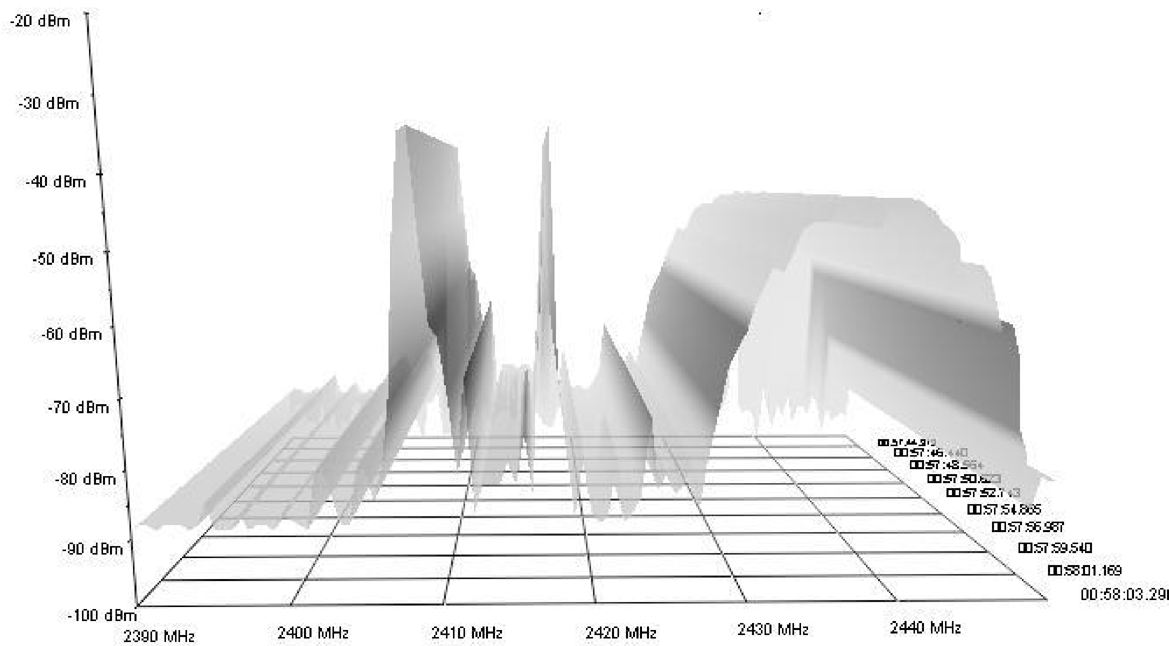
Literature Review & Feasibility Report

Neural Network based Modulation and Channel Coding Identification for SATCOM Systems



Prepared by:

Research Assistant: **Saad Iqbal**, BE
Electrical Engineering

Principal Investigator: **Syed Ali Hassan**, Ph.D.
Electrical Engineering
Department of Electrical Engineering
School of Electrical Engineering and Computer Science (SEECS)
National University of Sciences and Technology (NUST)
Islamabad, Pakistan

Table of Content

Preface

This document is prepared in reference to the “Neural Network based Modulation and Channel Coding Identification for SATCOM Systems” research project. The extensive study has been carried out of the existing literature. Furthermore, the system model is perceived for the evaluation of the project and its findings are integrated in the document as a feasibility report.

The literature review is subdivided into four sections. Sections are inter-linked to ensure the accurate understanding of the reader. *The first section* discusses the classical communication model and provides a complete guide for designing the digital communication system. *The second section* elaborates the modulation and coding schemes with their shortcomings. Moreover, the decomposition of received signals is discussed in detail to recognize the pattern which could lead the receiver side to determine the modulation and coding scheme without any prior knowledge. *The third section* discusses the existing classical models for Automatic Modulation and Coding Recognition (AMR). *The fourth section* completes the literature review by providing the detailed analysis of the existing neural networks which have proved themselves more potent than classical AMR models.

Afterward, *the perceived system model* is elaborated which would be the final outcome of the project. The off-shoot of this section provides the feasibility of the task in hand. It includes the detailed discussion of uncertainties/limitations, research gaps, hardware & software requirements of the project.

The report concludes on the *future work***,** which will elaborate the next stage of the project. This document provides the foundation work for the project and is considered as the primary document to start with.

Introduction

All radio communication signals (radio, television, mobile phones, etc.) are modulated before transmission. Modulation recognition is fundamental for correct demodulation. Moreover, if the transmission is using the forward error correction (FEC) scheme then the prior knowledge of the coding scheme is essential for interpretation of the signal. This is the summary of our quest, *to automatically recognize the modulation and coding scheme* embedded in the received signal.

This has many applications in military, intelligence, and civil communities. For example, Spectrum awareness is crucial in wireless communications systems for dynamic network environments. It is required for spectrum resource management, adaptive transmissions, and interference detection. The key aspects of the spectrum awareness are automatic recognition of *modulation and coding schemes* in digital baseband signals.

This research project [1] aims on developing the state of the art AMR based on Artificial Intelligence (AI). It has four milestone, which are:

- **Literature Review for Classifier Algorithm using Neural Network**
  - ★ A comprehensive survey regarding the AMR schemes and their existing automatic identification methods.
  - ★ Types of Neural Network based networks used for AMR identification.
- **Development and Simulation of Classifier Algorithm using NN**
  - ★ Formulation of Data packet library for MODCOD implementation i.e., M-PSK and M-APSK where M = 2, 4, 8, 16 and 32 with TC / LDPC or Polar Codes at different code rates i.e., $\frac{1}{4}, \frac{1}{3}, \frac{1}{2}, \frac{3}{4}, \frac{5}{7}, \frac{7}{8}$ etc.
  - ★ Preparation of Data sets for training NN.
  - ★ Implementation of classifier algorithm based on NN.

- ★ Simulation of proposed classifier algorithm in supporting software.

- **SDR-based Automatic Modulation Recognition (AMR) system implementation**
  - ★ Implementation of MODCOD Schemes on SDR.
  - ★ Integration of developed classifier algorithm with SDR.
- **Lab prototype for demonstration of proposed model**
  - ★ SDR based data transmission and reception.
  - ★ NN based MODCOD identification using SDR transmissions in a lab environment.

The first milestone is accomplished with the completion of the literature review of Neural Network based AMRs and providing the feasibility of the research project with the detailed discussion of the aspects of perceived system model.

Classical Communication Model

The sole purpose of any communication system is to transmit the payload from one point to another. In the *digital wireless communication system*, the payload is termed as "information" mapped onto binary data. The binary data is translated into electronic signals which travel from transmitter to receiver. At receiver as shown in figure no. 1, the received signal is translated back into binary data and thus information is successfully recovered at destination.

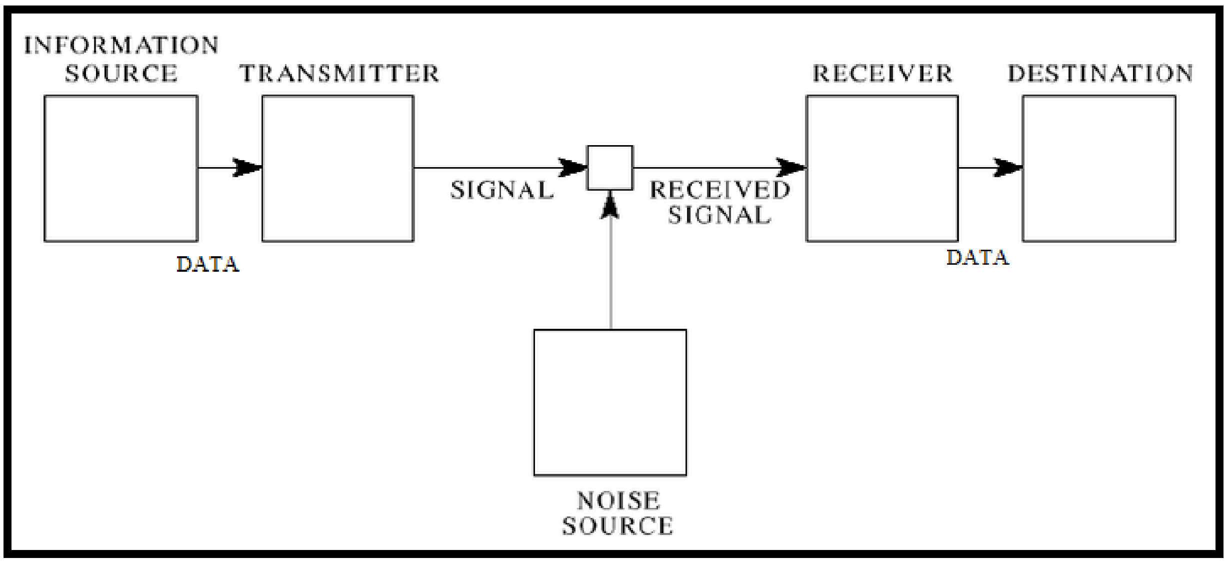


Figure 1: Block Diagram of Simple Communication System

From the transmitter to receiver, the electric signal is influenced by the channel state and deteriorated by the receiver noise. If we normalize the channel state for the moment, the theoretical tightest upper bound on the information rate of received data is:

$C = B(1 + S/N)$ ---------*Shannon–Hartley theorem*[2]

Where,

C is the channel capacity in bits per second

B is the Bandwidth of the channel in hertz

S is the average received signal power over the bandwidth

N is the average power of the noise at receiver

From the above discussion, it is understood that wireless communication has a certain limit in the form of channel capacity. This forces us to efficiently utilize the wireless medium to achieve maximum throughput which is possible via advanced channel coding & modulation scheme. This is the basic principle which governs *information theory*. For better understanding, the above block diagram is further decomposed in figure no. 2.

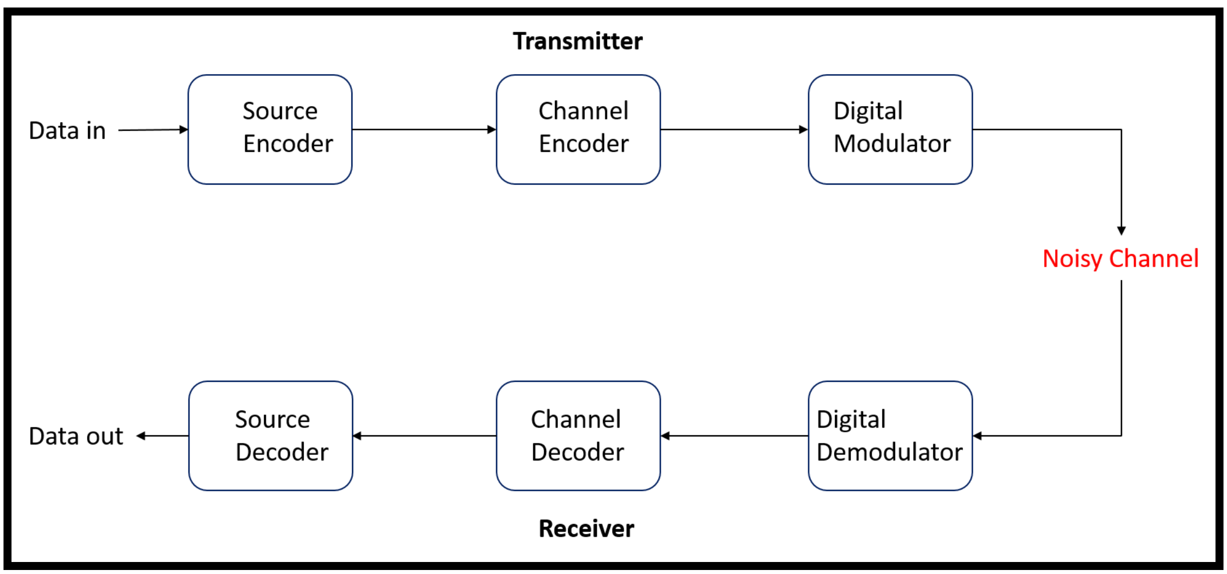


Figure 2: Digital Communication System

The *information/data* could be anything from message, image, video, etc. The data stream is injected into the source encoder**.**

The *source encoder* compresses the raw data into a minimum number of bits. This process helps in effective utilization of the bandwidth. It removes the redundant unnecessary excess bits.

The *channel encoder* does the coding for error correction. During the transmission of the signal, due to the noise in the channel, the signal may get altered and hence to avoid this, the channel encoder adds some redundant bits to the transmitted data. These are the error correcting bits.

The *digital modulator* encodes the digital information into the amplitude, frequency, phase of the baseband signal, which is mixed with the carrier later to travel through wireless medium.

The *communication channel* is the physical medium that is used to send the signal from the transmitter to the receiver. In wireless transmission, the channel would be the atmosphere (free space).

The *digital demodulator* demodulates the received signal into bits. This is the first step to reconstruct the message signal.

The *channel decoder*, after detecting the sequence, does some error corrections. The distortions which might occur during the transmission, are corrected by adding some redundant bits. This addition of bits helps in the complete recovery of the original message.

The *source decoder* accepts the output sequence from the channel decoder and, from knowledge of the source encoding method used, attempts to reconstruct the original message from the source.

Channel Coding & Modulation Scheme

The *channel encoder and digital modulator* blocks are responsible for key functions in the digital communication model. Here, we will discuss the major channel coding types and modulation schemes.

Channel Coding

Channel coding, also known as forward error control coding (FECC), is a process of detecting and correcting bit errors in Digital communication systems. To ensure the data integrity in digital transmission, there are two prominent methods:

1. Forward Error Control Coding (FECC)
2. Automatic Repeat Request (ARQ)

Forward Error Control Coding (FECC) relies on the controlled use of redundancy in the transmitted code word for both the detection and correction of error incurred during the course of transmission over the noisy channel. Irrespective of whether the decoding of the received codeword is successful, no further processing is performed at the receiver. Accordingly, channel coding techniques suitable for Forward Error Correction (FEC), requires only a one-way link between the transmitter and receiver i.e. broadcasting.

There is another approach known as *Automatic-Repeat Request (ARQ)* for solving the error-control problem. The underlying philosophy of ARQ is quite different from that of FEC. Specifically, ARQ uses redundancy merely for the purpose of error detection. Upon the detection of an error in a transmitted code word, the receiver requests a repeat transmission of the corrupted code word, which necessitates the use of a return path (i.e., a feedback channel). Therefore, ARQ can be used only on half-duplex or full-duplex links.

Forward Error Correction (FEC)

The FEC is the one of the fundamental techniques to achieve the theoretical limit of channel capacity prescribed in Shannon–Hartley theorem[2]. We will see in upcoming discussion regarding its role in improving bit-error rate in transmission.

Consider, the binary messages composed by only two symbols namely "0" and "1" are required to be transmitted through medium. If a 0 is sent, it is received as a 0 but it could happen that a 0 will be received as a 1, or vice versa. Thus, for each transmission the channel encoder adds redundancy and transforms the initial vector $u = (u_1, u_2, ..., u_k)$ of length k in a vector $c = (c_1, c_2, ..., c_N)$ of length n where the fraction $r = k/n$ is called the *Code Rate*. After that, c is mapped into modulation symbols x in the next block. There is a probability that the channel introduces errors such that it changes a 0 into a 1 in our transmitted message so an error occurs. The channel decoder receives a vector $y = (y_1, y_2, ..., y_N)$. The purpose of the decoder is to recover the input to the channel encoder from the channel output estimating $\hat{u}$. The probability of error after the decoder is called *Error Probability* (PB).

The Error Correction Codes are sub-categorized into *block codes and convolutional codes*. In a block code, the encoder outputs a code word of length *n* from each length *k* data word whereas in a convolutional code, the encoder outputs one coded stream. The code rate for a convolutional encoder is generally $r = k/n$ as a fraction of inputs *k* and outputs *n*. The prominent channel coding techniques are:

- ★ Turbo Codes
- ★ Low Density Polar Codes
- ★ Polar Codes

Turbo Codes

Turbo codes were presented back in 1993 and are known as the *most successful convolutional codes* which have achieved performance very close to the capacity limit. In its common form, turbo encoding is done using two recursive convolutional encoders. The input stream is passed to the first encoder, and a permuted version is passed to the second one. At the receiving side, two decoders are used, each one decodes the streams of the corresponding encoder. By exchanging probabilistic information, the two decoders can iteratively help each other in a manner similar to a turbo engine[3].

Consider an example case as shown in figure no. 3, a turbo coder consists of one input being the data sequence, and three outputs being the systematic output, output-I from encoder-I, and output-II from encoder-II.

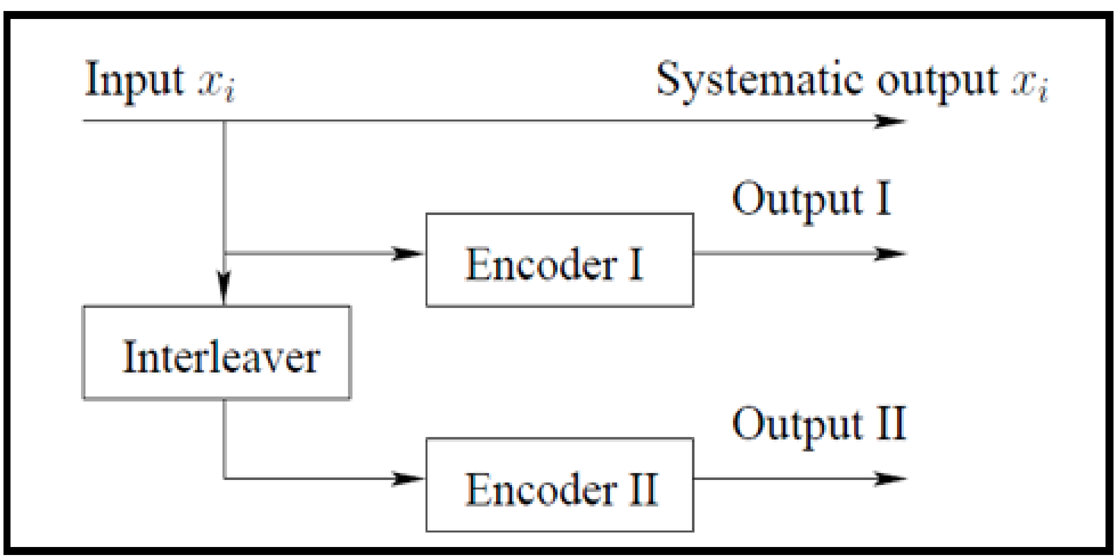


Figure 3: Block Diagram of Turbo Encoder

The input is splitted into two paths. On one path the input arrives at the output in a systematic way. On the other path the input is passed through an interleaver. An interleaver is responsible for scrambling the inputs in a particular fashion, which is termed as a *pseudo-random fashion*. The output from the inter-leaver is then passed through the two encoders. As the data is passed through the encoders, redundancy is introduced, which is the ultimate goal. The natural code rate of a turbo code is $1/3$.

Low Density Parity Codes

Low Density Parity Check (LDPC) was first introduced by Gallager in 1960. It has been shown that these codes have a comparable performance to turbo codes. This performance has resulted in inclusion of LDPC codes in several communication and broadcasting standards[4].

LDPC are *linear error-correcting block codes* also known as sparse parity check matrix, suitable for error correction in large block sizes transmitted via very noisy channels. Consider a parity based code that operates on a block on R bits. Out of the R bits in the block, N bits carry data and M bits carry parity. Thus, code rate would be

$$Coderate = N/M$$

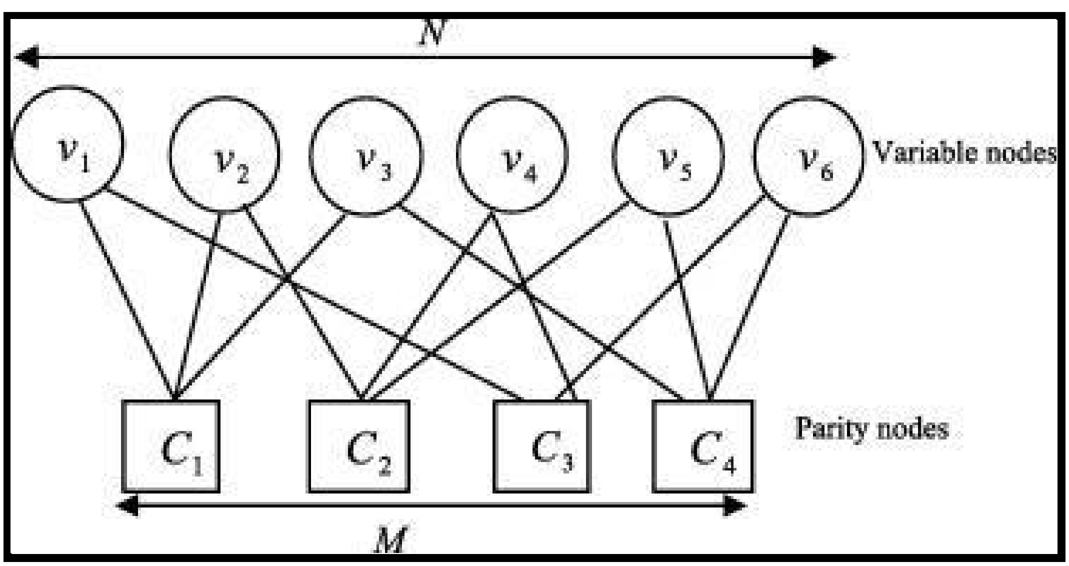


Figure 4: Tanner Graph of LDPC

These codes are represented by a parity check matrix or a Tanner graph as shown in figure no. 4. A *Tanner graph* is a bipartite graph that divides the set of nodes to variable nodes and parity nodes. Each parity node provides the parity of the subset of variable nodes from the block. This mechanism enhances the redundancy of decoding bits in data through the aid of multiple parity bits for each bit.

Polar Codes

A fairly recent type of codes, called polar codes, were introduced by Arıkan in 2008. They are constructed using the channel polarization transform. These codes are error-correcting codes, which are able to achieve the capacity of *binary-input memoryless symmetric (BMS) channels*. This means that one can transmit at the highest possible rate over that class of channels.

The main idea is that by channel combining and splitting, and at infinite length, the channels (bits' positions) will polarize in the sense that some of the channels will be highly reliable, and the rest will be unreliable[3]. The unreliable channels are usually known as frozen sets. The reliable channels will be chosen for information bits and others will be marked as zero. Formally, a specific polar code is fully defined by a 4-tuple (N, R, A, u_A) where:

N is the block length, i.e. the total number of bits transmitted over the channel.

R is the rate, R $\in$ [0, 1], i.e. the amount of information contained in one bit.

A is the information set, A $\subset$ {1,...,N}, i.e. the set of positions which contains the information bits.

$\mathbf{u_{A^c}}$ are the frozen bits, $u_{A^c} \in \{0,1\}^{N(1-R)}$, i.e. bits which have fixed values, shared between the encoder and the decoder.

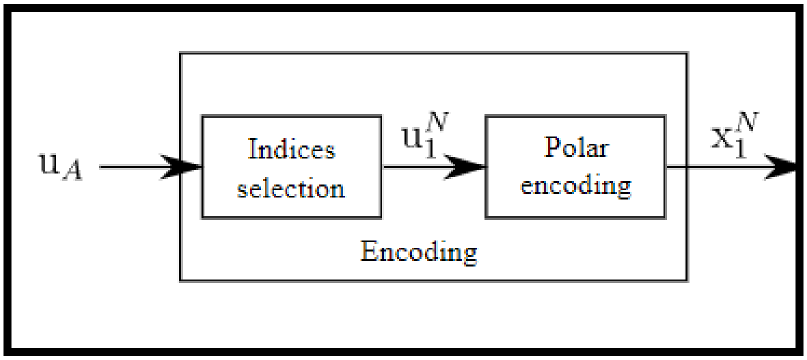


Figure 5: Block Diagram of Polar Code based transmission process

The transmission process as shown in figure no. 5 [6]. The first task is to choose the information set A. Note that the choice of A depends on the particular channel over which transmission takes place, i.e. different channels yield different information sets. We then want to transmit *NR* (information bits) contained in the vector u_A. N(1−R) frozen bits contained in the vector $u_A{}^c$ are fixed. u_{Ac} and u_A are combined to obtain u_1^N. u_1^N is encoded into x_1^N using the polar recursive encoding. This is a fast algorithm which allows to perform in O(N.logN) the linear encoding $x_1^N = u_1^N G_N$, with $G_N = [1\ 0\ 1\ 0]^{\otimes N}$ where ⊗ is the *Kronecker product. The comparison of Turbo, LDPC and polar code* is shown in figure no. 6 [3].

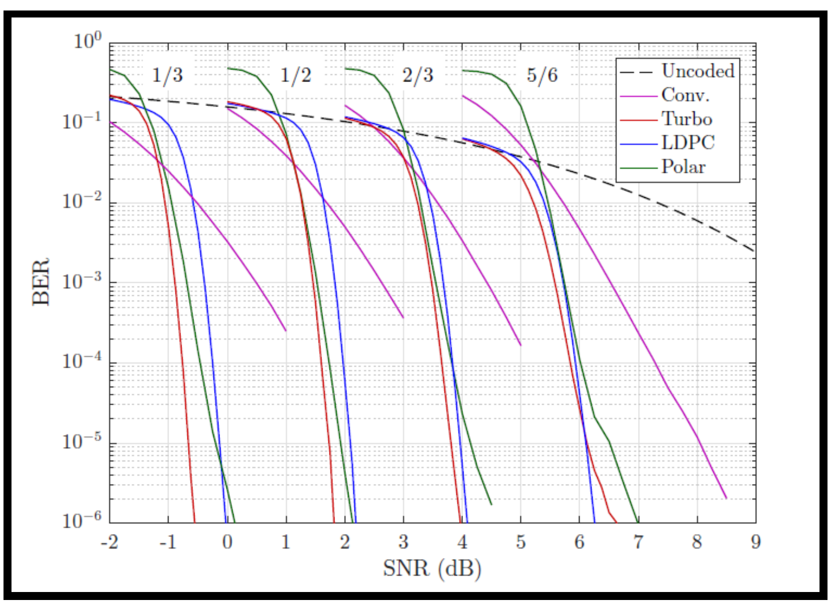


Figure 6: BER comparison of different channel Coding techniques at different code rate

Modulation Scheme

Modulation is a technique for impressing information (voice, music, picture, or data) on a radio-frequency carrier wave by varying one or more characteristics of the wave in accordance with the intelligence signal. The key characteristics of any RF signal are its amplitude, frequency and phase. There are two main techniques used in process of digital modulation/demodulation i.e. Matched filter and correlator elaborated in figure no. 7 and 8 respectively.

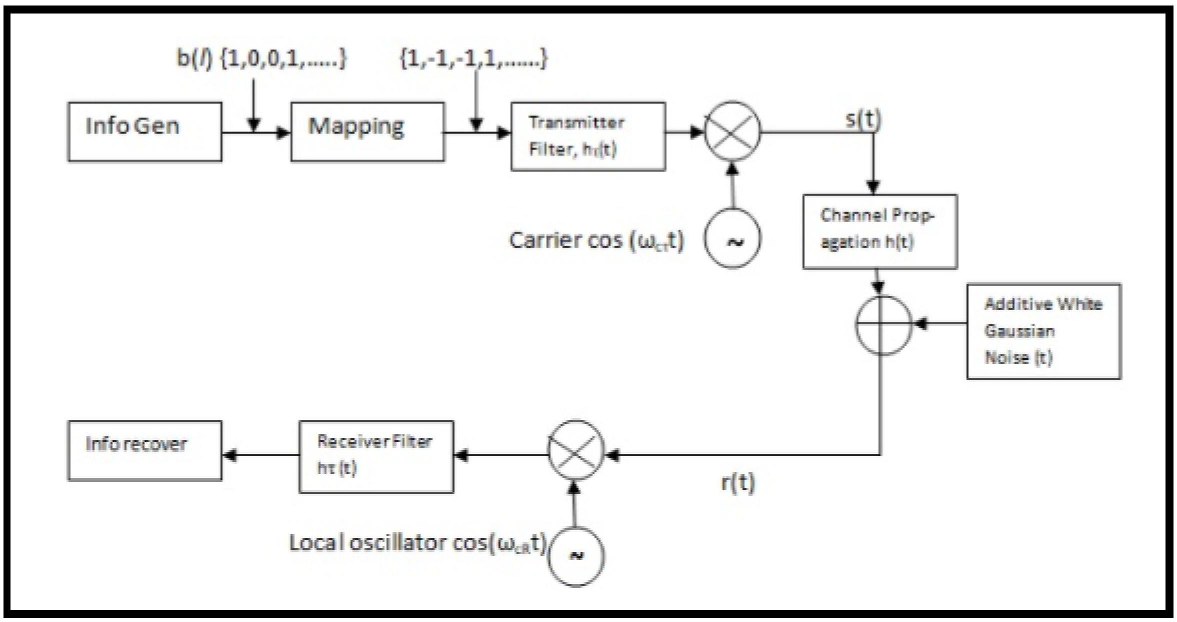


Figure 7: Digital Modulation/Demodulation using Matched Filter Technique

In *matched filter technique* as shown in figure no. 7, If the input signal, *s(t)*, is a wavelet, *w(t)* and *n(t)* is white noise, then the maximum SNR at the output of the receiver will occur when the filter has an impulse response that is the time-reverse of the *w(t)* . As, the convolution of the time-reversed is identical to cross-correlation of the wavelet with the wavelet *(autocorrelation)* in the input signal. Assume, the wavelet is of length, *T*, then the matched filter is defined by:

$$h(t) = w(T - t)$$

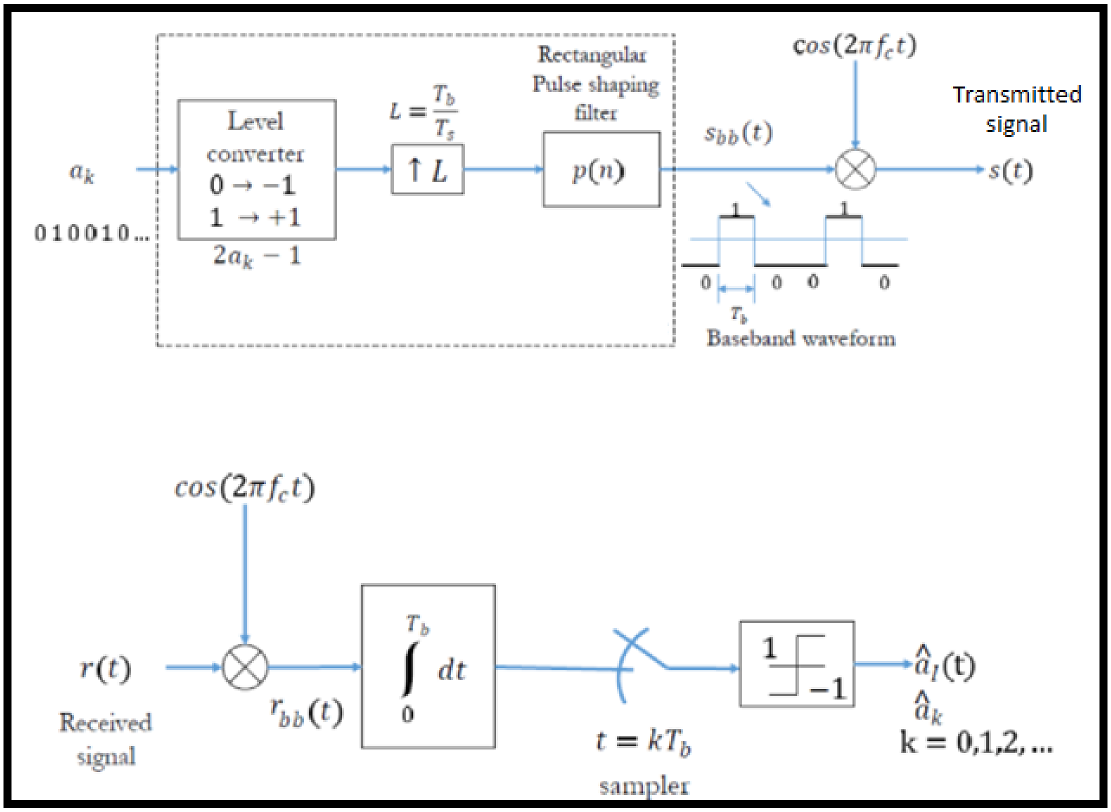


Figure 8: Digital Modulation/Demodulation using Correlator

In *correlator based Digital communication system* as shown in figure no. 8, the received signal *r(t)* is multiplied by a reference frequency signal from the carrier recovery blocks i.e. PLL or Costas loop. The multiplied output is integrated over one bit period using an *integrator*. A threshold detector makes a decision on each integrated bit based on a threshold.

Baseband Signal

The *baseband signal* refers to the waveform before it is mixed with the carrier signal. It contains the message in its wave nature. At the receiver, the matched filter/correlator are used to extract the message bits from the baseband signal. In order to comprehend the baseband signal construction, the constellation diagram plays an important role as shown in figure no. 9.

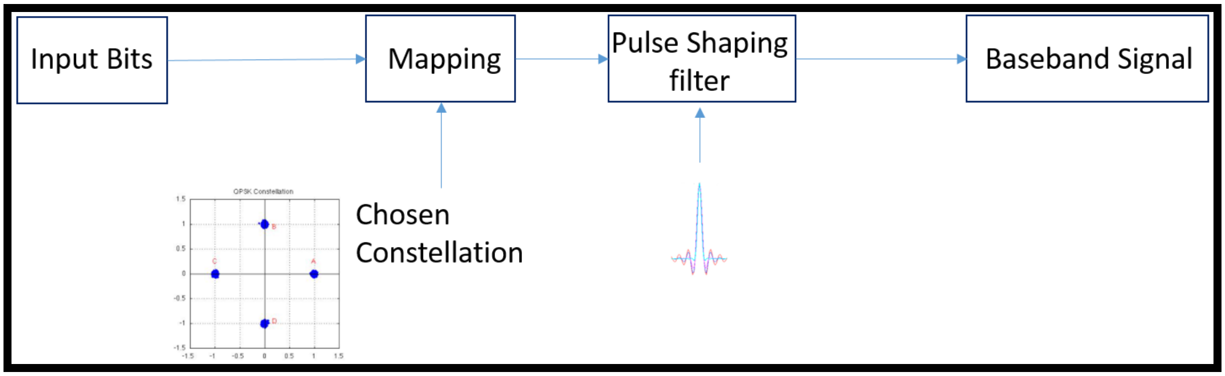


Figure 9: Construction of Baseband Signal

Sl.No.	Modulation formats	Symbols	Information capacity	Derived form	BW efficiency
01	BASK	01	Poor	ASK	Poor
02	BFSK	01	Better than BASK	FSK	Not efficient
03	BPSK	02	2 BFSK	PSK	Only for high speed data transfer
04	DPSK	02	2BFSK	PSK	Only for medium speed communication
05	QPSK	04	2BFSK	PSK	High
06	MSK	04	2BFSK	OQPSK	Lower than QPSK
07	QAM	04	Better than BASK	ASK & PSK	Less than other techniques
08	16 QAM	04	Better than QAM	ASK & PSK	Less than other techniques
09	64 QAM	04	Better than QAM	ASK & PSK	Less than other techniques
10	GMSK	04	Same as QAM	FSK	Excellent

Table 1: Digital Modulation schemes and their properties

At the stage of mapping, the modulation scheme of transmission is realized. There are numerous digital modulation schemes as shown in table no. 1. Every MOD scheme is identified through its constellation Diagram. Few important modulations and their constellation diagram are shown in figure no. 10.

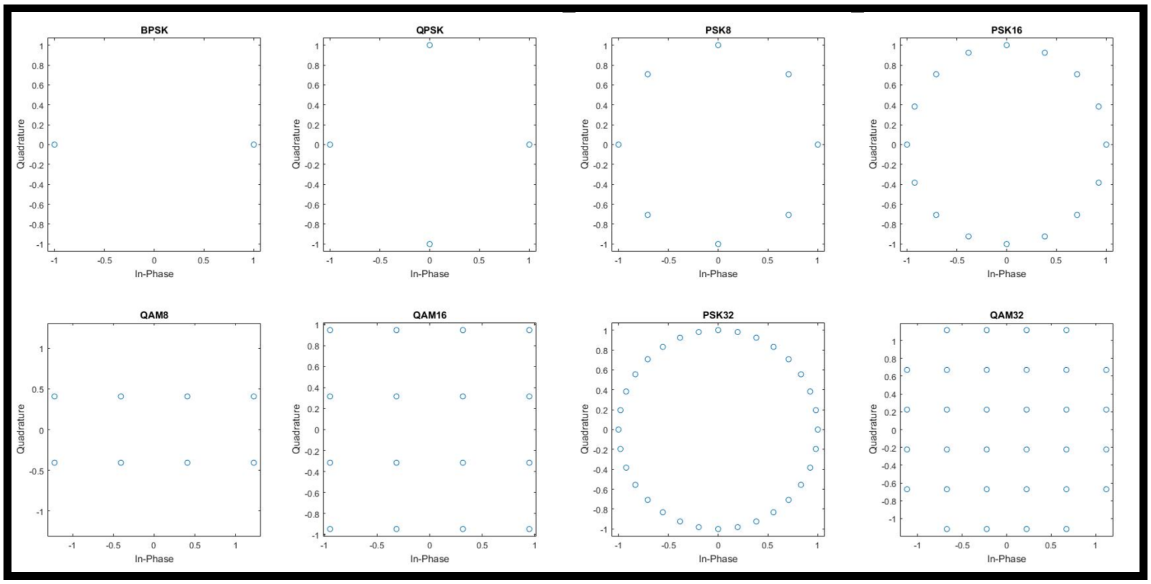


Figure 10: Constellation of Major Modulation Schemes

$Modulation$	$Detection\ method$	$Bit\ error\ rate(P_b)$
$BPSK$	$Coherent$	$0.5erfc(\sqrt{\frac{E_b}{N_0}})$
$QPSK$	$Coherent$	$0.5erfc(\sqrt{\frac{E_b}{N_0}})$
$M-PSK$	$Coherent$	$\frac{1}{m}\ erfc\left(\sqrt{\frac{mE_b}{N_0}}sin(\frac{\pi}{M})\right)$
$M-QAM(m=even)$	$Coherent$	$\frac{2}{m}\left(1-\frac{1}{\sqrt{M}}\right)erfc\left(\sqrt{\frac{3mE_b}{2(M-1)N_0}}\right)$
$D-BPSK$	$Non-coherent$	$0.5e^{-\frac{E_b}{N_0}}$
$D-QPSK$	$Non-coherent$	$Q_1(a,b)-0.5I_0(ab)e^{-0.5(a^2+b^2)}$ $where\ a=\sqrt{\frac{2E_b}{N_0}\left(1-\frac{1}{\sqrt{2}}\right)}$ $b=\sqrt{\frac{2E_b}{N_0}\left(1+\frac{1}{\sqrt{2}}\right)}$ $Q_1(a,b)=$ Marcum Q -function $I_0(ab)=$ Modified Bessel-function

Table 2: Theoretical BER analysis of Digital Modulation schemes

As the modulation order increases, the spectral efficiency would increase on the expense of Bit Error Rate (BER) as shown in table no. 2. The BER comparison of modulation schemes is given in figure no. 11 with AWGN noise at receiver from the book “Digital Communications” by John G.Proakis [5].

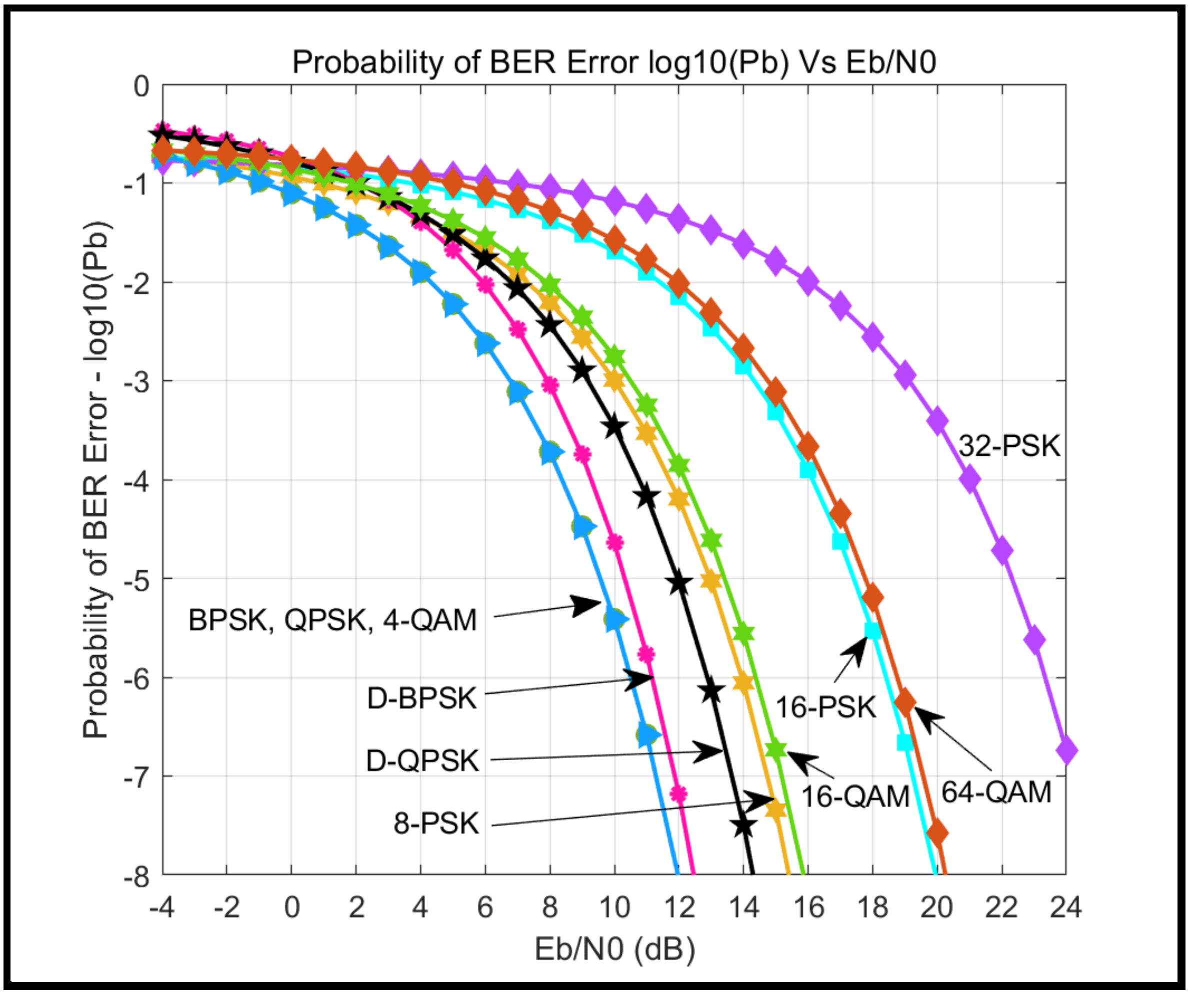


Figure 11 BER comparison of Major Modulation Schemes at different SNR level

Received signal decomposition

The receiver tuned to transmitter frequency would convert the received waveform into baseband signal. This would be the deteriorated copy of the sent baseband signal. The deterioration might occur due to channel impairment, receiver noise, Doppler Effect, etc. A general expression of the received baseband complex envelope would be given as:

$$r(t) = s'(t) + n(t)$$

Where,

$$s'(t) = a_i e^{j2\pi\delta f t} e^{j\theta} \sum_{k=1}^{K} e^{j\theta_k} s_k g(t - (k - 1)T - \sigma T),\ 0 \leq t \leq KT$$

is the noise-free baseband complex envelope of the received signal. a_i is the unknown signal amplitude, f is the carrier frequency offset, θ is the time-invariant carrier frequency introduced by the propagation delay, θ_k is the phase jitter, s_k denotes the vector of complex symbols, T represents the symbol period, σ is the normalized epoch for time offset between the transmitter and signal receiver, $g(t) = Ppulse(t) \otimes h(t)$ is the composite effect of the residual channel with h(t) as the channel impulse response and ⊠ as the convolution.

Now, the demodulator and channel encoder would try to extract the information from it. Now, there's a catch. Can we recover the message without having prior knowledge of coding and modulation schemes? Answer is YES.

It requires an identification of modulation and coding scheme in the received signal before decoding it. There comes the automatic modulation and coding recognition (AMR) algorithms. The conceptual block diagram of the abstract system is shown in figure No. 12.

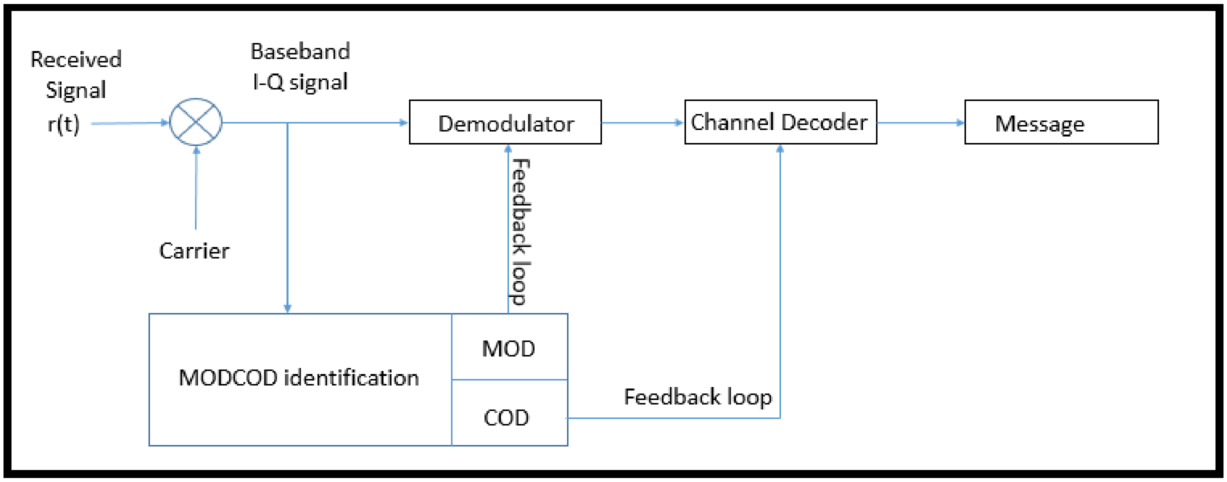


Figure 12: Assistance of AMR in Cognitive Radio

In the next part, we will firstly discuss the prominent features of the modulation & coding schemes which are exploited by the Classical Machine Learning (ML) algorithms to identify the MODCOD scheme. Next, we will shift our discussion to the use of Neural Networks (NN) for solving the situation in hand.

Conventional Models for Modulation & Coding scheme Classification

Early when the Artificial Intelligence models weren't mature enough, the classical approaches were built to develop the Automatic Modulation classifier for the cognitive radio applications. *The classical/statistical approach for automatic coding scheme classification hasn't been tried before in history*. Therefore only modulation recognition schemes would be discussed. There are two general classes of recognition algorithms in the literature, which are as follow:

1. Likelihood-based (LB)
2. Feature-based (FB)

Likelihood-based (LB)

Likelihood-based (LB) modulation classifiers are considered the most popular modulation classification approaches. The utilization of LB classifiers is motivated by the optimality of its classification accuracy when the perfect channel model and channel parameters are known to the classifiers.

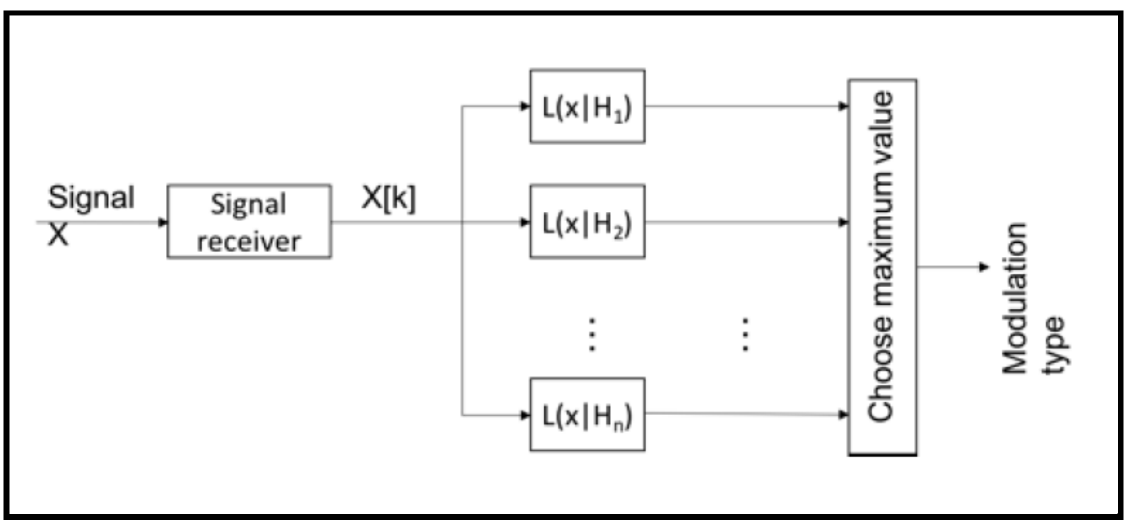


Figure 13: Likelihood-based modulation classification diagram

The methodology of an LB modulation classifier can be inscribed as two steps as shown in figure 13 [7]. In the *first step*, the likelihood is evaluated for each modulation hypothesis with their signal samples. The likelihood functions are derived from the selected signal model. In the *second step*, the likelihood of different modulation hypotheses are compared to conclude the classification decision. Earlier methods of decision making are enabled with a ratio test between two hypotheses. The more intuitive approach of decision making would be to find the *maximum likelihood* among all candidates. It is much easier to implement and does not require carefully designed thresholds.

Types of Likelihood Based Classifiers:

Assume the input signal is *r(t)*, then the system would calculate the likelihood value under each modulation hypothesis *H*. So the likelihood is given by

$$\wedge_A^{(i)}[r(t)] = \int \wedge[r(t)|v_i, H_i]p(v_i|H_i)dv_i$$

Where, $\wedge[r(t)|v_i, H_i]$ is the conditional likelihood function given H_i and unknown vector v_i for the i^{th} modulation scheme, $p(v_i|H_i)$ is the prior probability density function. The estimated modulation algorithm is finally decided by the probability density functions (PDFs). The prominent LB classifiers algorithms are *average likelihood ratio test (ALRT), generalized likelihood ratio test (GLRT) and hybrid likelihood ratio test (HLRT) algorithm.*

The *average likelihood ratio test (ALRT)* algorithm proposed by Kim in 1988 [7], which successfully distinguished between BPSK and QPSK, is the first LB algorithm based on Bayesian theory. The *generalized likelihood ratio test (GLRT)* algorithm proposed by Panagiotou et al [8] and Lay et al [9], which uses maximum likelihood for probability density function and feature estimation outperformed the ALRT in classification. However, it still had some short-comings which were later addressed by *hybrid likelihood ratio test (HLRT)*. HLRT [10] was the combination of ALRT and GLRT, which solved the multidimensional integration problem in ALRT and the nested constellations problem in GLRT by averaging unknown symbol*s*.

Feature-based (FB)

The feature based classifier extracts the key features from the incoming IQ-signal and utilizes them in decision making. A properly designed FB algorithm can show the same performance as the LB algorithm with less computation complexity. Therefore, The FB method usually includes two stages as shown in figure no. 14, which are as follow:

1. Extracting features for data representation
2. The decision making, i.e. classifiers

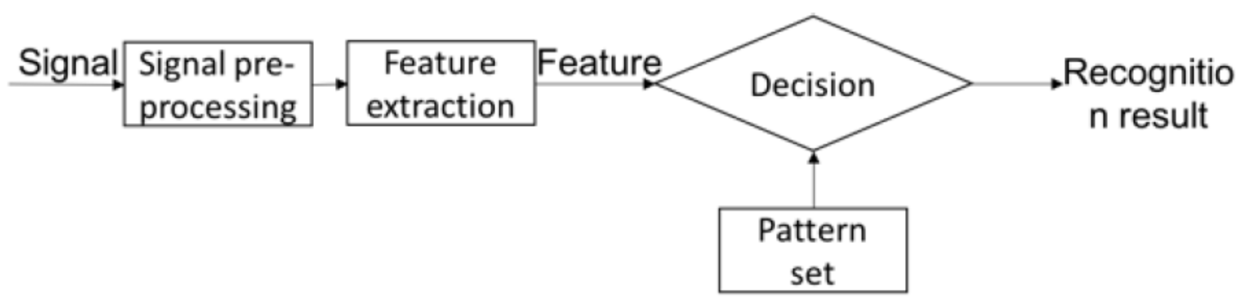


Figure 13: Feature-based modulation classification diagram

Extracting features for data representation

The key features can be categorized as time domain features including instantaneous *amplitude, phase and frequency* [11] [12] [13], transform domain features such as *wavelet transform* or *Fourier transform* of the signals [14]- [15], *higher order moments(HOMs)* and *higher order cumulants(HOCs)* [16]. The *fuzzy logic* [17] and *constellation shape features* [18] [19] are also utilized for AMC.

The instantaneous amplitude of the incoming baseband signal/wavelet is termed as a time domain feature. The well-known features are *peak values, mean, variance and envelope of the wavelet*. The digital modulations based on phase shift key (m-PSK) encode information in the phase of the waveform. Therefore, the pattern of *instantaneous phase variation* in the IQ-waveform has been used as a feature for the feature based AMC in literature.

The wavelet transform is similar to the Fourier transform (or much more to the windowed Fourier transform) with a completely different merit function. *Fourier transform* decomposes the signal into sines and cosines, i.e. the functions localized in Fourier space; in contrast the *wavelet transform* uses functions that are localized in both the real and Fourier space. These transforms of the incoming baseband waveform for the different modulation schemes have a unique pattern to some extent and are therefore utilized in the literature as a feature for the feature based AMC.

The HOM of the intercepted signal x is expressed as,

$$M_{pq} = E[x^{p-q}(x'')^{q}]$$

From these HOMs, a number of higher order cumulants (HOCs) are derived which have been shown to be effective discriminators for many modulation types. HOCs can be computed combinatorially using HOMs, each expression varying slightly i.e. the C40 HOC.

$$C_{40} = M_{40} - 3M_{20}^{2}$$

A constellation diagram is a representation of a signal modulated by a digital modulation. It is unique for every modulation scheme.It displays the signal as a two-dimensional xy-plane scatter diagram in the complex plane at symbol sampling instants. Due to its distinctive nature of every modulation, the constellation matrix of I-Q signal has been utilized as a feature in multiple articles for AMC application.

Decision making/Classifiers

The extracted features are the data representation for the decision making. The classifiers or pattern recognition methods for feature based AMC include artificial neural networks (ANN) [20], unsupervised clustering techniques, SVM [21] and decision tree [22]. Moreover, the KNN, random forest and Reinforcement Learning (RL) are also utilized as classifier and regression techniques.

The artificial neural network (ANN) has succeeded in many research areas and applications such as pattern recognition and signal processing. They are computational models that consist of several processing elements that receive inputs and deliver outputs based on their predefined activation functions. Number of ANN schemes have been developed as a second stage of feature based pattern recognition, including probabilistic NN and K-NN. Single multi-layer perceptrons (MLP) have been used widely. Furthermore, the use of cascaded MLP in ANN is suggested in numerous articles, in which the output of the previous layers are fed into latter layers as input.

An *unsupervised learning* method is a method in which we draw references from datasets consisting of input data without labeled responses. *Clustering* is the task of dividing the population or data points into a number of groups such that data points in the same groups are more similar to other data points in the same group and dissimilar to the data points in other groups. It is basically a collection of objects on the basis of similarity and dissimilarity between them.

On the other hand, *Support Vector Machine (SVM)* is a relatively simple Supervised Machine Learning Algorithm used for classification and/or regression. It is more preferred for classification but is sometimes very useful for regression as well. In SVM, each data item in the dataset is plotted in an N-dimensional space, where N is the number of features/attributes in the data. At the next stage, the optimal hyperplane is formulated to separate the data. Inherently, SVM can only perform binary classification (i.e., choose between two classes). However, there are various practical techniques to use for multi-class problems.

Decision trees are considered as powerful and popular tools for classification and prediction. A Decision tree can be perceived as flowchart like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label as shown in figure no. 14.

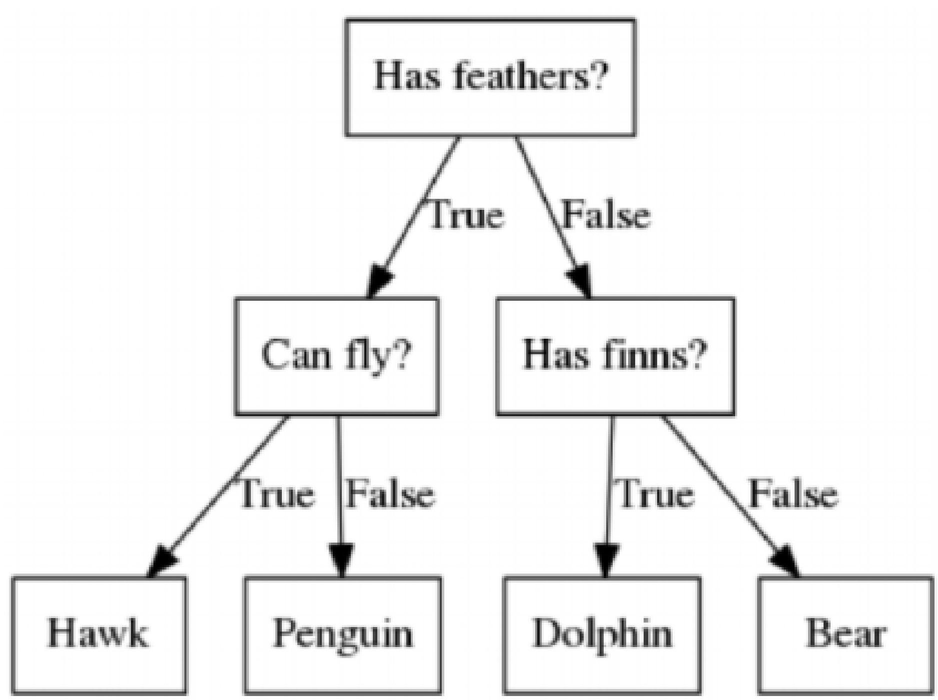


Figure 14: Example of Decision Tree (Courtesy of TowardsAI)

Why Neural Networks?

For many years, universal properties of the nervous system have inspired the modeling of complex and task-driven behaviour for end-to-end solutions [23]. Human nervous system learns the transformations of neural signals to the body system in order to efficiently interact with the environment. Similarly, spectrum sensing applications are being incorporated with neural networks [24] in place of conventional techniques to achieve higher success rate and generalized solution with minimal amount of input data.

Neural Network at its core, is a classifier. It infers its decision on the pre-learned training data just like humans. Their name and structure are inspired by the human brain, mimicking the way that biological neurons signal to one another. The block diagram of artificial NN is shown in figure no.15.

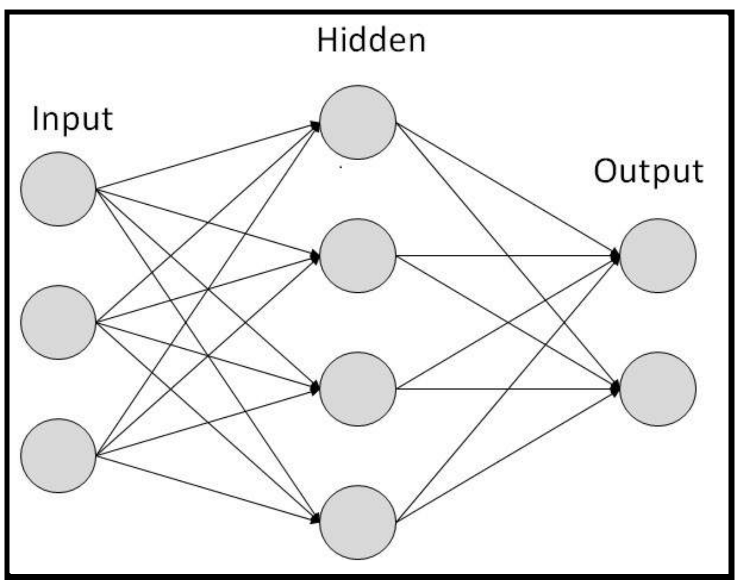


Figure 15: Block Diagram of Artificial Neural Network (NN)

The ANN based classifier outperformed the previous discussed statistical models i.e. Likelihood based (LB) and Feature Based (FB) in terms of computation, accuracy, flexibility and robustness. Following are the few comparisons between the conventional and Artificial NN techniques.

- ★ Given the same input features, *the MLP ANN superseded the decision tree method*[11].
- ★ Unlike LB and FB approaches, where the threshold for decision should be chosen manually, *the threshold in neural networks could be decided automatically and adaptively*.
- ★ Like most decision statistical algorithms presented, the probability of a correct decision on the modulation scheme depends on the sequence of the extracted key features. As can be seen that a different order of key feature application results in different success rates for the modulation type at the same SNR. In the ANN algorithms, all features are considered simultaneously by the model. So that the *probability of the correct decision becomes stable*.
- ★ With the availability of big data and computational resources, the ANN architectures are improved with the unmatchable drift than any existing conventional models.
- ★ While conventional AMRs suffer from noise uncertainty and high computational complexity, neural networks perform well in those situations.

Neural Networks For AMR Application

Artificial Neural Networks (ANNs) mimic the human brain through a set of algorithms. At a basic level, a neural network consists of four main components:

- Inputs
- Weights
- Bias or threshold
- Output

The algebraic formula would look something like this:

$$Y = w_1x_1 + w_2x_2 + w_3x_3 + Bias$$

Let's start with the simplest ANN i.e. *Single Input Neuron* [25] as shown in figure no. 16. The scalar input *p* is multiplied by the scalar weight *w* to form *wp* , one of the terms that is sent to the summer. The other input, 1, is multiplied by a bias *b* and then passed to the summer. The summer output *n*, often referred to as the *net input*, goes into an activation function *f*, which produces the scalar neuron output *a*.

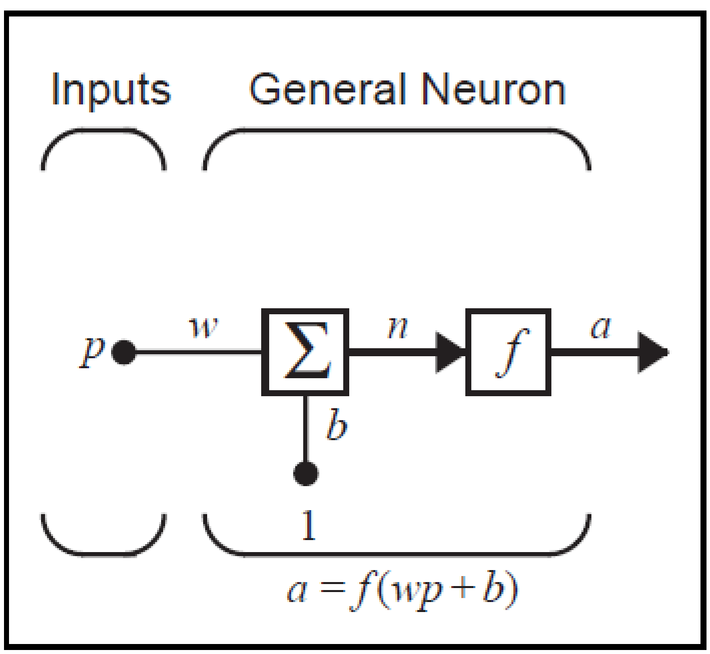


Figure 16: Single Input Neuron (Image courtesy of NN Design by Martin T. Hagan)

The *activation function* can be a linear or nonlinear function. The general activation functions are *Hard Limit, Symmetrical Hard Limit, Linear, Saturating Linear, Symmetric Saturating Linear, Log-Sigmoid, Hyperbolic Tangent Sigmoid, Positive Linear and Competitive*.

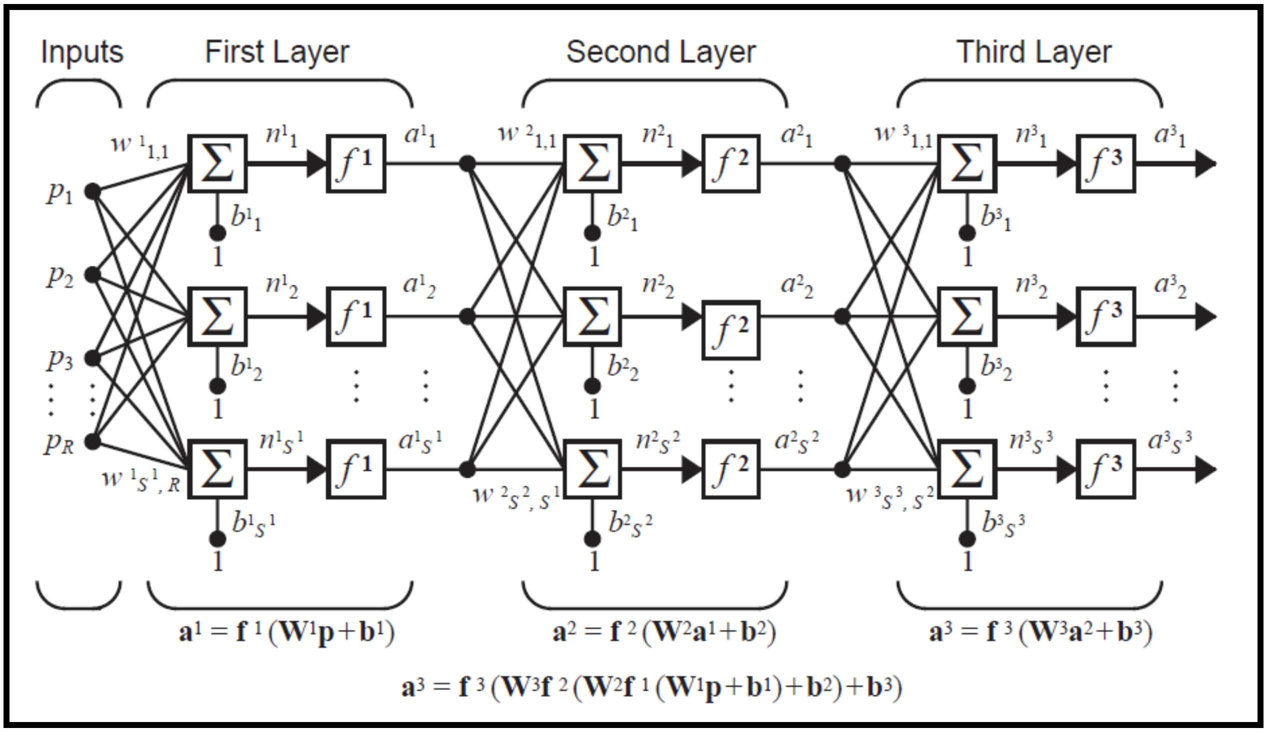


Figure 17: Three Layer ANN (Image courtesy of NN Design by Martin T. Hagan)

As the NN model complexity increases, a single neuron is replaced with the multiple neurons, known as a layer. In the more complex ANN architectures, the multiple layers are introduced in the model. The output from the successive layer becomes the input for the next layer and so on. These are strictly feedforward models with no backward connection. The three layers ANN is illustrated in figure no. 17.

There is another class of networks known as *Recurrent Networks*. A recurrent network is a network with feedback, some of its outputs are connected to its inputs. The building block for RNNs is a *delay and integrator block* as shown in figure no.18. The single Layer RNN is illustrated in figure no. 19 with the *Symmetric Saturating Linear* as activation function .

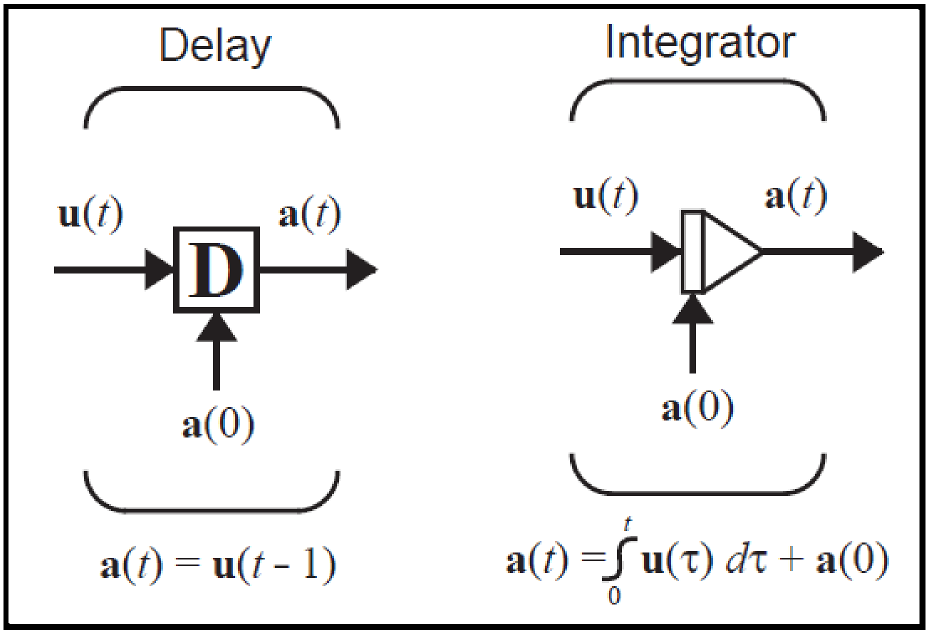


Figure 18: Building Block of RNNs (Image courtesy of NN Design by Martin T. Hagan)

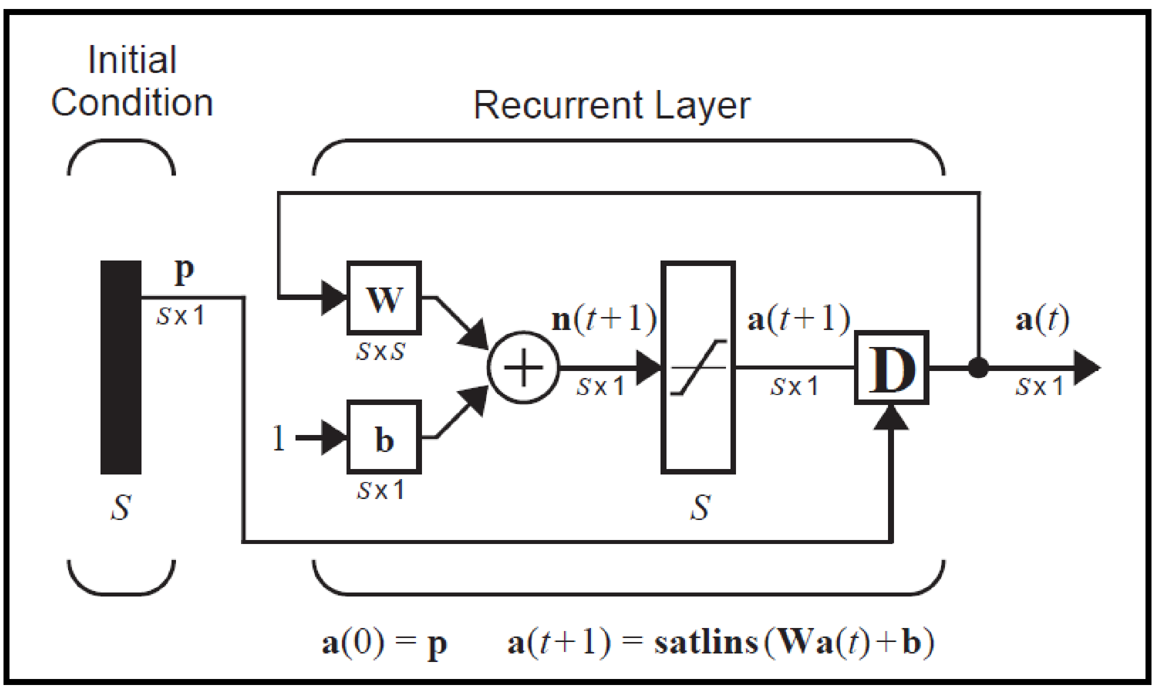


Figure 18: Basic Recurrent Neural Network (Image courtesy of NN Design by Martin T. Hagan)

Types of Neural Networks

Earlier, we discussed the basic architectures of Neural Networks i.e. Feed Forward (FF) and Recurrent Neural Network (RNN). Now, we will discuss the major neural networks which are currently used in the applications of cognitive radios:

- ★ Convolutional Neural Network (CNN)
- ★ Long / Short Term Memory (LSTM)
- ★ Residual Neural Network (ResNet)
- ★ Dense Convolutional Network(DenseNet)
- ★ Convolutional Long Short-Term Memory Deep Neural Network (CLDNN)
- ★ Deep Convolutional Neural Networks (DCNN)

Convolutional Neural Network (CNN)

CNNs are a class of Artificial Neural Networks that can recognize and classify particular features from pictorial data and are widely acceptable for analyzing visual images. Their applications range from *image and video recognition, image classification, medical image analysis, computer vision and natural language processing.*

The basic architecture of the CNN consists of two parts i.e. *convolution tool and fully connected layers*. *A convolution tool* separates and identifies the multiple features of the pictorial input for analysis in a process called Feature Extraction. *A fully connected layer* utilizes the output from the convolution process and predicts the class of the input based on the features extracted in previous stages as shown in figure no. 19. The basic building block of ConvNets is *convolutional Layer*.

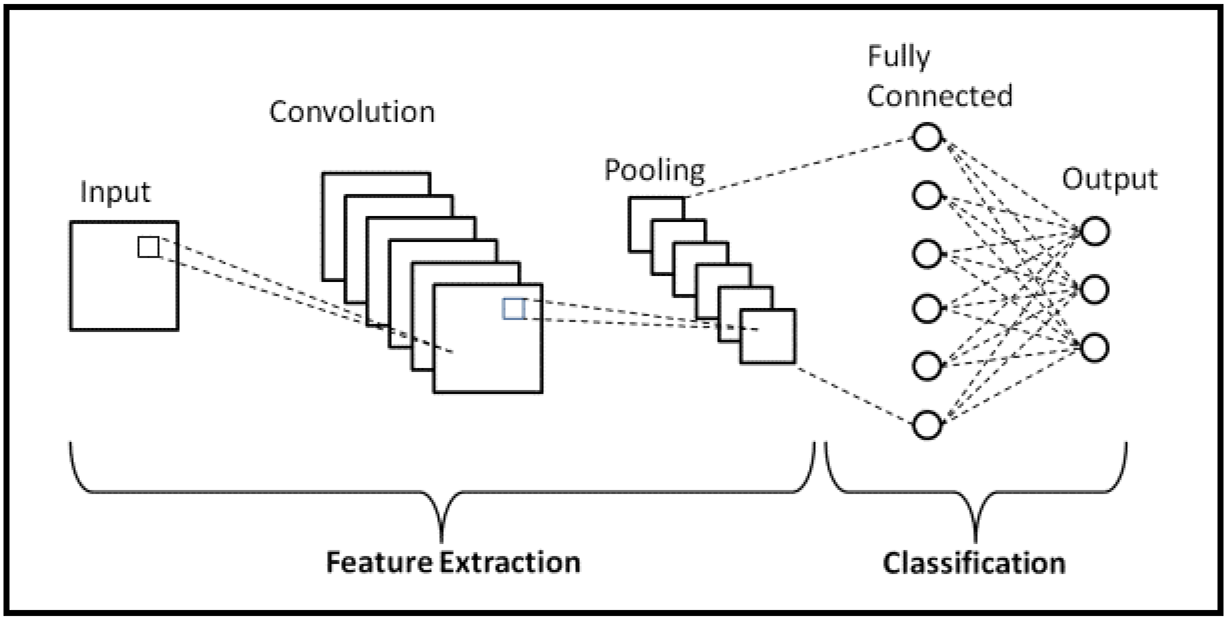


Figure 19: Block Diagram of CNN (Image courtesy of medium.com)

Convolutional Layers:

Convolutional layers are categorized into three main classes i.e. *Convolutional Layers, Pooling Layers and Fully-Connected Layers.* When these layers are stacked, a CNN architecture will be formed. In addition to these three layers, there are two more important parameters which are the *dropout layer and the activation function*.

1. Convolutional Layer

Convolutional Layer is the first layer that performs the extraction of the various features from the pictorial input. In this layer, the convolution is performed between the input image and a filter of a predefined size MxM.

2. Polling Layer

Convolutional Layer is followed by a *Pooling Layer*. The primary aim of this layer is to decrease the size of the convolved feature map to reduce the computational costs. The Max Polling and Average Polling are the common pooling layers. It acts as a bridge between the Convolutional Layer and the FC Layer.

3. Fully Connected Layer

The Fully Connected (FC) layers comprises the weights and biases along with the neurons and is used to connect the neurons between two different layers. At this stage, all the inputs from the successive layer are mapped at the activation unit of the next layer. These layers are placed before the output layer of a CNN Architecture and are responsible for the classification process.

4. Dropout Layer

when all the features are connected to the FC layer, it would cause overfitting in the training dataset. *Overfitting* occurs when a particular model works so well on the training data causing a negative impact in the model's performance when used on new data. To mitigate this problem, a *dropout layer* is utilised wherein a few neurons are dropped from the neural network during training process resulting in reduced size of the model.

5. Activation Function Layer

The activation function layers play the most important role in the CNN architecture. They are used to learn and approximate any kind of continuous and complex relationship between variables of the network. Commonly known activation functions are the ReLU, Softmax, tanH and the Sigmoid functions.

Long / Short Term Memory (LSTM)

LSTM is a recurrent neural network (RNN) architecture that keeps the values over arbitrary intervals. They are well-suited to classify, process and predict time series given time lags of unknown duration. Relative insensitivity to gap length makes the LSTM architecture best suitable over alternatives i.e. general RNNs, hidden Markov models and other sequence learning methods.

LSTMs deal with both *Long Term Memory (LTM) and Short Term Memory (STM)* and it uses the concept of gate for making the calculations simple and effective as shown in figure no. 20. Building block gates of LSTM architecture are categories as:

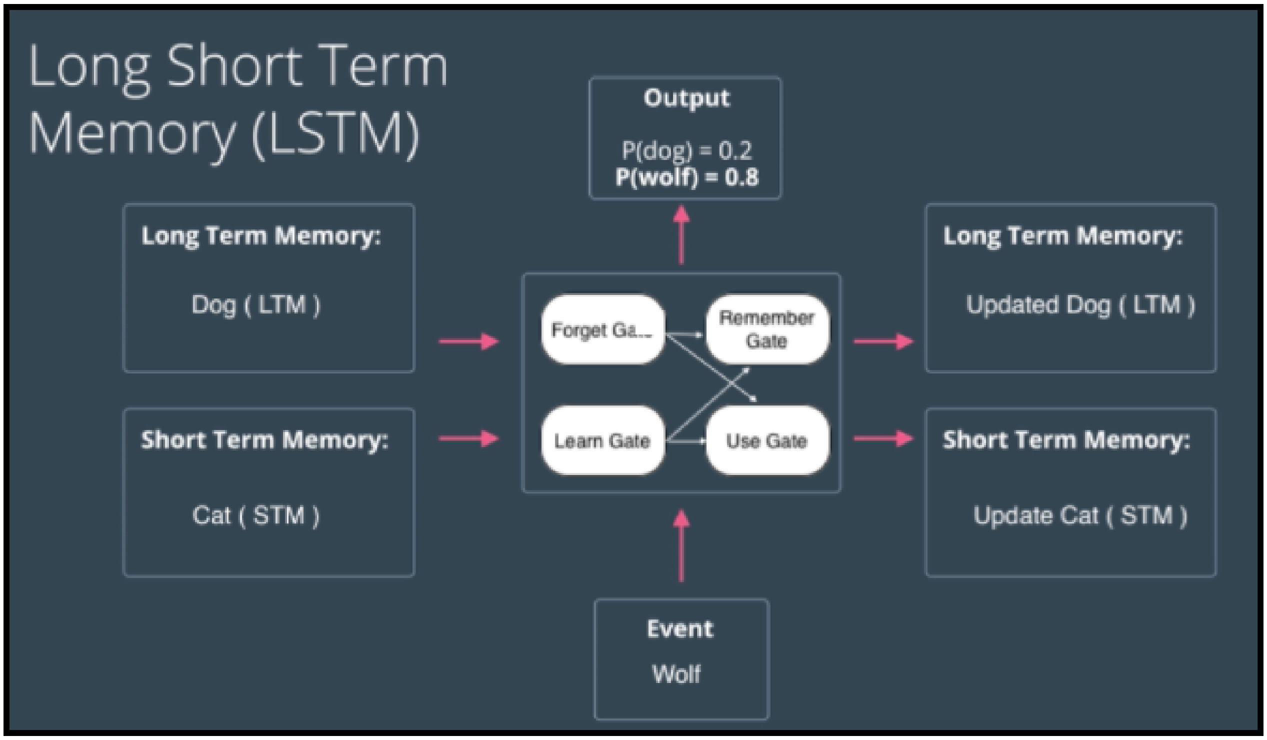


Figure 20: Block diagram of LSTM (Image Source "Udacity")

1. Forget Gate:

It takes Previous Long Term Memory (LTM_{t-1}) as input and decides on which information should be kept and which to forget.

2. Learn Gate:

It takes Event (E_t) and Previous Short Term Memory (STM_{t-1}) as input and keeps only relevant information for prediction.

3. Remember Gate:

Combine Previous Short Term Memory (STM_{t-1}) and Current Event (E_t) to produce output.

4. Use Gate:

Combine important information from Previous LTM and Previous STM to create STM for next and produce output for the current event.

Residual Neural Network (ResNet)

The Residual Neural Network was proposed by researchers at Microsoft Research in 2015. In order to solve the problem of the vanishing/exploding gradient in CNN framework, ResNet introduced the concept called Residual Network.

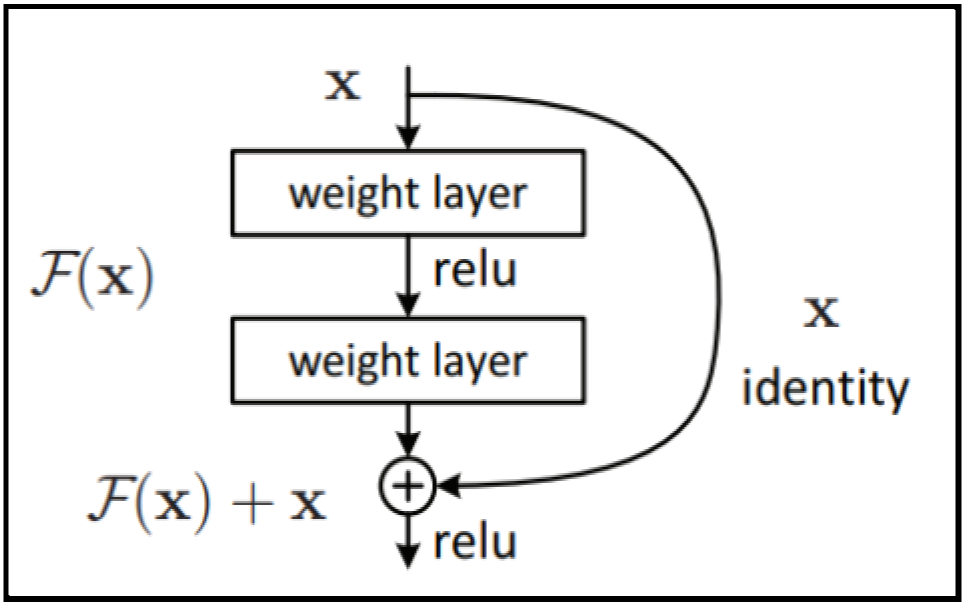


Figure 21: Building Block of ResNet

In this network we use a technique called *skip connections*. The skip connection skips training from a few layers and connects directly to the output. A building block of a residual learning network can be expressed as the function in figure no. 21, where x is the input and $H(x)$ is the output of this block, respectively. Instead of finding the mapping function $H(x) = x$ which is difficult in a deep network, the ResNet adds a shortcut path so that it now learns the residual mapping function $F(x) = H(x) + x$. $F(x)$ is more sensitive to the input than $H(x)$ so the training of deeper networks becomes easier compared to the CNNs.

Dense Convolutional Network(DenseNet)

DenseNet is a type of convolutional neural network (CNN) that utilizes dense connections between layers via Dense Blocks. All layers are connected directly with each other as shown in figure no. 22. The feature output from all the previous layers contributing as input for each layer solves the information blocking problem faced in ResNet and CNNs.

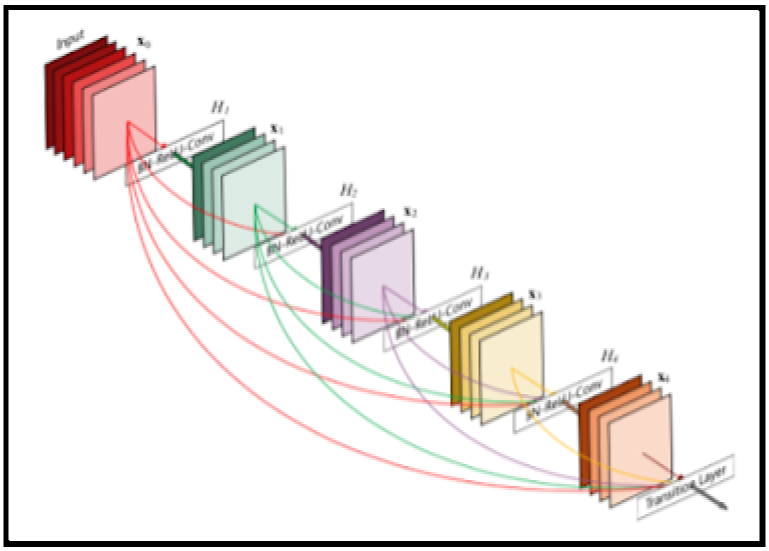


Figure 22: Architecture of DNN (Image source: https://arxiv.org/abs/1608.06993)

Convolutional Long Short-Term Memory Deep Neural Network (CLDNN)

The *Convolutional Long Short-Term Memory Deep Neural Network (CLDNN)* was first introduced by Sainath et al. [26] as an end-to-end model for acoustic learning. It is composed of sequentially connected CNN, LSTM and fully connected neural networks. The raw voice waveform is passed into a CNN, then modeled passed through LSTM and finally resulted in a 3% improvement in accuracy without any pre-processing expense.

The CLDNN are effective for the continuous stream data applications i.e. voice recognition, video filtering and signal processing. At its core, CLDNN are cascading CNNs with LSTMs as shown in figure no.23. It captures both spatial and temporal features, hence proved to have superior performance than all previously discussed architectures.

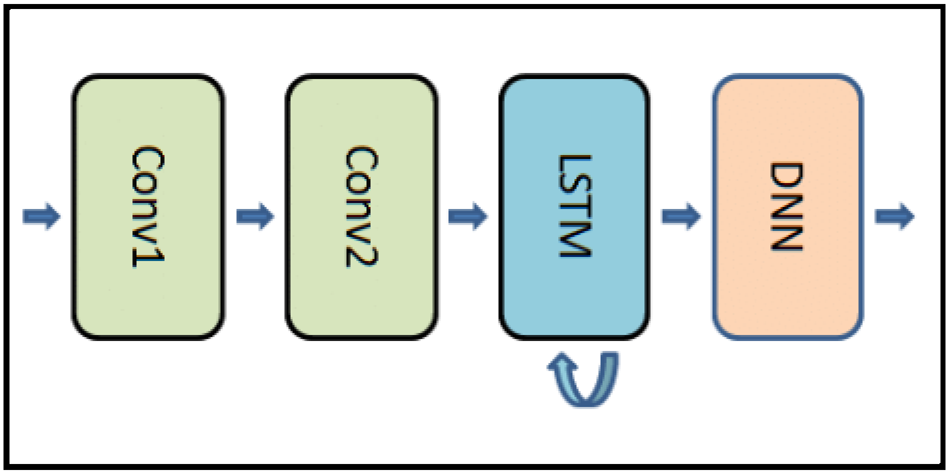


Figure 23: The Architecture of CLDNN

Deep Convolutional Neural Networks (DCNN)

The Deep Convolutional Neural Networks (DCNNs) are developed for image recognition purposes and they use a three-dimensional neural network to process the Red, Green, and Blue elements of the image at the same time. Due to its three-dimensional nature, they require considerably less number of artificial neurons required to process an image, compared to traditional feed forward neural networks.

The architecture of DCNN illustrated in figure no. 24 is similar to CNNs. However, the density of each layer is reduced with the use of a multidimensional approach for feature extraction. It enables the n-dimensional input stream to be mapped on n-dimensional layers enhancing the feature extraction ability with the optimal complexity of the layer.

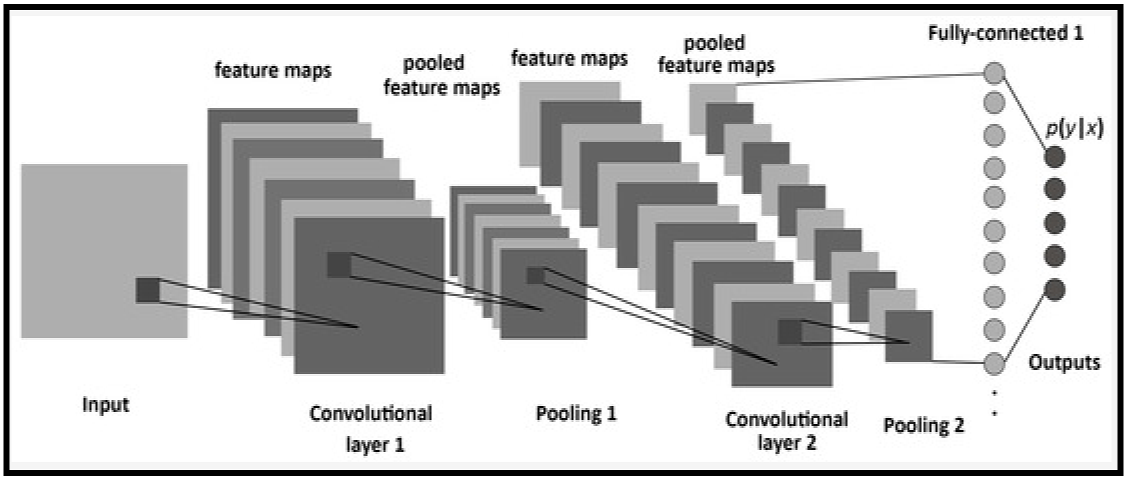


Figure 24: The Architecture of DCNN

Overview on NNs Training methodology

Sehier et al. [27] proposed hierarchical NNs with backpropagation training in 1993. The training of NN has three steps as illustrated in figure no. 25.

1. Preprocessing of the input signal which is different from the first step in *traditional signal processing*. The preprocessing step in ANNs extract key features by itself from an input segment.
2. Training phase learns from these deep features embedded in the input stream and adjusts parameters in classifiers i.e. weights and bias.
3. Finally, the test phase evaluates the model's accuracy using the test data.

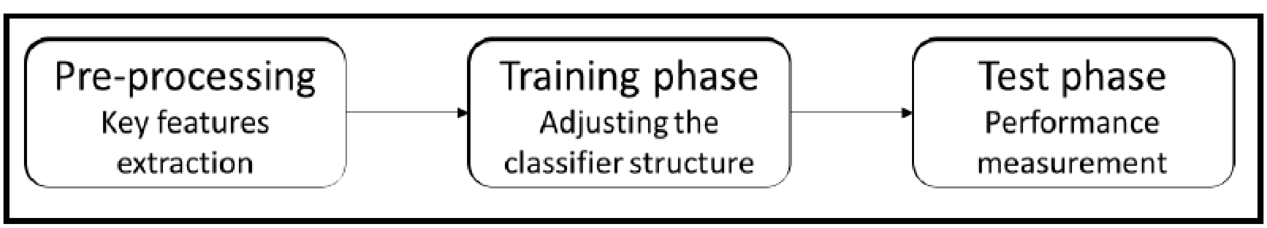


Figure 25: Training sequence for the ANNs

Application of NNs

Nowadays, every existing solution is starting to get challenged by the NNs based alternative approach. But, the input ceritera and decision matrix for each application has its own merits. Therefore, in order to utilize the NNs strength, the different approaches are enlisted in existing articles.

1. Herricical NNs

Hierarchical neural networks consist of multiple neural networks concreted in a form of an acyclic graph. Examples of hierarchical neural networks include Decision Tree, CNN, etc. Major applications of hierarchical classifiers: recognition of 3D objects, optical character recognition, gesture recognition.

2. Cascaded NNs

Cascaded neural networks consist of two or more networks in tandem such that the output of the first NN feeds the input of the second NN. Examples of Cascaded NNs include CLDNN and any network in which multiple NNs are cascaded together. This has application in classification of the input stream in multiple output domain i.e. "set1 = {car, buse,van}, set2 = {KIA car, TOYOTA car, SUZUKI car}".

3. Stacked NNs

Cascaded neural networks consist of two or more networks stacked on each other such that they extract features independent from each other. Examples of Stacked NNs include DCNN and any network in which multiple NNs are stacked together for enhancing the feature extraction in parallel fashion.

4. End-to-End NNs

End-to-end (E2E) NNs refers to a possibly complex learning system represented by a single model (specifically a Deep Neural Network) that represents the complete target system, bypassing the intermediate layers usually described in traditional pipeline designs.

Existing NNs based AMRs

Uptill now, we have discussed the major Neural Networks (NNs) architecture and studied their recipes for different application purposes i.e. Herricical, cascaded, stacked and end-to-end. The next section is categorized into two sub-parts i.e. *NNs for modulation schemes classification* and *NNs for coding schemes classification* to elaborate the NNs contribution in AMRs.

NNs for Modulation Schemes

Automatic modulation classification (AMC) is an essential component of several intelligent communication systems. J. Jagannath et al [28], proposed the ANN based AMC improved by introducing Nesterov accelerated adaptive moment in 2018 for SDRs which outperformed the existing Hybrid Hierarchical AMCs (HH-AMC) as shown in figure no.26. The [28], utilized the instantaneous amplitude for SNR estimation and the features were "$var(\gamma_{max})$ and $E(\gamma_{max})$" where γ_{max} is the measure of deviation of the PSD of the signal from its average value.

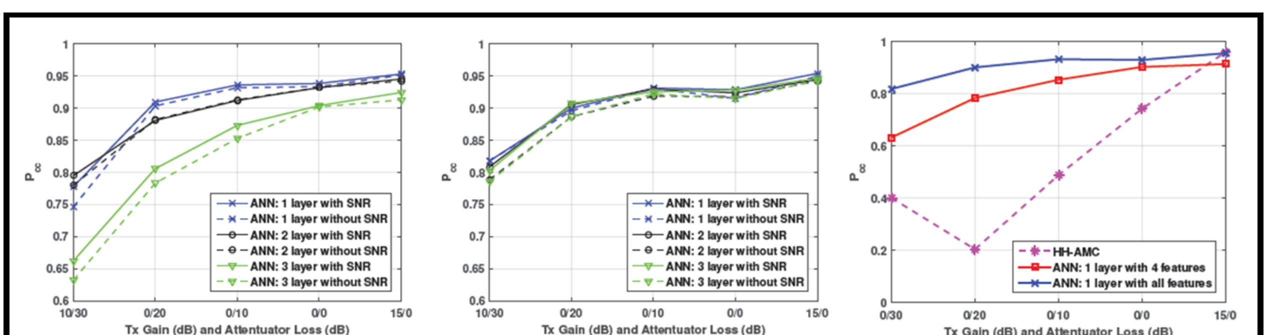


Fig. 4: P_{cc} vs SNR (without $Var(f)$) Fig. 5: P_{cc} vs SNR (with all features) Fig. 6: P_{cc} vs SNR

Figure 26: Comparison of HH-AMC and AMC proposed in [28]

S. Peng et al [29], proposed the Modulation Classification Based on Signal Constellation using DCNN. [29] utilized the existing image NN based classifier i.e. AlexNet and GoogLeNet for the modulation classification. The time-series data is translated into the constellation diagram and used as the input for the classifier. The accuracy results are shown in figure no. 27

TABLE I
CLASSIFICATION RESULTS OF FOUR IMAGE TYPES FOR 8PSK WITH SNR = 2 dB AND ALEXNET

IMAGE TYPES	BPSK	4ASK	QPSK	OQPSK	8PSK	16QAM	32QAM	64QAM	ACCURACY
Constellation Diagram	0	0	5	5	828	22	140	0	82.8%
Gray Image	0	0	6	8	888	16	80	2	88.8%
Enhanced Gray Image	0	0	2	3	965	1	29	0	96.5%
3-Channel Image	0	0	0	1	971	5	23	0	97.1%

Figure 27: The classification results from [29]

P. Wu et al [30], proposed network for AMC based on SDRs combine the advantages of Inception network and ResNet, which have faster convergence rate and larger receptive field. It peaks at 93.76% when the SNR is 14 dB, which is 6 percent higher than that of LSTM and 13 percent higher than that of MentorNet, Inception, and ResNet purely.

N. Daldal et al [31], utilized the time-frequency information (STFT) as a feature for AMC using CNN as classifier. The classifier results for m-PSK modulation scheme are given in figure No. 28.

N. Daldal, Z. Cömert and K. Polat / Applied Soft Computing Journal xxx (xxxx) xxx

Table 6
Confusion matrix of classification result with included all SNR rates in classes.

		Actual labels					
		ASK	**FSK**	**PSK**	**QASK**	**QFSK**	**QPSK**
Predicted labels	**ASK**	1527	21	0	1	0	0
	FSK	3	1509	0	0	0	0
	PSK	0	0	1530	0	0	0
	QASK	0	0	0	1526	1	0
	QFSK	0	0	0	1	1505	3
	QPSK	0	0	0	2	24	1527

Figure 28: Classifier performance results [30]

Z. Zhang et al [32], proposed the AMC based on CNN architecture with more than 8-features including smooth pseudo wigner-ville distribution (SPWVD) and Born-Jordan distribution (BJD). The overall classification accuracy is more than 96%, and some of them even achieve 100% at 0dB. However, the further classification of the QAM modulation scheme wasn't mentioned.

T. J. O'Shea et al [33], provided the analysis on deep learning approaches for radio signal classification for radio communications. The Baseline, CNN and ResNet were tested on more than eighteen modulation schemes in simulation and OTA environments. The RN approach achieves state-of-the-art modulation classification performance on a difficult new signal database both synthetically and in OTA among the others.

M. Kulin et al [34], proposed the End-to-End DCNN approach for modulation classification. The IQ, Amplitude/phase and FFT data were selected as incoming signals. Specifically, in the modulation recognition case study for medium-high SNR the CNN model trained on amplitude/phase representations outperformed the other two models with a 2% and 10% performance improvement, while for low SNR conditions the model trained on IQ data representations showed best performance. For the task of detecting interference, the model trained on FFT data outperformed amplitude/phase and IQ data representation models by up to 20% for low SNR conditions, while for medium high SNR up to 5% classification accuracy improvements.

Comparison

As, we have already seen that the multiple Neural Networks techniques have been used to correctly recognize the modulation scheme. The comparison of popular NNs under AMC is given in table no. 3

Algorithm	**Input**	**Modulation**	**ROC Performance**
SVM [35]	Extracted Features	BPSK, QPSK, 8PSK, 4QAM, 16QAM and 64QAM	90% at 10 dB 70% at 0 dB
K-NN [36]	Extracted Features	BPSK, QPSK, 8PSK, 4QAM, 16QAM and 64QAM	98% at 10 dB 80% at 0 dB
CNN [37]	Extracted Features	BPSK, QPSK, 8PSK, 16QAM, 64QAM, BFSK, CPFSK, PAM4, WB-FM, AM-SSB and AM	99% at 10 dB 84% at 0 dB
RNN [38]	Extracted Features	BPSK, QPSK, 8PSK, 16QAM, 64QAM, BFSK, CPFSK, PAM4, WB-FM, AM-SSB and AM	92% at 10 dB 84% at 0 dB 60% at -5 dB
DL [39]	End-toEnd system	BPSK, QPSK, 8PSK, 16QAM, 64QAM, BFSK, CPFSK, PAM4, WB-FM, AM-SSB, and AM-DSB	90% at 10 dB 85% at 0 dB 70% at -5 dB

Table 3: Comparison of existing NN architectures in AMR

NNs for Channel Coding Schemes

Fan Mei at al [40], has introduced the Blind Recognition of Forward Error Correction Codes embedded in BPSK modulation based on Recurrent Neural Network (RNN). The selected FEC were Linear Block codes, convolutional codes, LDPC codes, BCH codes and Turbo codes.

The experimental setup [40], simulated 1 million samples, each of which is 10,000 in length and stored in int32. The SNR of the sample is evenly distributed between -10 dB and 10 dB. The 70% of data set used as the training channel coding type recognition model and the data set containing 30% of the samples was used as the performance evaluation of the model. In the training process, the learning rate was set as 0.002, if training loss has not reduced after every 10 epochs, set the study rate as 10% of the primary. Training network using batch size as 16 to train 200 epochs. The RNN model accuracy results are shown in figure no. 29.

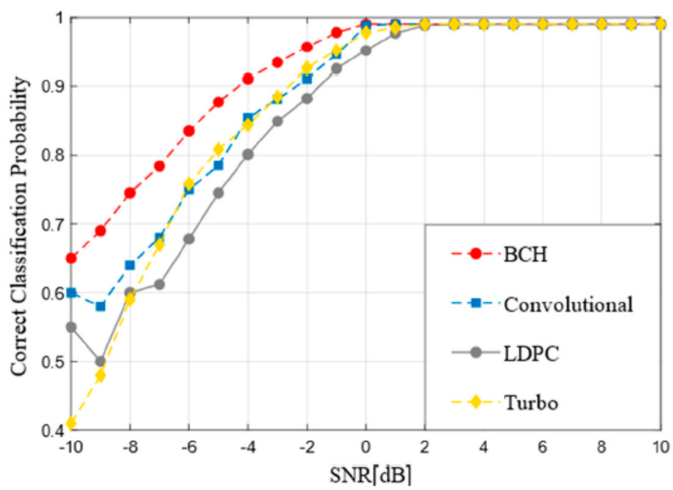


Figure 4. Recognition performance of the improved RNN model for each code type.

Figure 29: FEC Recognition performance of the RNN model proposed in [40]

The existing literature for coding schemes is very limited . Mostly authors have used the simplest modulation scheme i.e. BPSK in the data generation. This enhanced the complexity of our research project because there's a huge research gap in the development of "Automatic Coding scheme recognition".

Our Approach

While we were conducting the literature review, we started working on the perceived system design for elimination of major uncertainties in the project i.e. hardware inaccessibility, Software/computational limitations, etc. Last week the replica of the final demonstration was conducted successfully as shown in figure no. 30. This is the exact test-bed, which would also be used later for gathering of training data for our NN model. The hardware utilization report is as follow:

S.No	Items	Description	Quantity
1.	Software Defined Radio	USRP N200 Kit (USRP N200, 2 SMA-Bulkhead RF Cables, Ethernet Cable, Power)	02
2.	RF Daughter Board	CBX USRP DAUGHTERBOARD (1.2 -6GHZ)	02
3.	MIMO Cable	Cable Assembly, USRP MIMO Data and Sync Cable, 0.5M	01
4.	Gigabit Switch	16-Port Unmanaged Gigabit Switch DGS-1016A	01
5.	Network Cables	1 meter CAT-6 ethernet Cable	03
6.	RF cables	2 meter RF Cable with SMA type connector	02
7.	RF Antennas	VERT2450 Vertical Antenna (2.4-2.5 and 4.9-5.9 GHz) Dualband	02
8.	Power Extension Board	Xiaomi Mi Power Strip 3 Power Sockets	01
9.	Surveillance Source	Logitech C310 HD WEBCAM	01
10.	GPU Machine	General Purpose Computing Machine located at IPT Lab	01
11.	Laptop	Dell Inspiron 15 5593 Core i7 10th Generation Laptop	01

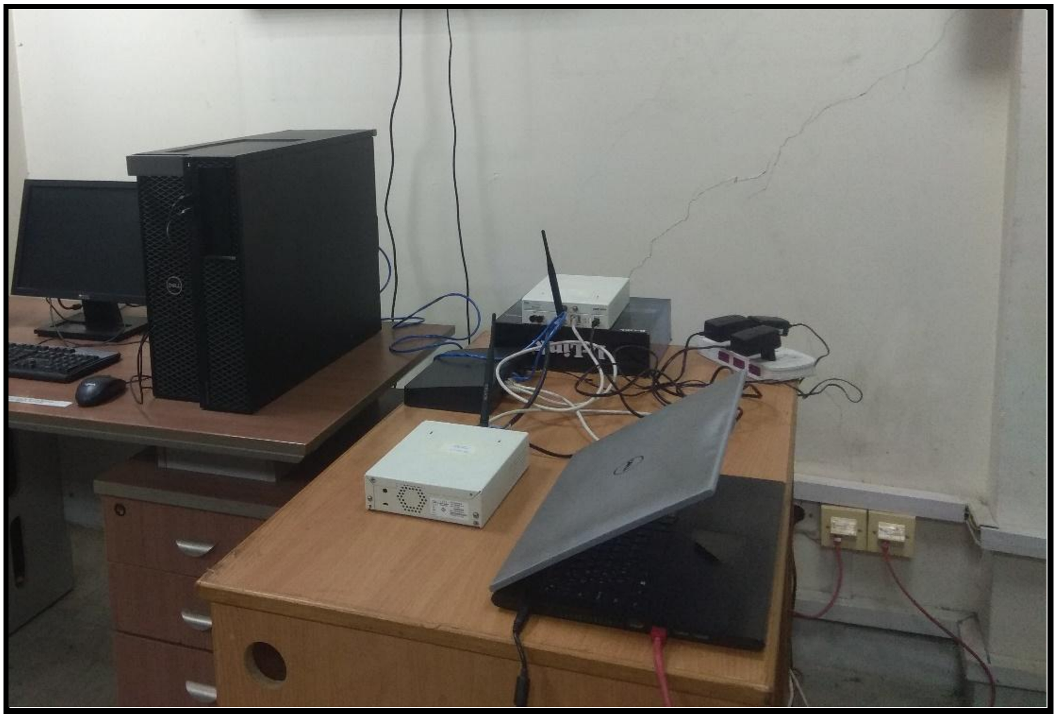


Figure 30: Testbed of AMR for Lab demonstration

The *MATLAB software tool* is selected for the formulation of Data packets and the *python AI libraries* would be used for designing the Neural Network model for MODCOD recognition. In the hardware implementation of AMR, the *GNU Radio (free software development toolkit)* is selected to interface with RF hardware (SDRs N200). All softwares is available at the premises of the National University of Science and Technology (NUST).

Feasibility

The outcome of the last month suggested that the research project has many new elements which are the key features in the success of the overall project.

Input criteria

The simulation results of every Neural Networks (designed for AMR) are based on the software designed I-Q signal. In order to test the following NN model in a lab environment, the SDR based setup is required. Our SDR based perceived design is capable of taking the raw IQ signal over the air, which fulfills the above requirement.

However, the good SNR measurement tool isn't around and we have to estimate SNR using other techniques. This could add little uncertainty in the Lab based results.

NN model Tuning

The multiple Neural Network models are available off the shelf. However, it's not physically possible to test every type of NN in our project. The optimal NN model would be selected after thorough Literature Review and would be implemented. The tuning of the NN would be done accordingly to achieve better results rather than shifting to a new approach.

Research gaps

Coding scheme classification is novel and rare as discussed earlier. Therefore, it requires a fresh start from the beginning. It will be challenging. Therefore, the AMR research project would be divided into two parts i.e. *Modulation recognition and FEC recognition*. Initially, the coding recognition would be tested on the demodulated data. **The results would define our next approach.**

Future work

The project “Neural Network based Modulation and Channel Coding Identification for SATCOM Systems” is new of its kind. As we discussed earlier in feasibility analysis, there are a lot of design limitations in implementing the commercial standard AMR. These technical issues bring a lot of future research opportunities in the betterment of future AMR. Some of them are enlisted here:

- Real-time identification of MODCOD in the SATCOM systems.
- Designing of End-to-End Neural Network model for MODCOD recognition application.
- Development of Modulation/coding schemes which can't be easily recognizable using existing AMR.
- MODCOD recognition in OFDM signals to exploit the Wi-Fi, 3GPP, WiMAX, etc.

References

1. Syed Ali Hassan, “Project Proposal-Neural Network based Modulation and Channel Coding Identification for SATCOM Systems”, NUST (2021).
2. Shannon, C. (n.d.). Communication In The Presence Of Noise. *Proceedings of the IEEE 86*.
3. Tahir, B. (2017). BER comparison between Convolutional, Turbo, LDPC, and Polar codes. *24th International Conference on Telecommunications (ICT)*. DOI:10.1109/ICT.2017.7998249
4. School of Electrical Engineering, Aalto University, Espoo, Finland. (2015). Identification of Error Correction Codes in Signals Intelligence. Master’s Thesis.
5. Proakis, J. (2000). *Digital Communications 4th Edition*. McGraw-Hill, Boston.
6. Sabatier, F. (n.d.). *Polar coding tutorial*. EPFL. Retrieved December 15, 2021, from http://ipgdemos.epfl.ch/polarcodestutorial/
7. K. Kim and A. Polydoros, “Digital modulation classification: the bpsk versus qpsk case," Proc. IEEE Military Communications Conference (MILCOM), pp. 431-436, 1988.
8. P. Panagiotou, A. Anastasopoulos, and A. Polydoros, “Likelihood ratio tests for modulation classification," Proc. IEEE MILCOM, vol. 2, pp. 670-674, 2000.
9. N. E. Lay and A. Polydoros, “Per-survivor processing for channel acquisition, data detection and modulation classification," IEEE Asilomar Conference onSignals, Systems, and Computers, vol. 2, pp. 1169-1173, 1994.
10. K. M. Chugg, C.-S. Long, and A. Polydoros, “Combined likelihood power estimation and multiple hypothesis modulation classification," IEEE Asilomar Conference on Signals, Systems, and Computers, vol. 2, pp. 1137-1141, 1995.
11. A. K. Nandi and E. E. Azzouz, “Algorithms for automatic modulation recognition of communication signals," IEEE Transactions on Communications, vol. 46, no. 4, pp. 431-436, 1998.
12. X. Zhinan and B. Wenle, “A novel blind recognition algorithm for modulated m-qam signals," IEEE WRI International Conference on Communications and Mobile Computing, vol. 1, pp. 461-465, 2009.
13. M. P. DeSimio and G. E. Prescott, “Adaptive generation of decision functions for classification of digitally modulated signals," Proc. IEEE National Aerospace and Electronics Conference, pp. 1010-1014, 1988.
14. K. Ho, W. Prokopiw, and Y. Chan, “Modulation identification by the wavelet transform," Proc. IEEE MILCOM, vol. 2, pp. 886-890, 1995.
15. V. Ramakomar, D. Habibi, and A. Bouzerdoum, “Automatic recognition of digitally modulated communications signals," Proc. IEEE International Symposium on Processing and Its Applications, vol. 2, pp. 753-756, 1999.
16. J. J. Popoola and R. van Olst, A novel modulation-sensing method," IEEE Vehicular Technology Magazine, vol. 6, no. 3, pp. 60-69, 2011.
17. W. Wei and J. M. Mendel, A fuzzy logic method for modulation classification in nonideal environments," IEEE Transactions on Fuzzy Systems, vol. 7, no. 3, pp. 333-344, 1999.

18. M. Pedzisz and A. Mansour, “Automatic modulation recognition of mpsk signals using constellation rotation and its 4th order cumulant," Digital Signal Processing, vol. 15, no. 3, pp. 295-304, 2005.
19. S. Gulati and R. Bhattacharjee, “Automatic blind recognition of noisy and faded digitally modulated MQAM signals," IEEE India Conference, pp. 1-6, 2006.
20. C. Louis and P. Sehier, “Automatic modulation recognition with a hierarchical neural network," Proc. IEEE MILCOM, pp. 713-717, 1994.
21. H. Gang, L. Jiandong, and L. Donghua, “Study of modulation recognition based on hocs and svm," IEEE VTC, vol. 2, pp. 898-902, 2004.
22. L. De Vito and S. Rapuano, “Validating a method for classifying digitally modulated signals," Measurement, vol. 42, no. 3, pp. 427-435, 2009.
23. Dario Floreano, Auke Jan Ijspeert, and Stefan Schaal. Robotics and neuroscience. Current Biology, 24(18):R910–R920, 2014. ISSN 0960-9822.
24. Yu-Jie Tang, Qin-Yu Zhang, and Wei Lin. Artificial neural network based spectrum sensing method for cognitive radio. In 2010 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM), pages 1–4, 2010.
25. Howard B. Demuth, Mark H. Beale, Orlando De Jess, and Martin T. Hagan, 2014, “Neural Network Design (2nd. ed.)”, Martin Hagan, Stillwater, OK, USA
26. T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, “Convolutional, long short-term memory, fully connected deep neural networks," International Conference on Acoustics, Speech and Signal Processing, pp. 4580-4584, 2015.
27. C. Louis and P. Sehier, “Automatic modulation recognition with a hierarchical neural network," Proc. IEEE MILCOM, pp. 713-717, 1994.
28. J. Jagannath et al., "Artificial Neural Network Based Automatic Modulation Classification over a Software Defined Radio Testbed," 2018 IEEE International Conference on Communications (ICC), 2018, pp. 1-6, doi: 10.1109/ICC.2018.8422346.
29. S. Peng et al., "Modulation Classification Based on Signal Constellation Diagrams and Deep Learning," in IEEE Transactions on Neural Networks and Learning Systems, vol. 30, no. 3, pp. 718-727, March 2019, doi: 10.1109/TNNLS.2018.2850703.
30. P. Wu, B. Sun, and S. Su, “Automatic modulation classification based on deep learning for software-defined radio,” Mathematical Problems in Engineering, vol. 2020, Article ID 2678310, 13 pages, 2020.
31. N. Daldal, Z. Cömert and K. Polat, "Automatic determination of digital modulation types with different noises using convolutional neural networks based on time–frequency information", Appl. Soft Comput., vol. 86, Jan. 2020
32. Z. Zhang, C. Wang, C. Gan, S. Sun and M. Wang, "Automatic Modulation Classification Using Convolutional Neural Network With Features Fusion of SPWVD and BJD," in IEEE Transactions on Signal and Information Processing over Networks, vol. 5, no. 3, pp. 469-478, Sept. 2019, doi: 10.1109/TSIPN.2019.2900201
33. T. J. O’Shea, T. Roy and T. C. Clancy, "Over-the-Air Deep Learning Based Radio Signal Classification," in *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 168-179, Feb. 2018, doi: 10.1109/JSTSP.2018.2797022.
34. M. Kulin, T. Kazaz, I. Moerman and E. De Poorter, "End-to-End Learning From Spectrum Data: A Deep Learning Approach for Wireless Signal Identification in

Spectrum Monitoring Applications," in IEEE Access, vol. 6, pp. 18484-18501, 2018, doi: 10.1109/ACCESS.2018.2818794.

35. P. Ghasemzadeh, S. Banerjee, M. Hempel and H. Sharif, "Accuracy Analysis of Feature-based Automatic Modulation Classification with Blind Modulation Detection," 2019 International Conference on Computing, Networking and Communications (ICNC), 2019, pp. 1000-1004, doi: 10.1109/ICCNC.2019.8685638.
36. Subbarao, M.V. and Samundiswary, P., 2019. K-nearest neighbors based automatic modulation classifier for next generation adaptive radio systems. Int. J. Sec. Appl.(IJSIA), NADIA, 13(4), pp.41-50.
37. Y. Zeng, M. Zhang, F. Han, Y. Gong and J. Zhang, "Spectrum Analysis and Convolutional Neural Network for Automatic Modulation Recognition," in IEEE Wireless Communications Letters, vol. 8, no. 3, pp. 929-932, June 2019, doi: 10.1109/LWC.2019.2900247.
38. Kaisheng Liao, Yaodong Zhao, Jie Gu, Yaping Zhang, and Yi Zhong. Sequential convolutional recurrent neural networks for fast automatic modulation classification. IEEE Access, 9:27182– 27188, 2021
39. Meng Zhang, Yuan Zeng, Zidong Han, and Yi Gong. Automatic modulation recognition using deep learning architectures. In 2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), pages 1–5, 2018. doi: 10.1109/SPAWC.2018.8446021.
40. Mei F, Chen H, Lei Y. Blind Recognition of Forward Error Correction Codes Based on Recurrent Neural Network. Sensors (Basel). 2021 Jun 4;21(11):3884. doi: 10.3390/s21113884. PMID: 34199837; PMCID: PMC8200067.