QAM. Due to the smaller number of modulations, we evaluate the 2/2 case, whereby training is performed on two classes and testing on the remaining two classes. As with our synthetic datasets, we derive all possible testing/training combinations (in this case 6) and report accuracy of classification. Fig. 4 presents average accuracy across the six combinations with increasing SNR. The key take-aways agree with the results presented in our synthetic dataset analysis. First, across all SNR levels, ModRec-0 significantly outperforms a random guess. Higher SNR regimes gain better performance, with an average of 0.67 at SNR of 0dBm and 0.82 at SNR of 20dBm.

TABLE IV: Breakdown of performance across combinations for over-the-air traces.

	Training	Testing	0dBm	10dBm	20dBm
1	4PSK, 64QAM	2PSK, 16QAM	0.99	1.00	1.00
2	4PSK, 16QAM	2PSK, 64QAM	0.99	1.00	1.00
3	2PSK, 64QAM	4PSK, 16QAM	0.49	0.68	0.94
4	4PSK, 16QAM	4PSK, 64QAM	0.42	0.86	1.00
5	16/64QAM	2/4PSK	0.53	0.50	0.50
6	2/4PSK	16/64QAM	0.50	0.50	0.50

We further explore breakdown of classification accuracy per case in Tbl. IV. Due to the limited number of modulations, we can only observe the effects of capturing the cross-family variance in training, however, we are not able to explore the effects of in-family variation. Nevertheless, we see that in all cases that train for the cross-family variance (i.e. 1-4), we are able to achieve nearly perfect recognition at 20dBm, which gradually deteriorates as the SNR decreases. We also note that transfer from lower to higher order modulations is more challenging in low-SNR regimes. For example, in combination 3, training on 2-PSK and testing on 4-PSK hampers classification. Similarly, in combination 4, training on 16-QAM while predicting 64-QAM does not transfer adequately. Finally, combinations 5 and 6 train on one family while testing on the other, which leads to a performance commensurate with a random guess.

C. Is expert knowledge beneficial?

As detailed in our methodology, we use expert knowledge to inform the attribute matrix design. In this section, we set forth to evaluate the importance of this expert knowledge in our zero-shot ModRec framework. For this experiment, we randomly permute our original attribute matrix effectively resulting in a random non-informed set of attributes. We evaluate the accuracy drop of ModRec-0 with the random attributes compared to expert attributes. Let A_R be the accuracy with random attributes, and A_E be the accuracy with expert knowledge. The accuracy drop is then defined as $(A_R - A_E)/A_E$. Our evaluation uses the 4/2 training/testing combinations (i.e. Tbl. I top) at 20dBm. Fig. 5 presents our results for each testing/training combination. Each bar reports the average and standard deviation of the accuracy drop across 100 random permutations of the attribute matrix. Across all combinations, we note a substantial drop of performance, between 15 and 50% if expert knowledge was not utilized. This result underpins the importance of expert input in attribute

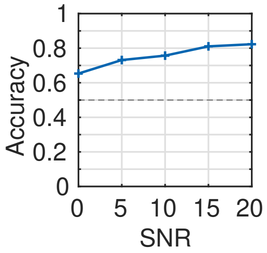


Fig. 4: Average accuracy for over-the-air traces with increasing SNR.

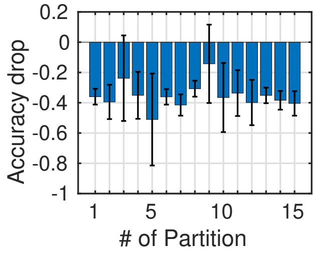


Fig. 5: Accuracy drop with randomly permuted attribute matrix for the 20dBm test cases from Tbl I(top).

design and the potential for performance improvement if more attributes are added to the semantic representation.

V. Discussion and Conclusion

In this paper we explore whether learned classifiers can be transferred to *unobserved* modulations towards modulation recognition of previously *observed* signals. To this end, we designed and evaluated a zero-shot modulation recognition framework called `ModRec-0`. We demonstrated that `ModRec-0` gains high accuracy and is robust across various SNR regimes, indicating the feasibility of real-world classification of previously unobserved modulations. We also explore cases where the classification becomes challenging and draw general insights of how zero-shot frameworks should be employed.

Implications on zero-shot ModRec. Our results show that there are a few considerations to be had. First, the training of zero-shot classifiers should ensure diverse representation of modulation families with at least two observed classes per family. Second, the largest performance gains were observed with SNR-aware testing, suggesting that future zero-shot `ModRec` should incorporate channel estimation prior to classification. Finally, we found that while cross-family transfer is possible from complex to simpler modulations (i.e. QAM to PSK), transfer in the opposite direction is challenging, which mandates that training for complex modulations be performed on high-order members of the same family.

Future directions. While this is the first step towards zero-shot modulation recognition, there are several avenues to be explored to further improve the classification performance. First, feature engineering [6]–[9] could be a promising directions for future exploration. Second, in addition to our defined attributes, we can further employ expert knowledge to expand our semantic representation which may further boost performance [22]. Finally, we can incorporate various other transfer learning techniques [17] in the framework.

VI. Acknowledgements

This work was supported by NSF CISE Research Initiation Initiative (CRII) grant CNS-1657476, NSF Smart and Connected Communities (S&CC) grant CMMI-1831547 and NSF CAREER grant CNS-1845858.

References

[1] O. A. Dobre, A. Abdi, Y. Bar-Ness, and W. Su, "Survey of automatic modulation classification techniques: classical approaches and new trends," *IET Communications*, vol. 1, no. 2, pp. 137–156, 2007.

[2] Z. Zhu and A. K. Nandi, *Automatic modulation classification: principles, algorithms and applications*. John Wiley & Sons, 2014.

[3] M. D. Wong and A. K. Nandi, "Automatic digital modulation recognition using artificial neural network and genetic algorithm," *Signal Processing*, vol. 84, no. 2, pp. 351–365, 2004.

[4] H. Gang, L. Jiandong, and L. Donghua, "Study of modulation recognition based on HOCs and SVM," in *Proc. IEEE VTC 2004-Spring*, vol. 2, pp. 898–902, IEEE, 2004.

[5] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *International conference on engineering applications of neural networks*, pp. 213–226, Springer, 2016.

[6] A. Swami and B. M. Sadler, "Hierarchical digital modulation classification using cumulants," *IEEE Trans. Communications*, vol. 48, no. 3, pp. 416–429, 2000.

[7] L. Han, F. Gao, Z. Li, and O. A. Dobre, "Low complexity automatic modulation classification based on order-statistics," *IEEE Transactions on Wireless Communications*, vol. 16, no. 1, pp. 400–411, 2016.

[8] S. Peng, H. Jiang, H. Wang, H. Alwageed, Y. Zhou, M. M. Sebdani, and Y.-D. Yao, "Modulation classification based on signal constellation diagrams and deep learning," *IEEE Trans. Neural Networks And Learning Systems*, vol. 30, no. 3, pp. 718–727, 2018.

[9] W. Xiong, P. Bogdanov, and M. Zheleva, "Robust and efficient modulation recognition based on local sequential iq features," in *Proc. IEEE INFOCOM*, pp. 1612–1620, 2019.

[10] C. de Vrieze, L. Simic, and P. Mahonen, "The Importance of Being Earnest: Performance of Modulation Classification for Real RF Signals," in *Proc. IEEE DySPAN*, pp. 1–5, 2018.

[11] J. Li, M. Wen, X. Cheng, Y. Yan, S. Song, and M. H. Lee, "Generalized precoding-aided quadrature spatial modulation," *IEEE Trans. Vehicular Technology*, vol. 66, no. 2, pp. 1881–1886, 2017.

[12] B. G. Mobasseri, "Digital modulation classification using constellation shape," *Signal processing*, vol. 80, no. 2, pp. 251–277, 2000.

[13] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *Proc. IEEE CVPR*, pp. 951–958, 2009.

[14] S. J. Hwang, F. Sha, and K. Grauman, "Sharing features between objects and their attributes," in *Proc. IEEE CVPR*, pp. 1761–1768, 2011.

[15] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, "Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly," *IEEE Trans. Pattern Analysis and Machine Intelligence*, pp. 1–1, 2018.

[16] L. Zhang, T. Xiang, and S. Gong, "Learning a deep embedding model for zero-shot learning," in *Proc. IEEE CVPR*, pp. 3010–3019, 2017.

[17] Y. Fu, T. Xiang, Y.-G. Jiang, X. Xue, L. Sigal, and S. Gong, "Recent advances in zero-shot recognition: Toward data-efficient understanding of visual content," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 112–125, 2018.

[18] S. J. Pan, Q. Yang, *et al.*, "A survey on transfer learning," *IEEE Trans. Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[19] M. W. Aslam, Z. Zhu, and A. K. Nandi, "Automatic modulation classification using combination of genetic programming and knn," *IEEE Trans. Wireless Communications*, vol. 11, no. 8, pp. 2742–2750, 2012.

[20] B. Romera-Paredes and P. Torr, "An embarrassingly simple approach to zero-shot learning," in *Proc. ICML*, pp. 2152–2161, 2015.

[21] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid, "Label-embedding for attribute-based classification," in *Proc. IEEE CVPR*, pp. 819–826, 2013.

[22] B. Zhao, Y. Fu, R. Liang, J. Wu, Y. Wang, and Y. Wang, "A large-scale attribute dataset for zero-shot learning," in *Proc. IEEE CVPR Workshops*, pp. 0–0, 2019.

[23] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth, "Describing objects by their attributes," in *Proc. IEEE CVPR*, pp. 1778–1785, 2009.

[24] J. Qin, Y. Wang, L. Liu, J. Chen, and L. Shao, "Beyond semantic attributes: Discrete latent attributes learning for zero-shot recognition.," *IEEE Signal Processing Letters*, vol. 23, no. 11, pp. 1667–1671, 2016.

[25] D. R. Pauluzzi and N. C. Beaulieu, "A comparison of snr estimation techniques for the awgn channel," *IEEE Trans. Communications*, vol. 48, no. 10, pp. 1681–1691, 2000.