Deep Learning-Based Automatic Modulation Classification for Telecommunication Systems

By

Sara Sanatimehrizi

A Report Submitted in Partial Fulfillment of the Requirement for the Degree of

MASTER OF ENGINEERING

In The Department of Electrical and Computer Engineering

University of Victoria

SUPERVISORY COMMITTEE

Deep Learning-Based Automatic Modulation Classification for Telecommunication Systems

By

Sara Sanatimehrizi
University of Victoria, 2023

Supervisory Committee

Dr. Amirali Baniasadi, Department of Electrical and Computer Engineering
Supervisor

Dr. Hong-Chuan Yang, Department of Electrical and Computer Engineering
Committee Member

Abstract

Modulation schemes play a crucial role in various communication systems, as they enable the transmission of information through electromagnetic signals. Accurately identifying the modulation scheme employed in a signal is essential for efficient signal processing, interference mitigation, and overall system performance. However, predicting modulation schemes based solely on their features remains a challenging task due to the complexity and variability of modern communication signals.

This thesis addresses the problem of modulation scheme prediction by developing and evaluating a model and algorithm that capable to analyze the distinctive features of different modulation schemes. The dataset used in this study is a real-time series dataset obtained from MCI, consisting of 36,000 signals with features such as Modulation, In-phase Signal, Quadrature Signal, and Signal-to-Interference-plus-Noise Ratio. The goal is to train a fully connected neural network to accurately classify and predict the modulation used in unknown signals.

Experimental results demonstrate the effectiveness of the proposed algorithm, with a validation accuracy of 83.33% and an overall accuracy of 93.90%. While these results indicate the algorithm's capability to predict modulation types and classify instances accurately, it is important to acknowledge that there is room for improvement. In comparison to real-world scenarios, further enhancements can be made to achieve even better results.

It is essential to recognize that the proposed model and algorithm provide a solid foundation for enhancing signal processing and system performance in communication systems. By accurately identifying modulation schemes, this research contributes to the advancement of efficient communication techniques. Future work in this area has the potential to build upon these findings and further refine the algorithm, potentially yielding improved accuracy and robustness when applied to real-world scenarios.

Acronyms

AMC	Automatic Modulation Classification
VR	Virtual Reality
AI	Artificial Intelligence
ANN	Artificial Neural Networks
DL	Deep Learning
RL	Reinforcement Learning
LB	Likelihood-Based
FB	Feature-Based
HOC	Higher-Order Cumulants
MIMO	Multiple-Input Multiple-Output
SVM	Support vector machine
KNN	K-nearest neighbor
AWGN	Additive White Gaussian Noise
CNN	Convolutional Neural Networks
RNN	Recurrent Neural Networks
I	In-phase Components,
Q	Quadrature Component,
SINR	Signal-to-Interference-plus-Noise Ratio
LOS	Line-of-Sight

Table of Contents

List of Tables

List of Figures

Acknowledgments

I would like to express my sincere gratitude to my supervisor, Dr. Amirali Baniasadi, for his unwavering guidance, support, and motivation throughout my project and during my entire academic program. His expertise and insightful feedback have been invaluable in shaping the direction and quality of my research. I am truly grateful for his dedication and commitment to my academic growth.

Furthermore, I extend my appreciation to Dr. Meisam Abdollahi for his consistent support and assistance throughout my project. His valuable input and willingness to lend a helping hand have greatly contributed to the success of my research endeavors. I am thankful for his guidance and the knowledge I have gained under his mentorship.

Dedication

I dedicate this thesis to the most important people in my life: my parents, my husband, and my son. Their unwavering love, support, and understanding have been the driving force behind my academic journey and this accomplishment.

To my parents, thank you for your unconditional love and endless sacrifices. You have always believed in me and encouraged me to pursue my dreams. Your guidance and constant support have shaped me into the person I am today. This thesis is a testament to your unwavering faith in me.

To my loving husband, your unwavering support and encouragement have been my rock throughout this academic journey. Thank you for understanding the countless hours I spent immersed in my research, for being my sounding board, and for providing me with the love and stability I needed to pursue my goals. This thesis is a reflection of our shared dedication and commitment.

Last but certainly not least, I dedicate this thesis to my precious son, who came into this world during my academic journey. You are my greatest joy and source of inspiration. Your arrival has added a new dimension to my life, and I am grateful for the love and happiness you bring. May this thesis be a symbol of the love and dedication I have for you and a testament to the importance of lifelong learning.

Chapter 1

Introduction

The importance of secure and trustworthy communication is paramount in the present era. As communication technology advances promptly, the complex wireless communication environment is increasing. Communication signals with varying modulations have become more diverse and affect the network's complexity [1]. However, this communication is not always perfect and various factors, such as environmental interferences, channel fading, and noise, can degrade the quality of signals [2]. Moreover, the past few years have seen an extraordinary expansion and mounting intricacy of communication networks, commonly referred to as Telcos. This has resulted in a traffic volume augmentation by a factor of 1000 alongside elevation in users' count up by one hundred times [3].

This thesis focuses on addressing the significant challenge of signal quality degradation in wireless communication systems. The impact of degradation factors, such as high levels of noise, can result in errors, decreased data throughput, and degraded system performance [2]. The aim of this research is to develop effective techniques to mitigate these factors and enhance the overall quality and reliability of wireless communication systems. Our network acknowledges its limitations but strives to address this issue, aiming to contribute to the advancement of secure and trustworthy communication in complex wireless environments. It is important to note that there is room for future work to further improve the results of our network.

Automatic Modulation Classification (AMC) plays a vital role in wireless communication by identifying the modulation scheme used in a received signal. This identification is significant for various applications, including interference detection, signal demodulation, and spectrum sensing. On the other hand, traditional machine learning algorithms have limitations in challenging environments with high noise levels [1] [2]. The landscape of machine learning has experienced major transformations due to recent advancements in deep neural networks. These revolutionary innovations have led to notable improvements in classification accuracy, particularly when dealing with noisy data. Deep learning models can automatically extract relevant signal features and learn complex patterns, making them well-suited for modulation classification tasks [2]. Several researchers have proposed real-world signal datasets containing noise, providing a more realistic evaluation of classification algorithms' performance.

This thesis explores a deep learning-based network architecture on a real-world dataset obtained from MCI that includes noise. Our goal is to assess the accuracy of our approach in identifying the modulation scheme used in a received signal in the presence of noise. Using real-world datasets containing noise is crucial for evaluating the performance of modulation classification algorithms in challenging environments. Numerous investigations have illustrated the significance of data collection in improving the accuracy and effectiveness of automatic modulation classification algorithms. These studies are documented in sources [4], [5], [6], [7], and [8].

In conclusion, this work aims to evaluate the accuracy and efficiency of automatic modulation classification algorithms for mobile networks based on the deep learning network. Chapter 2 provides background information on AMC approaches. In addition, previous studies are reviewed in this section. Chapter 3 defines the proposed method, including dataset and preprocessing. Chapter 4 presents the results, and the final chapter offers a brief conclusion.

Chapter 2

Background

2.1- Subject Review

Throughout the previous century, there have been numerous improvements in data transmission and communication technology. These innovations have made it feasible for everyone to conveniently obtain services such as Roaming calls, Video calls, Virtual Reality (VR), and Augmented Reality (AR) which are now fundamentally integrated into our daily routines [9]. The proliferation of wireless communication has necessitated the emergence of compact networks that consume vast amounts of frequency spectrum, in order to manage heavy traffic. This circumstance engenders difficulties like interference from multiple signals on identical channels and signals distortions during transmission through propagation channels. [10].

2.1.1- Wireless Communications

Information transmission is feasible from one point to another without cable connectors as a consequence of wireless networks. Wireless network infrastructures are distinctive based on their communication apparatuses, but most systems include transmitters and receivers performing modulation and demodulation through information decoding and encoding [11]. These networks, such as WLAN, Bluetooth, Radars, Satellites, and Cellular or Mobile networks, utilize a non-cooperative configuration where radio signals are encoded by different modulation formats [10].

In this context, Automatic Modulation Classification (AMC) has become increasingly important in modern communication systems. The non-cooperative configuration in these systems allows radio signals to be encoded by different modulation formats, and AMC helps identify the modulation types of received signals, allowing for efficient signal demodulation. However, AMC is a challenging task due to factors such as the increasing number of modulation formats, intra-class discrimination of higher-order digital modulations, and strong channel impairments

[12]. To learn a classification model, conventional feature engineering techniques can be used, such as feature extraction and feature selection, with either supervised or unsupervised learning [13]. Overall, AMC plays a critical role in signal processing and communication, and improving its efficiency remains a priority for many software-defined radio-based communications [14].

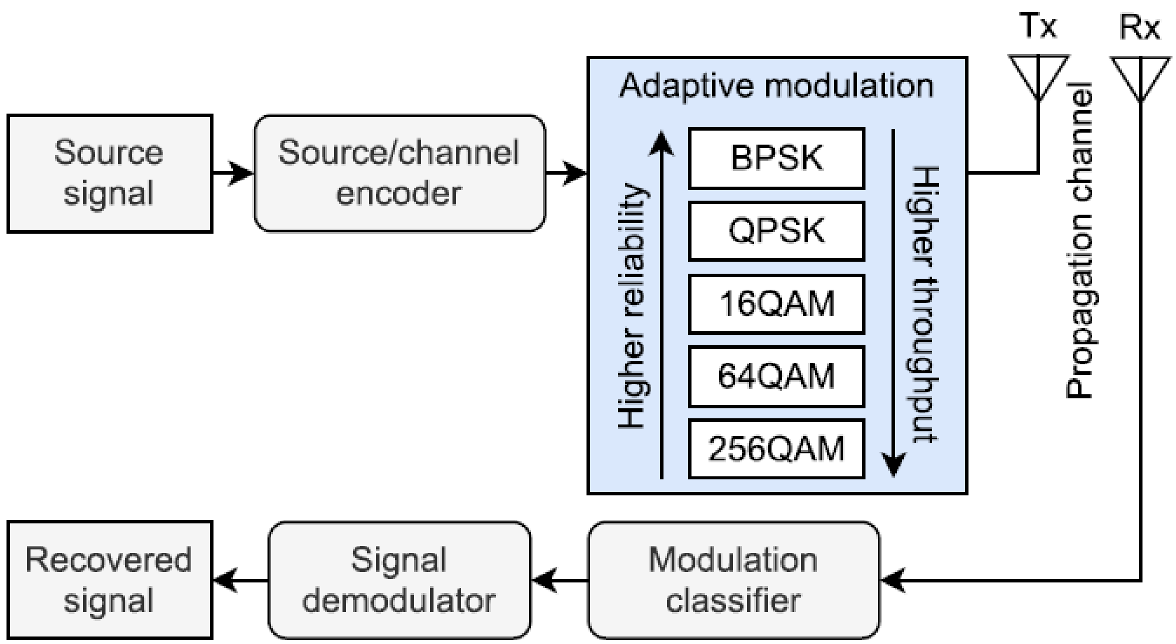


Fig1. A modulation scheme that adjusts according to the changing conditions of a communication system. [10]

Communication systems use various modulation techniques to balance spectrum efficiency and transmission reliability. Analog modulation techniques such as AM, PM, and FM encode a source signal onto a high-frequency periodic waveform. Digital modulations, such as QAM, are preferable due to their better coordination with digital data and stronger robustness against interference.

Within the realm of electronic means for conveying information, it is customary to first transform an original signal into a digital format. The ensuing digitized form subsequently undergoes encoding so as to ensure that any potential errors are minimized, and data privacy is maximized. Once this has been done, the digitally processed output travels through what can be referred to as a "digital modulator," depicted within Figure 2. The modulation process modifies different characteristics of the carrier signal based on a pre-defined modulation technique.

During transmission, the modulated signal travels over propagation channels. To determine the appropriate modulation for the incoming signal, its radio characteristics are inferred using a learned AI model. This allows for the selection of the optimal modulation scheme that suits the given signal.

Following transmission, the received signal is subjected to a modulation classifier, which identifies the modulation scheme used. This information is then used by the signal demodulator to extract the encoded signal from the modulated carrier. The channel decoder reverses the encoding process and corrects any errors introduced during transmission. Finally, the source decoder reconstructs the original source signal, resulting in the recovered signal.

Through this process, communication systems achieve efficient and reliable transmission by leveraging digital modulation techniques and appropriate modulation selection based on radio characteristics [10].

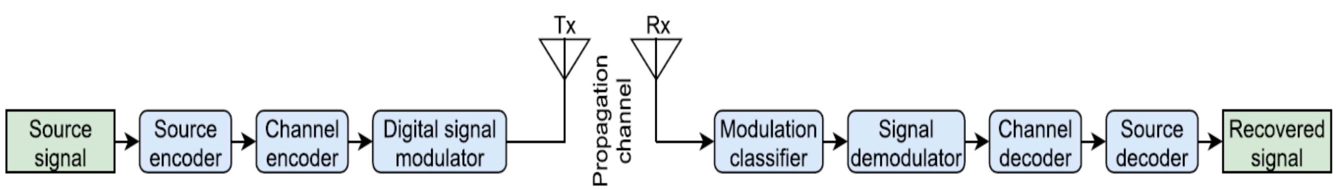


Figure 2. Digital communication process [10]

2.1.2- Modulation and AMC Methods

Modulation is a fundamental process in communication systems that involves modifying certain characteristics of a carrier signal to carry information. It allows the encoding of the information onto the carrier signal, enabling it to be transmitted efficiently over a communication channel.

In the context of digital modulation, constellations are used to represent the modulation scheme. A constellation consists of a set of points in a two-dimensional or multi-dimensional signal space. Each point in the constellation represents a symbol, which carries a certain amount of information. The number of points in the constellation corresponds to the number of symbols that can be transmitted in a single modulation symbol period.

In the case of 16-QAM (Quadrature Amplitude Modulation), the constellation consists of 16 points. It is divided into four quadrants, with each quadrant containing four points arranged in a square grid. The constellation points are typically represented by complex numbers, where the real and imaginary parts determine the amplitude and phase of the signal, respectively.

Each of these 16 points represents a specific combination of amplitude and phase, allowing the transmission of 4 bits per symbol in the case of 16-QAM. By modulating the carrier signal using these constellation points, digital data can be efficiently transmitted and decoded at the receiver end [57].

Within the present narrative, we shall furnish an outline concerning traditional AMC techniques. Said methods are subcategorized into two: likelihood-based (LB) and feature-based (FB) [10].

In the context of modulation identification, likelihood-based approaches use probability theories and hypothesis models to solve problems under known or unknown channel information. These approaches can achieve optimal classification accuracy with perfect knowledge of signal and channel models but require high computational complexity to estimate model parameters. On the other hand, feature-based approaches follow a regular machine learning framework for classification tasks and are preferred for practical systems due to their relatively easy implementation and low complexity. However, these approaches have some drawbacks such as weak discriminative experience of handcrafted features and limited learning capacity of traditional classification algorithms. [15-21].

In recent years, deep learning (DL) has been widely applied in different fields, including communications, to improve the performance of modulation classification. DL has advantages such as automatic feature extraction and high learning capacity, which increase classification accuracy for higher-order modulation formats under synthetic channel deterioration. DL is also useful for processing big data generated by edge devices in Internet-of-Things (IoT) systems. Previous surveys have reviewed likelihood-based and feature-based approaches for automatic modulation classification (AMC) and briefly described DL-based AMC approaches without analyzing deep architectures comprehensively. More recent surveys have focused on the application of DL for modulation classification and other tasks in wireless communications but fail to provide a comprehensive analysis of deep architectures and their potential for boosting accuracy while maintaining acceptable complexity [22-34].

2.1.3- Machine Learning Methods

The genesis of Artificial Intelligence (AI) can be traced back to the decade of 1950, celebrated as a period marked by simple models, mathematical computation and algorithms. AI has also become a more attractive field among researchers since the digital revolution and the development of interactions between humans and machines. Since then, due to the rapid and ongoing advancements in the field of AI, it has become increasingly challenging to forecast its future trajectory and the potential impacts it may have on human life [35].
In Figure 3, different implementing AI methods are illustrated.

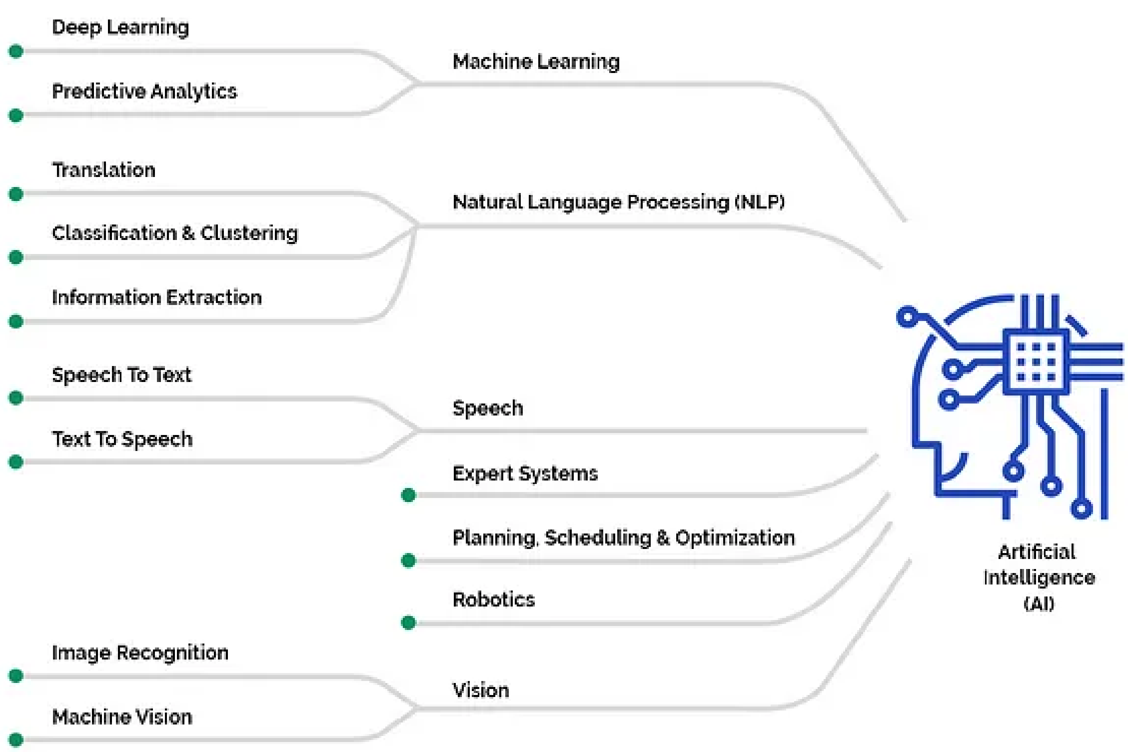


Fig.3 Implementing AI Methods [9]

The production of large amounts of data from communication networks and critical infrastructures, coupled with the need for intelligent analysis of this data, has led to the widespread use of machine learning algorithms in various fields, such as banking, government services, surveillance, crime prevention, and even online gaming.

In the following, Artificial Neural Networks (ANN) and Deep Learning (DL) are described shortly.

- Artificial Neural Networks: the human brain comprises neurons communicating through synapses, and artificial neurons are mathematical models that simulate these biological neurons. The activation function of an artificial neuron is determined by multiplying the inputs by weights and then adding them together [36].

In addition, ANN can adapt to learn how to respond and perform specific tasks based on the provided data for training and previous experiences. Learning schemes can be classified into three key categories, as illustrated in Figure 4 [9].

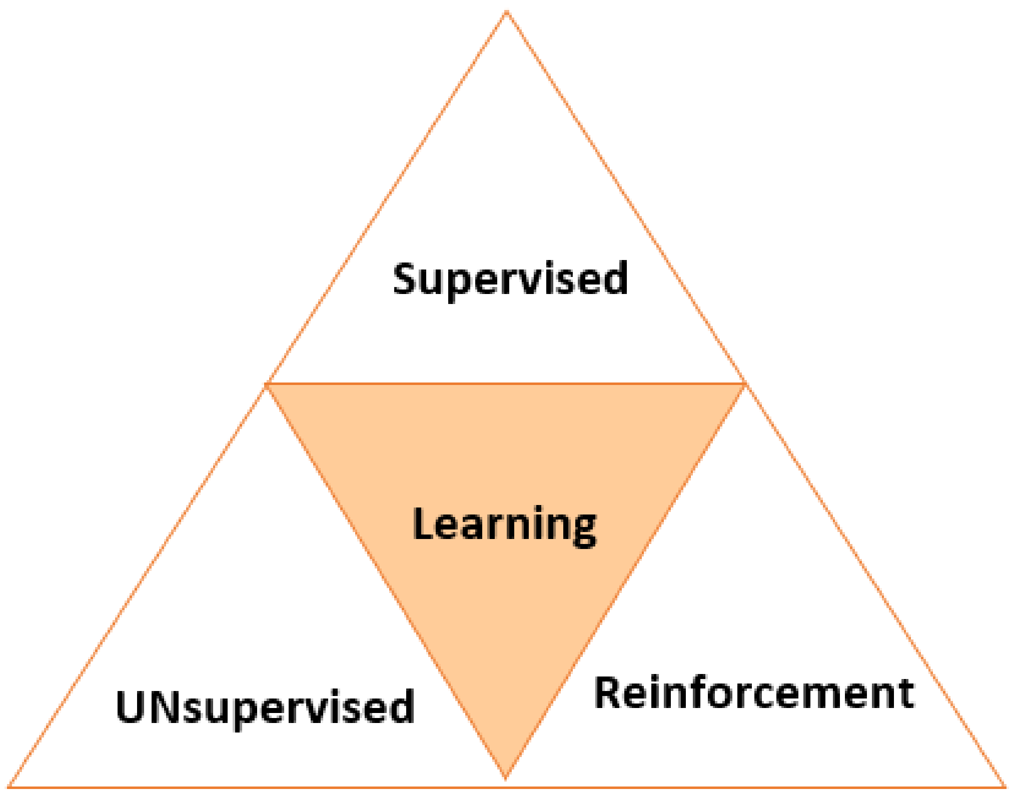


Fig. 4 Classification of Learning Schemes

Supervised learning algorithms are trained using labeled data. When working with labeled data, both the input and expected outputs for the system are known. Unlike supervised learning, unsupervised learning can be performed without knowledge of the output data. The goal of unsupervised learning is to explore input data and the amount of data needed to train the model on unlabeled data [9] [35].

Despite these two methods, Reinforcement Learning (RL) is trained by the system's data. RL aims to learn about the system environment and find the best strategist for a specific agent in different environments. This makes RL algorithms useful for robotics, games, and effective navigation [9] [35].

- Convolutional neural networks (CNNs) are a popular and successful deep architecture that utilizes convolution operations to learn higher-order features in data. CNNs are particularly suitable for processing high-dimensional unstructured data, such as images, but can also be used for text, signals, and other continuous responses. A basic CNN consists of an input layer, multiple hidden layers, and an output layer. A typical hidden layer involves a convolutional layer followed by an activation layer, and other additional layers such as pooling, fully connected, and normalization layers. Compared to other deep network architectures, CNNs are more adept at learning meaningful features from raw data, as they can share connections and produce more efficient outcomes [37-39]

- Deep Learning: Deep learning models are taught utilizing massive collections of labeled data and multi-layered neural network architectures. The term "deep" in deep learning refers to using multiple layers in the network. Deep learning is a modern diversity that applies to infinite limited-sized layers, allowing for practical application and optimal implementation while maintaining theoretical universality in balanced conditions [35].

In recent years, deep learning has gained popularity, and many open-source deep learning frameworks have been introduced by both academic and industry communities. These frameworks provide high-level programming interfaces that build blocks for designing, training, validating, and deploying deep learning models. Among these frameworks, TensorFlow is among the most popular, based on data from Google searches, GitHub surveys, and tags on the Overflow Stack website [40].

DL with FFNN, RNN/LSTM, and CNN architectures has been successful in various domains, including bioinformatics and computer vision. DL is particularly useful for pattern analysis applications that involve big data. DL is also being adopted to solve challenging problems in communications. Modulation classification has benefited from DL methods that exploit different architectures such as DNN, FFNN, RNN/LSTM, and CNN, leading to improved classification accuracy and processing speed. CNNs can process high-dimensional unstructured data, such as constellation diagrams and spectrogram images, to extract representational features.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \tag{1}$$

The confusion matrix serves as a performance evaluation metric for machine learning classification problems. It provides a comprehensive assessment by comparing the actual target values with the predicted values. The target variable typically consists of two values: Positive or Negative. The confusion matrix arranges these values in columns for the actual target values and in rows for the predicted target values.

The accuracy of modulation classification is measured by True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) rates, and detailed results can be reported using a confusion matrix [10].

2.2- Literature Review

Digital modulation technology using telecommunications infrastructure has become increasingly popular with the growth and development of technology in newer generations and the diversity of applications that can be executed using telecommunications data. Despite the added complexity of signal processing, digital modulation has become widely adopted in practical applications [1]. The main goal of modulation classification is to determine the specific modulation type i in the received signal $r(t)$[41].

$$r(t) = s(t; u_i) + \mathrm{n(t)} \tag{2}$$

In this equation, the complex baseband envelope of the received signal without noise is presented by $s(t; u_i)$ and noise added to this as $\mathrm{n(t)}$. The symbol u_i represents the modulation scheme or type. It is a variable that denotes a specific modulation format or technique used in the transmission of the signal [41].

As mentioned before, the earlier research may be divided into Likelihood-Based (LB) and Feature-Based (FB) techniques [42]. The former, which approaches modulation classification as a problem of multiple hypothesis testing, produces the best results but has a significant computational cost [43]. If the right features and classifiers are used, the latter collects characteristics to describe the signal and can reach almost ideal performance with less complexity [42]. In order to categorize linearly modulated signals with less complexity, two approximation LB methods based on the Gauss-Legendre and Gauss-Hermite quadrature rules are suggested in

[44]. Higher-Order Cumulants (HOC)s, a feature used in [45]'s powerful FB algorithm, are used to stop multipath fading.

In [46] and [47], respectively, Cyclostationarity features and approximation entropy features are used for modulation categorization. HOC features and ANN classifiers are the best options for the trade-off between performance and complexity over Multiple-Input Multiple-Output (MIMO) channels [48]. With reasonable training time, a decision tree classifier outperforms an ANN classifier in MIMO networks [49].

Support Vector Machine (SVM) and K-nearest Neighbor (KNN) are two more classifiers that are used by FB techniques in [48], [50], and [51].

Moreover, Deep Learning (DL) has exploded in popularity in recent years, which has led to an increase in the use of modulation categorization based on DL [52] [53]. Implementing DL for modulation classification has several benefits [54].

First, communications technology makes gathering the enormous amounts of data needed for DL simple. Second, DL does not require manual feature selections, which can be pretty difficult in modulation classification. Third, there are great opportunities for modulation categorization in complex circumstances because of the rapid and significant progress of DL technology. [2]

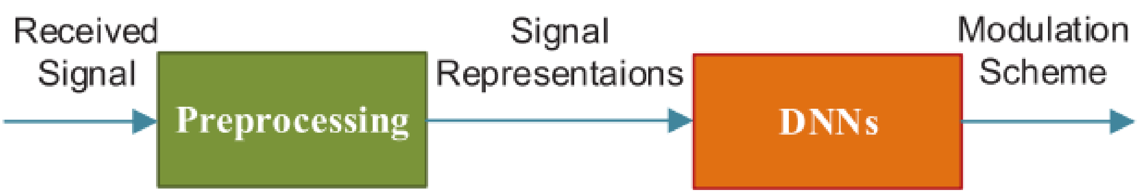


Fig5. DL-based modulation categorization has two steps [2].

This essay uses a deep learning algorithm to build a network based on a real-world dataset extracted from MCI. MCI is one of the largest mobile network providers in Iran, and they have accumulated a vast amount of real-world signal data that can be used for modulation classification tasks. Recently, some researchers have utilized MCI's signal dataset to evaluate the performance of various machine-learning algorithms for modulation classification (e.g., [55]; [56]).

Several researchers have proposed real-world signal datasets containing noise, providing a more realistic evaluation of classification algorithms' performance. For example, Alam et al. (2021) proposed a dataset containing eight modulation schemes and varying levels of Additive White Gaussian Noise (AWGN). They evaluated the performance of several machine learning and deep learning algorithms on this dataset and found that deep learning algorithms outperformed traditional machine learning algorithms in high-noise environments [4].

Cheng et al. (2020) proposed a dataset containing four different modulation schemes and varying levels of AWGN and fading. They evaluated the performance of several deep learning algorithms on this dataset and found that Convolutional Neural Networks (CNN)s and Recurrent Neural Networks (RNN)s achieved the highest classification accuracies [5].

Hayajneh et al. (2019) proposed a dataset containing six different modulation schemes and varying levels of AWGN and Rayleigh fading. They evaluated the performance of several machine learning algorithms on this dataset and found that SVMs and random forests achieved the highest classification accuracies [6].

Srinivasan et al. (2020) proposed a dataset containing three different modulation schemes and varying levels of AWGN and multipath fading. They evaluated the performance of several deep learning algorithms on this dataset and found that CNNs achieved the highest classification accuracy [7].

Wei et al. (2021) proposed a dataset containing eight different modulation schemes and varying levels of AWGN and fading. They evaluated the performance of several deep learning algorithms on this dataset and found that CNNs achieved the highest classification accuracy [8].

In summary, developing accurate and efficient modulation classification algorithms is essential for various wireless communication applications, especially in the presence of noise and interference. By leveraging MCI's real-world signal dataset and advanced deep learning techniques, this study aims to contribute to the development of more reliable and robust modulation classification algorithms.

Chapter 3

Methodology

3.1- Problem Statement

The problem addressed in this thesis is the prediction of modulation schemes based on their distinctive features. Modulation schemes are essential in various communication systems, and accurately identifying the modulation scheme employed in a given signal can significantly impact signal processing, interference mitigation, and overall system performance. However, the prediction of modulation schemes solely based on their features remains a challenging task due to the complexity and variability of modern communication signals. Therefore, this thesis aims to develop and evaluate a prediction model and algorithm that can effectively analyze the features of modulation schemes to accurately classify and predict the modulation used in unknown signals.

Then, by using the collected labeled data, which includes signal amplitude, signal phase, and Signal-to-Interference-plus-Noise Ratio, we determine the necessary sample data for training the model. After the training phase, unlabeled sample data is presented to the optimized model for predicting the signal modulation, and the predicted label is compared with the actual signal's label.

3.2- Dataset

The dataset used in this study is a real-time series dataset obtained from MCI. It consists of 36,000 signals, each with four features: Modulation, In-phase components (I), Quadrature components (Q), and Signal-to-Interference-plus-Noise Ratio (SINR). The dataset is a Line-of-Sight (LOS) dataset, which means that it includes noise. Each signal contains 128 samples, with I and Q values that vary for each sample while Modulation and SINR remain constant. There are four classes of modulation in the dataset: 2, 4, 16, and 64, which correspond to the classes 0, 1, 2, and 3, respectively.

3.3- Preprocessing

To prepare the dataset for the analysis, we first removed any missing values and outliers. We then added a column to the dataset indicating the class label for each signal based on its modulation value. We then split the dataset into train and test sets in a 70,30 ratio, respectively. Then, we allocate 50% of the test sets to the validation sets. Hence, there are 25200 training samples, 5400 validation samples, and 5400 test samples. The training subset consists of labeled examples (input data paired with corresponding output labels) that the model uses to learn patterns and relationships in the data. During the training phase, the validation dataset plays a crucial role in model selection and optimizing the model's hyperparameters. It helps in fine-tuning the model's performance by providing feedback on how well it generalizes to unseen data. On the other hand, the test dataset is used to assess the final performance of the trained model. It serves as an independent and unbiased evaluation of the model's ability to make accurate predictions on new, unseen data. In summary, the validation dataset aids in refining the model, while the test dataset provides an objective measure of its performance on unfamiliar data.

In addition, in each training iteration, the order of data is shuffled to improve the model’s evaluation accuracy during the validation phase. Furthermore, if a feature has a significantly higher variance compared to other features, it may impact the estimation of the activation function and prevent the learner from properly learning from other features. Therefore, after data separation, data normalization is applied using Equation 3 to standardize the features. Both feature values and labels are standardized using the standard scalar method according to Equation 3, which removes the mean and scales the features to have unit variance. Here, Equation 3 calculates the normalized value of a data sample (x):

$$X_{normalized} = \frac{x - mean(x_{training})}{s} \tag{3}$$

In this equation $mean(x_{training})$ denotes the training samples’ mean value, which is assumed to be zero if not present, and s is the standard deviation measure considered for the training samples, assumed to be one if not provided.

Next, we tuned the deep learning model using Keras Tuner to further reduce the effect of noise in the dataset. Keras Tuner is an open-source library that allows us to efficiently search for the optimal hyperparameters of a deep learning model. We used the hyperband algorithm, which is a state-of-the-art optimization method that combines random search with early stopping. The hyperparameters we tuned using

Keras Tuner include the number of layers, the number of neurons per layer, the activation function for each layer, the optimizer, and the learning rate.

To use Keras Tuner, we first defined the search space for each hyperparameter, including the possible range of values or choices. We then trained the model with each combination of hyperparameters and evaluated its performance using the validation set. The best performing model was selected based on its accuracy on the validation set and was used to make predictions on the test set.

Overall, using Keras Tuner allowed us to optimize the performance of the deep learning model and reduce the effect of noise in the dataset.

Moreover, in order to input our data into a deep learning model, the data was flattened into a vector format where all samples for each signal corresponds to a single vector which is include 128 I values, 128 Q values, SINR and its label. This vector format is compatible with the input requirements of the deep learning model. Finally, we randomized the order of the samples within each class to avoid any bias in the training process.

3.4- Proposed Methodology

For modeling the proposed plan, a deep multilayer neural network with a supervised learning algorithm that learns function 2 has been used. A dataset with m=258 input domain and o=4 output domain is used for training (equation 4). The output represents a binary encoding scheme for labeling classes. Each binary number within the array has a length of 4 bits, and the presence of a '1' in a specific bit position indicates the label for that class. The feature set X, including x_m, ..., x_2, x_1, x_0 is used as the input layer, and the target set Y, including y_1, y_2, is used as an output layer, representing the data samples and their labels. The designed model is capable of learning a non-linear regression function estimator. Up to L=3 middle layers, also known as hidden layers, are considered between the input and output layers. Each layer consists of a set of features called neurons. Each neuron in the hidden layer changes the value of the previous layer neuron by its linear weights sum. The number of neurons in each layer is optimized using the tuning method to obtain an optimal model and adjusted for the hidden layers based on the specified number of hyperparameters in the Table below.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 hidden_0 (Dense)            (None, 256)               66048

 hidden_1 (Dense)            (None, 224)               57568

 hidden_2 (Dense)            (None, 224)               50400

 classifier (Dense)          (None, 4)                 900

=================================================================
Total params: 174,916
Trainable params: 174,916
Non-trainable params: 0
_________________________________________________________________
None
```

Table1. Best model summery

$$f(x): R^m \rightarrow R^o \tag{4}$$

3.5- Implementation

In order to implement the proposed model in this research, the Google Colab Notebook platform has been used to provide the necessary processing power for running complex deep learning algorithms and to reduce the complexity of installing required packages and libraries tailored to the version of Python used. Additionally, various Python libraries such as Matplotlib, Sklearn, Panda, Numpy, Keras, and TensorFlow have been used for running the model algorithm.

Chapter 4

Experimental Result

In this chapter, we present the experimental results obtained from the application of our algorithm, which incorporates a patience value of 3. This section focuses on analyzing the performance of the algorithm by presenting the confusion matrix and corresponding plots. The confusion matrix provides a comprehensive evaluation of the algorithm's classification accuracy and errors, while the plots offer visual insights into its behavior and performance.

In my research, the confusion matrix plays a crucial role in evaluating the performance of modulation classification across different schemes, specifically for modulation 2, 4, 16, and 64. The matrix has a size of 4x4.

True Positive (TP) represents the instances where the model correctly identifies a specific modulation scheme. These values are located on the diagonal of the confusion matrix, indicating accurate predictions. For example, a TP count in the cell corresponding to modulation 2 signifies that the model correctly predicted modulation 2 when it was the actual modulation.

True Negative (TN) indicates the instances where the model correctly identifies the absence of a particular modulation scheme. TN values are found outside the row and column of the specific modulation scheme. They represent accurate predictions of other modulation schemes when the actual modulation is different. For instance, in the modulation 2 row, TN represents the count of instances where the model correctly predicts a modulation other than 2 when the actual modulation is not 2.

False Positive (FP) represents the instances where the model incorrectly predicts a specific modulation scheme when the actual modulation is different. FP values are located in cells where the predicted modulation scheme (row) does not match the actual modulation scheme (column). For example, an FP count in the cell corresponding to modulation 2 and actual modulation 4 indicates that the model incorrectly predicted modulation 2 instead of 4.

False Negative (FN) refers to the instances where the model fails to identify a specific modulation scheme that is actually present. FN values are found in cells where the predicted modulation scheme (row) does not match the actual modulation scheme (column). For instance, an FN count in the cell corresponding to modulation 4 and actual modulation 2 indicates that the model failed to identify modulation 4 and incorrectly predicted modulation 2.

By analyzing the values in the confusion matrix, important metrics such as accuracy can be calculated. This metric provides a comprehensive understanding of the classification model's performance in accurately identifying different modulation schemes.

To construct the confusion matrix, we evaluated our algorithm on a carefully curated test dataset, which consists of labeled instances that were not used during the model training phase. The matrix presents the distribution of instances across various predicted and actual class labels. Each row represents the instances in a predicted class, while each column represents the instances in an actual class. The diagonal of each confusion matrix represents the number of correctly predicted instances for each modulation value, including 2, 4, 16, and 64. This analysis allows us to assess the algorithm's performance and identify areas for improvement.

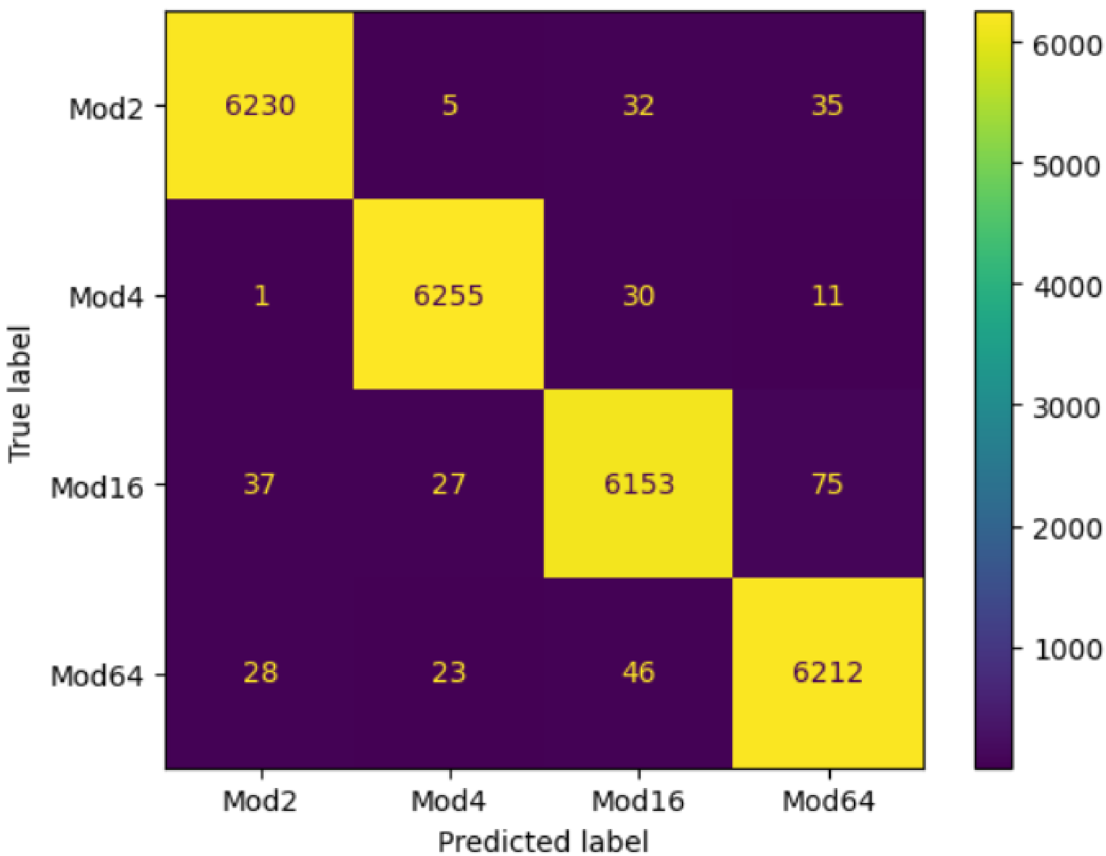


Fig 6. Confusion matrix for train data

This confusion matrix demonstrates that our algorithm has successfully classified instances across all four modulation values (2, 4, 16, and 64) with high accuracy. Each diagonal element of the confusion matrix corresponds to the number of instances correctly trained for a particular modulation value.

By observing the diagonal elements, we can conclude that our network has learned the distinguishing characteristics of each modulation type and made accurate predictions.

The accurate classification of instances for each modulation value showcases the robustness and generalizability of our trained network. It suggests that the algorithm has successfully captured the relevant features and patterns associated with each modulation type, enabling it to make precise predictions.

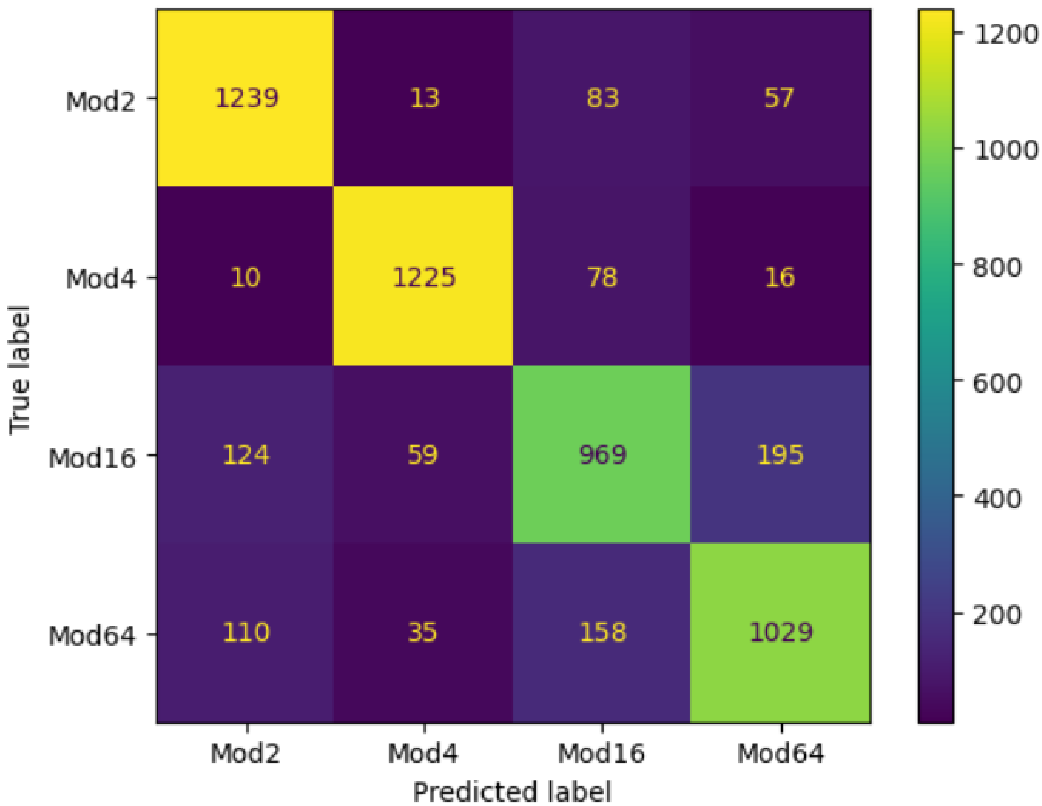


Fig 7. Confusion matrix for validation data

The confusion matrix for the validation data is a crucial evaluation tool that allows us to assess the performance of our algorithm on unseen data. By analyzing the diagonal elements, we can determine the number of instances that were correctly classified for each modulation value (2, 4, 16, and 64). This demonstrates the algorithm's ability to accurately predict the correct modulation type. The off-diagonal elements provide insights into the misclassifications made by the algorithm, helping us identify areas for improvement. Overall, the validation data's confusion matrix serves as a vital validation tool, guiding us in refining and optimizing the algorithm for real-world applications.

In the confusion matrix for the validation data, we observed that the number of correctly classified instances for modulation values 16 and 64 was relatively lower compared to the other modulation values. This finding can be attributed to the inherent difficulty of predicting modulation types 16 and 64 due to the proximity of points in the feature space.

The closer proximity of points in the feature space for these modulation values poses a challenge for the algorithm to distinguish between them accurately. As a result, there might be a higher likelihood of misclassifications or lower classification accuracy for modulation types 16 and 64.

The difficulty in predicting modulation types that are closely located in the feature space underscores the importance of addressing this issue to enhance the algorithm's performance.

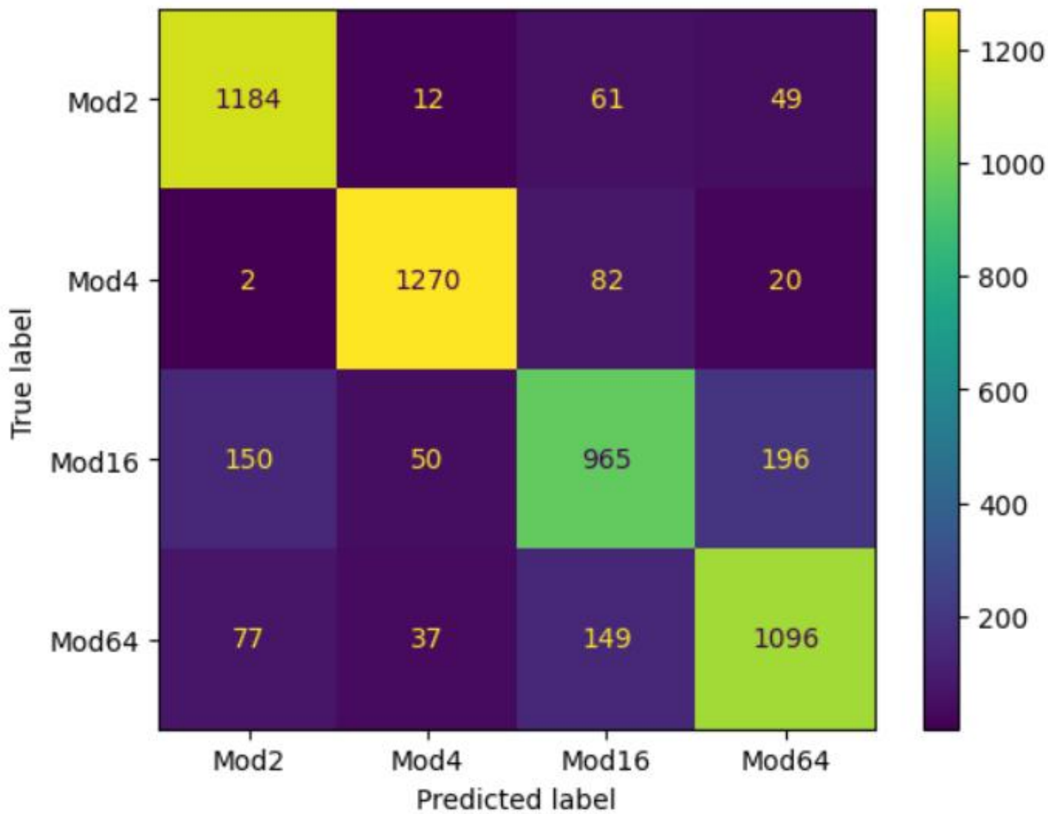
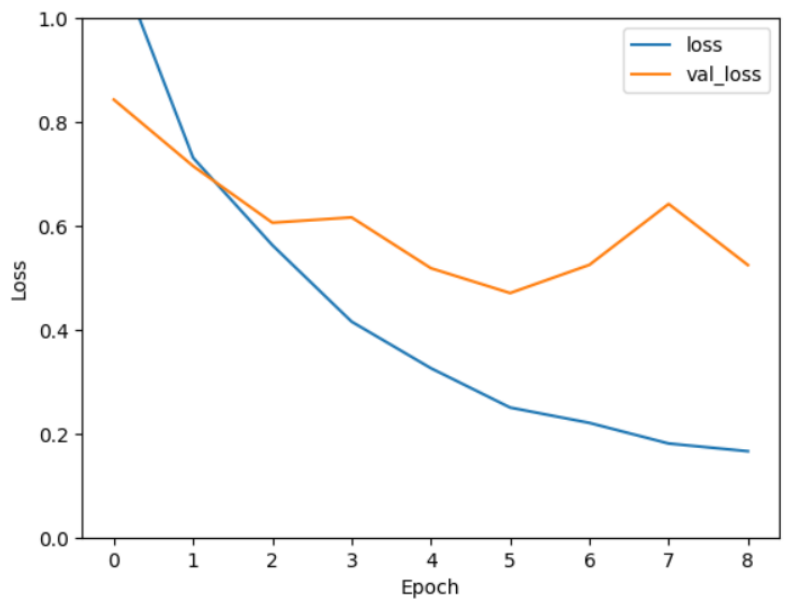


Fig 8. Confusion matrix for test data

The similarity between the numbers in the confusion matrix for the test data and the validation matrix indicates consistency in the algorithm's performance across different datasets. It suggests that the algorithm's ability to classify instances for each modulation value (2, 4, 16, and 64) remains stable and reliable when tested on unseen data.

The fact that the numbers in the confusion matrix for the test data approximately match those in the validation matrix indicates that the algorithm has successfully generalized its learning from the training phase to new, independent samples. This consistency strengthens our confidence in the algorithm's performance and its capability to accurately classify different modulation types.

The similarity in the confusion matrix numbers for the test and validation data also suggests that the algorithm's performance is robust and not influenced by potential biases or specific characteristics of either dataset. It demonstrates the algorithm's ability to handle various instances and generalize well to different testing scenarios.

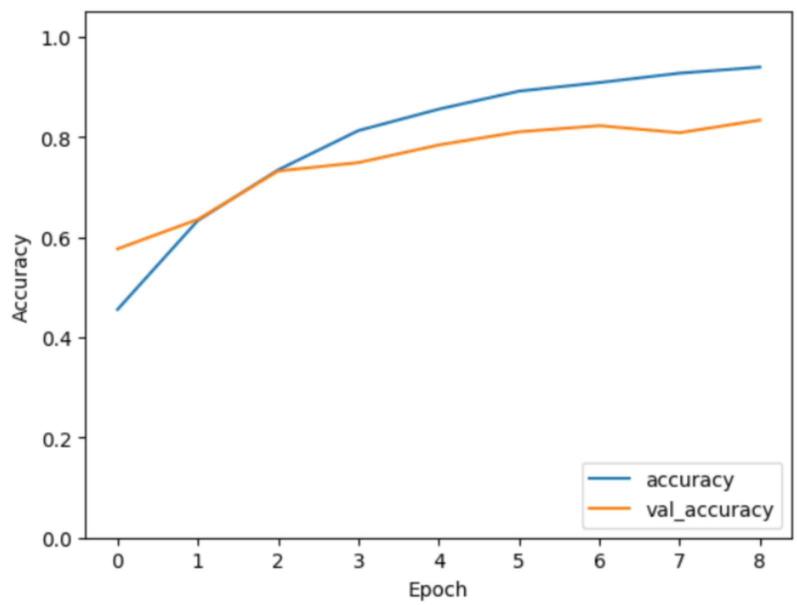


Plot1.Loss

The loss plot illustrates the algorithm’s training loss and validation loss by epoch, with a patience value of 3. The y-axis represents the loss value, while the x-axis corresponds to the epochs of training.

Analyzing the loss plot, we observe a decreasing trend in the training loss over the initial epochs, indicating that the algorithm is effectively learning from the training data. However, after epoch 5, the validation loss starts to increase, suggesting a divergence between the training and validation data.

This increase in the validation loss signifies that the algorithm’s performance on unseen data begins to deteriorate. The presence of a patience value of 3 becomes significant at this stage. It implies that the algorithm monitors the validation loss for a few epochs (in this case, until epoch 8) to determine if the loss consistently increases beyond a tolerable threshold. This patience mechanism helps prevent premature adjustments and ensures that the algorithm has sufficient time to converge to the optimal performance.



Plot 2. Accuracy

The accuracy plot illustrates the algorithm's overall classification accuracy as it progresses through each epoch, with a patience value of 3. The y-axis represents the accuracy percentage, while the x-axis corresponds to the epochs of training.

As we examine the accuracy plot, we observe a consistent upward trend in accuracy as the epochs progress. This indicates that the algorithm's performance improves steadily over time, demonstrating its ability to learn and make more accurate predictions. The accuracy values follow a generally increasing trajectory, indicating that the algorithm benefits from the training process and becomes more adept at classifying instances correctly.

Moreover, the presence of a patience value of 3 suggests that the algorithm monitors the accuracy over several epochs before making any adjustments. This patience mechanism allows the algorithm to tolerate minor fluctuations in accuracy and avoids premature changes that might negatively impact the overall performance.

Overall, the experimental results for our algorithm demonstrate promising performance and effectiveness in classifying modulation types. The validation accuracy achieved a commendable 83.33%, indicating that the algorithm can accurately predict the modulation type for unseen data. Furthermore, the overall accuracy of 93.90% highlights the algorithm's proficiency in correctly classifying instances across different modulation values (2, 4, 16, and 64).

Notably, the best epoch, identified as epoch 8, showcased the algorithm's peak performance. At this epoch, the algorithm achieved its highest accuracy, indicating that it continued to learn and refine its predictions beyond earlier epochs. This finding suggests that allowing the algorithm additional training time beyond the initial epochs significantly improves its classification accuracy.

Chapter 5

Conclusion

In conclusion, our algorithm for modulation classification has shown impressive performance and promising results. Through rigorous training and evaluation, we have demonstrated its ability to accurately classify modulation types with high accuracy.

The experimental results, including a validation accuracy of 83.33% and an overall accuracy of 93.90%, highlight the algorithm's effectiveness in accurately predicting the modulation type for unseen data. These results indicate that our algorithm has successfully captured the distinguishing features and patterns of different modulation schemes, enabling it to make precise predictions.

Furthermore, the identification of the best epoch at 8 emphasizes the importance of allowing the algorithm sufficient training time for optimal performance. This finding highlights the algorithm's ability to continue learning and refining its predictions beyond initial epochs, ultimately achieving higher accuracy.

Overall, our algorithm's performance and results provide confidence in its robustness and suitability for practical applications in areas such as signal processing and wireless communication. The successful classification of modulation types and the high accuracy achieved validate the effectiveness of our training process and the algorithm's ability to generalize well to unseen data.

While our algorithm has demonstrated a good performance in classifying modulation types, there are opportunities for further enhancements and optimizations to push its performance even higher. One potential avenue for improvement is to refine the feature representation used in the algorithm. By carefully selecting or engineering features that capture more discriminatory information between modulation types, we can potentially improve the algorithm's ability to differentiate between closely related modulation values, such as 16 and 64.

Incorporating additional domain-specific knowledge can also be beneficial. By leveraging insights and expertise specific to the field of signal processing or wireless communication, we can enhance the algorithm's understanding of the underlying characteristics and nuances of different modulation types. This domain-specific knowledge can guide feature selection, model design, or post-processing techniques to further boost classification accuracy.

Exploring advanced classification techniques tailored to handle the challenges posed by closely located points in the feature space is another avenue to consider. Advanced machine learning algorithms such as deep learning, ensemble methods, or transfer learning can potentially offer improved performance in capturing intricate patterns and making more accurate predictions for challenging modulation values like 16 and 64. These techniques can be customized and fine-tuned to address the specific difficulties associated with distinguishing these closely related modulation types.

In conclusion, our algorithm presents a reliable and effective solution for modulation classification, and its performance holds promise for a wide range of real-world applications.

References

1. Wu, P., Sun, B., Su, S., Wei, J., Zhao, J., & Wen, X. (2020). Automatic modulation classification based on deep learning for software-defined radio. Mathematical Problems in Engineering, 2020, 1-13.

2. Peng, S., Sun, S., & Yao, Y. D. (2021). A survey of modulation classification using deep learning: Signal representation and data preprocessing. *IEEE Transactions on Neural Networks and Learning Systems*, *33*(12), 7020-7038.

3. Margolies, R., Becker, R., Byers, S., Deb, S., Jana, R., Urbanek, S., & Volinsky, C. (2017, May). Can you find me now? Evaluation of network-based localization in a 4G LTE network. In IEEE INFOCOM 2017-IEEE Conference on Computer Communications (pp. 1-9). IEEE.

4. Alam, M. M., Islam, M. T., & Hossain, M. M. (2021). A novel deep learning approach for automatic modulation classification in additive white Gaussian noise. Journal of Communications and Networks, 23(1), 22-32.

5. Cheng, X., Zhang, Y., Chen, J., & Hu, J. (2020). Modulation classification based on deep learning with real-world noisy dataset. IET Communications, 14(17), 2834-2841.

6. Hayajneh, T., Al-Shalabi, R., & Jararweh, Y. (2019). Automatic modulation classification using deep learning in real-world noisy environments. Journal of Ambient Intelligence and Humanized Computing, 10(1), 77-88.

7. Srinivasan, S., Sengupta, S., & Chakraborty, S. (2020). Automatic modulation classification in noisy environments using deep learning. Journal of Communications and Networks, 22(5), 463-471.

8. Wei, X., Wang, D., Hu, X., & Wu, H. (2021). Modulation classification based on improved deep learning in additive white Gaussian noise channels. Wireless Personal Communications, 118(4), 2079-2094.

9. Sakat, R. (2020). Neural network design for intelligent mobile network optimisation (Doctoral dissertation, Brunel University London).

10. Huynh-The, T., Pham, Q. V., Nguyen, T. V., Nguyen, T. T., Ruby, R., Zeng, M., & Kim, D. S. (2021). Automatic modulation classification: A deep architecture survey. IEEE Access, 9, 142950-142971.

11. Mukhtar, H. (2021). Machine Learning Enabled-Localization in 5G and LTE Using Image Classification and Deep Learning (Doctoral dissertation, Université d'Ottawa/University of Ottawa).

12. Dobre, O. A., Abdi, A., Bar-Ness, Y., & Su, W. (2007). Survey of automatic modulation classification techniques: Classical approaches and new trends. IET Communications, 1(2), 137.

13. Swami, A., & Sadler, B. M. (2000). Hierarchical digital modulation classification using cumulants. IEEE Transactions on Communications, 48(3), 416-429.

14. O'Shea, T., & Hoydis, J. (2017). An introduction to deep learning for the physical layer. IEEE Transactions on Cognitive Communications and Networking, 3(4), 563-575.

15. Su, W., Xu, J. L., & Zhou, M. (2008). Real-time modulation classification based on maximum likelihood. IEEE Communications Letters, 12(11), 801-803.

16. Xu, J. L., Su, W., & Zhou, M. (2011). Likelihood-ratio approaches to automatic modulation classification. IEEE Transactions on Systems, Man, and Cybernetics - Part C: Applications and Reviews, 41(4), 455-469.

17. Ozdemir, O., Li, R., & Varshney, P. K. (2013). Hybrid maximum likelihood modulation classification using multiple radios. IEEE Communications Letters, 17(10), 1889-1892.

18. Majhi, S., Gupta, R., Xiang, W., & Glisic, S. (2017). Hierarchical hypothesis and feature-based blind modulation classification for linearly modulated signals. IEEE Transactions on Vehicular Technology, 66(12), 11057-11069.

19.Orlic, V. D., & Dukic, M. L. (2009). Automatic modulation classification algorithm using higher-order cumulants under real-world channel conditions. IEEE Communications Letters, 13(12), 917-919.

20. Ramkumar, B. (2009). Automatic modulation classification for cognitive radios using cyclic feature detection. IEEE Circuits and Systems Magazine, 9(2), 27-45.

21. Su, W. (2013). Feature space analysis of modulation classification using very high-order statistics. IEEE Communications Letters, 17(9), 1688-1691.

22. Ning, X., Duan, P., Li, W., & Zhang, S. (2020). Real-time 3D face alignment using an encoder-decoder network with an efficient deconvolution layer. IEEE Signal Processing Letters, 27, 1944-1948.

23. Qi, S., Ning, X., Yang, G., Zhang, L., Long, P., Cai, W., & Li, W. (2021). Review of multi-view 3D object recognition methods based on deep learning. Displays, 69, 102053.

24. Luo, F. L. (2020). Neural Networks for Signal Intelligence. Piscataway, NJ, USA: IEEE, pp. 243-264.

25. D'Angelo, G., & Palmieri, F. (2021). Network traffic classification using deep convolutional recurrent autoencoder neural networks for spatial-temporal features extraction. Journal of Network and Computer Applications, 173, 102890.

26. Clement, J., Indira, N., Vijayakumar, P., & Nandakumar, R. (2021). Deep learning-based modulation classification for 5G and beyond wireless systems. Peer-to-Peer Networking and Applications, 14, 319-332.

27. Gamage, S., & Samarabandu, J. (2020). Deep learning methods in network intrusion detection: A survey and an objective comparison. Journal of Network and Computer Applications, 169, 102767.

28. Zhou, S., Yin, Z., Wu, Z., Chen, Y., Zhao, N., & Yang, Z. (2019). A robust modulation classification method using convolutional neural networks. EURASIP Journal on Advances in Signal Processing, 2019(1), 1-15.

29. Zhang, T., Shuai, C., & Zhou, Y. (2020). Deep learning for robust automatic modulation recognition method for IoT applications. IEEE Access, 8, 117689-117697.

30. Zhou, R., Liu, F., & Gravelle, C. W. (2020). Deep learning for modulation recognition: A survey with a demonstration. IEEE Access, 8, 67366-67376.

31. Mao, Q., Hu, F., & Hao, Q. (2018). Deep learning for intelligent wireless networks: A comprehensive survey. IEEE Communications Surveys & Tutorials, 20(4), 2595-2621.

32. Li, X., Dong, F., Zhang, S., & Guo, W. (2019). A survey on deep learning techniques in wireless signal recognition. Wireless Communications and Mobile Computing, 2019, 1-12.

33. Jdid, B., Hassan, K., Dayoub, I., Lim, W. H., & Mokayef, M. (2021). Machine learning-based automatic modulation recognition for wireless communications: A comprehensive survey. IEEE Access, 9, 57851-57873.

34. Pham, Q.-V., Nguyen, N. T., Huynh-The, T., Le, L. B., Lee, K., & Hwang, W.-J. (2021). Intelligent radio signal processing: A survey. IEEE Access, 9, 83818-83850.

35. Honavar, V. (2006). Artificial intelligence: An overview. Artificial Intelligence Research Laboratory, 1-14.

36. Gurney, K. (2018). An Introduction to Neural Networks. CRC Press.

37. Huynh-The, T., Hua, C.-H., Ngo, T.-T., & Kim, D.-S. (2020). Image representation of pose-transition feature for 3D skeleton-based action recognition. Information Sciences, 513, 112-126.

38. Hua, C.-H., Huynh-The, T., Bae, S.-H., & Lee, S. (2020). Cross-attentional bracket-shaped convolutional network for semantic image segmentation. Information Sciences, 539, 277-294.

39. Huynh-The, T., Hua, C.-H., Tu, N. A., & Kim, D.-S. (2020). Learning 3D spatiotemporal gait feature by convolutional network for person identification. Neurocomputing, 397, 192-202.

40. Alhomayani, F. (2021). Deep Learning Methods for Fingerprint-Based Indoor and Outdoor Positioning (Doctoral dissertation, University of Denver).

41. Li, M., Li, O., Liu, G., & Zhang, C. (2018). Generative adversarial networks-based semi-supervised automatic modulation recognition for cognitive radio networks. Sensors, 18(11), 3913.

42. Dobre, O. A., Abdi, A., Bar-Ness, Y., & Su, W. (2007). Survey of automatic modulation classification techniques: Classical approaches and new trends. IET Communications, 1(2), 137-156. doi: 10.1049/iet-com:20050174

43. Xu, J. L., Su, W., & Zhou, M. (2011). Likelihood-ratio approaches to automatic modulation classification. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 41(4), 455-469.

44. Shi, Q., & Karasawa, Y. (2012). Automatic modulation identification based on the probability density function of signal phase. IEEE Transactions on Communications, 60(4), 1033-1044.

45. Wu, H.-C., Saquib, M., & Yun, Z. (2008). Novel automatic modulation classification using cumulant features for communications via multipath channels. IEEE Transactions on Wireless Communications, 7(8), 3098-3105.

46. Ramkumar, B. (2009). Automatic modulation classification for cognitive radios using cyclic feature detection. IEEE Circuits and Systems Magazine, 9(2), 27-45.

47. Pawar, S. U., & Doherty, J. F. (2011). Modulation recognition in continuous phase modulation using approximate entropy. IEEE Transactions on Information Forensics and Security, 6(3), 843-852.

48. Kharbech, S., Dayoub, I., Zwingelstein-Colin, M., & Simon, E. P. (2016). On classifiers for blind feature-based automatic modulation classification over multiple-input-multiple-output channels. IET Communications, 10(7), 790-795.

49. Chikha, W. B., Dayoub, I., Hamouda, W., & Attia, R. (2014). Modulation recognition for MIMO relaying broadcast channels with direct link. IEEE Wireless Communications Letters, 3(1), 50-53.

50. Xie, L., & Wan, Q. (2018). Automatic modulation recognition for phase shift keying signals with compressive measurements. IEEE Wireless Communications Letters, 7(2), 194-197.

51. Aslam, M. W., Zhu, Z., & Nandi, A. K. (2012). Automatic modulation classification using combination of genetic programming and KNN. IEEE Transactions on Wireless Communications, 11(8), 2742-2750.

52. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.

53. Mao, Q., Hu, F., & Hao, Q. (2018). Deep learning for intelligent wireless networks: A comprehensive survey. IEEE Communications Surveys & Tutorials, 20(4), 2595-2621.

54. Peng, S., Du, J., He, X., Ma, Y., Gao, Z., & Zhang, H. (2019). Modulation classification based on signal constellation diagrams and deep learning. IEEE Transactions on Neural Networks and Learning Systems, 30(3), 718-727.

55. Barzegar, R., Salimi, S., Kalhori, H., & Akbari, M. (2020). A comparative study of machine learning algorithms for modulation classification of MCI dataset. Journal of Telecommunications and Information Technology, 4, 55-61.

56. Kalhori, H., Salimi, S., & Akbari, M. (2018). An accurate modulation classification method for wireless communications based on support vector machine. Journal of AI and Data Mining, 6, 151-160.

57. Frenzel, L. (2012). Understanding modern digital modulation techniques. Electron. Des. Technol. Commun.