An autoencoder-based indoor localization approach to reduce the class imbalance problem in fingerprint data

A thesis submitted in partial fulfillment of the requirement for the degree of
Master of Engineering
in
Computer Science and Engineering

Submitted by
Saba Nadim
Registration No.: 154181 of 2020-2021,
Examination Roll No.: M6TCT23018
Session: 2020-2023

Under the Supervision of
Dr. Chandreyee Chowdhury
Department of Computer Science and Engineering
Jadavpur University,
188, Raja S.C. Mallick Rd,
Kolkata - 700032,
West Bengal, India

FACULTY OF ENGINEERING AND TECHNOLOGY
JADAVPUR UNIVERSITY

Certificate of Recommendation

This is to certify that this is a bonafide record of the project entitled **"An autoencoder-based indoor localization approach to reduce the class imbalance problem in fingerprint data"**, submitted by Saba Nadim (University Registration No.: 154181 of 2020-2021, Examination Roll No.: M6TCT23018) is hereby approved of a creditable study of a technological subject carried out under my supervision and presented in a manner satisfactory to warrant its acceptance for partial fulfillment of the requirements of the degree of Master of Engineering in Computer Science and Engineering. The research results presented in the thesis have not been included in any other paper submitted for the award of any degree in any other university or institute.

Supervisor

. . . .

Dr. Chandreyee Chowdhury
Dept. of Computer Science & Engineering
Jadavpur University, Kolkata-32, India

Countersigned

. . . .

Prof. Nandini Mukhopadhyay
Head, Dept. of Computer Science & Engineering
Jadavpur University, Kolkata-32, India

. . . .

Prof. Ardhendu Ghoshal
Dean, Faculty of Engineering and Technology
Jadavpur University, Kolkata-32, India

FACULTY OF ENGINEERING AND TECHNOLOGY
JADAVPUR UNIVERSITY

Certificate of Approval

This is to certify that the thesis entitled **"An autoencoder-based indoor localization approach to reduce the class imbalance problem in fingerprint data"**, is a bonafide record of work carried out by Saba Nadim in partial fulfillment of the requirements of the degree of Master of Engineering in Computer Science and Engineering in Department of Computer Science and Engineering, Jadavpur University during the period of September 2020 to June 2023. It is understood that by this approval the undersigned do not necessarily endorse any statement made, opinion expressed or conclusion drawn therein but approve the thesis only for the purpose for which it has been submitted.

. . . .
Signature of Examiner 1
Date:

. . . .
Signature of Examiner 2
Date:

FACULTY OF ENGINEERING AND TECHNOLOGY
JADAVPUR UNIVERSITY

Declaration of Originality and Compliance with Academic Ethics

I hereby declare that this thesis entitled "**An autoencoder-based indoor localization approach to reduce the class imbalance problem in fingerprint data**" contains a literature survey and original research work by the undersigned candidate as part of his Degree of Master of Engineering in Computer Science and Engineering.

All information has been obtained and presented under academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Name: Saba Nadim

Registration No.: 154181 of 2020-2021

Examination Roll No.: M6TCT23018

Thesis Title: **An autoencoder-based indoor localization approach to reduce the class imbalance problem in fingerprint data**

. . . .
Signature of Student with date

ACKNOWLEDGEMENT

The success and outcome of this thesis required a lot of guidance and assistance from many people and I am extremely privileged to have got this all along the completion of my project. All that I have done is only due to such supervision and assistance and I would not forget to thank them.

I owe my deep gratitude to my mentor and thesis guide **Dr. Chandreyee Chowdhury**, who took a keen interest in my thesis work and guided me all along till the completion of my thesis work by providing all the necessary information for developing a good end-to-end solution. I am extremely thankful to him for his assistance and dedicated involvement in every step throughout the process, despite the devastating COVID-19 epidemiological situation. His constant suggestions and editorial enhancements have been an everlasting source of motivation and inspiration to me. Working under a person with eternal wisdom yet kind, a simplistic attitude has been an honour.

I would also like to thank **Prof. Nandini Mukhopadhyay, Head of the department**, Department of Computer Science and Engineering, at Jadavpur University, for extending his help to complete this thesis.

Besides my supervisor, I would also like to thank **Mrs. Manjarini Mallik** for their constant support, encouragement, insightful comments, and research expertise, which helped me the most. Without their enthusiasm, support and continuous optimism, this thesis wouldn't have progressed much.

Most importantly, none of this could have happened without the love and support of my family. To **my father, Mr. Md Yearaf Hossain, my mother, Mrs. Hasnaara Begam** and **my brother Mr. Sahil Nadim** – it would be an understatement to say that their unconditional love and encouragement have always helped me in my need. Without their forbearance and whole-hearted support, this thesis would not have been able to see the light of day.

Finally, I would like to thank all my Professors at Jadavpur University, my classmates, especially my guide **Dr. Chandreyee Chowdhury,** and the many well-wishers who have been a constant source of help and inspiration to me during my thesis work. My thanks and appreciation to all of them for being a part of this wonderful journey and making this possible.

. . . .

Saba Nadim
Examination Roll No.: M6TCT23018
Dept. of Computer Science & Engineering
Jadavpur University
Kolkata, India

Abstract

The demand for indoor localization services for indoor environments, and indoor positioning based on fingerprinting has attracted significant interest due to its high accuracy. Working with data having class imbalance has led to biased localization results, with lower accuracy. In this paper, we presented a class imbalance problem solution using an autoencoder model with RSSI fingerprint data. The autoencoder architecture includes two phases. In the first stage, the autoencoder is trained using a large dataset of fingerprint samples, irrespective of their location labels. In the second phase, the data generated through the autoencoder are augmented with the original data. Further, we used two mathematical approaches such as KL- Divergence and Euclidean Distance, to evaluate the localization and checked the accuracy with classifiers, such as a KNN, SVM and RF. Experimental results are presented to ensure that the autoencoder-based indoor localization approach offers a promising solution to mitigate the class imbalance problem in fingerprint data by effectively reducing location error.

Table of Content

Chapter 1

Introduction

Object, person, and asset detection or navigation in a closed environment have been essential in modern life. An indoor localization system has made it possible, for users to precisely locate things, persons, or assets inside a building using cell phones, mobile devices, tracking tags, or other devices. To detect, a device's position, the system can make use of several RSSI Wi-Fi signals and the strength of those signals. The position can be approximated, by connecting the signal strengths of many Wi-Fi access points spread across the indoor area. Fingerprint-based indoor localization with WiFi data is used to examine the distinctive features of WiFi signals to pinpoint a user's, object's location inside an indoor environment [1]. It is predicated on the idea that WiFi signals behave differently depending on the environment due to elements like signal strength, interference, and pattern of signal transmission. The method entails compiling a database of signal fingerprints that represent the various indoor locations' distinct WiFi signal patterns. This is achieved through an offline training phase in which WiFi signal data from several access points (APs) located across the environment is gathered. Compared to GPS signals it can be stated that GPS relies on satellite signals, and these signals can be significantly attenuated or completely blocked in indoor spaces [2]. On the contrary, WiFi signals are readily available in most indoor environments as WiFi access points are commonly installed indoors. This makes fingerprint-based WiFi data more suitable for indoor localization where GPS signals may not be accessible. GPS primarily provides two-dimensional (latitude and longitude) positioning information, which is insufficient for multi-floor indoor environments.

Fingerprint-based WiFi data can be utilized to achieve vertical positioning as well, enabling localization on different floors within a building.

1.1 Applications of ILS

There are several application extents of indoor localization and specific real-time applications discussed below.

Localization, Tracking and Navigation - There are many objectives of ILS such as tracking down the static position of any object and monitoring the real sequence of positions of objects that are in motion. It navigates gently under indoor environments such as buildings, malls, universities etc. Moreover, the viability of a WiFi-based indoor positioning system for construction sites is presented by [3]. The technique was developed for monitoring construction sites and tracking the locations of construction resources, including labour, materials machinery, and vehicles using the fingerprint process of Received Signal Strength Indication (RSSI) from the Access Point (AP). One of the indoor localization systems, which is based on RFID technology and a hierarchical structure of classifiers, can localize patients in a hospital [4]. Further, it also has an application, an effective and wearable mobility aid for people who are suffering from a visual disability, completely established on 3D computer vision and machine learning techniques [5] etc.

Tracking Assets and Monitoring Warehouse - To achieve cost-effective asset management, real-time indoor localization for asset tracking through Bluetooth location-based indoor positioning system technology has been applied [6]. Further, indoor locating technology analyzes the real-time locating and usage of RFID radio frequency identification technology and helps in monitoring warehouse commodities [7].

Autonomous Vehicle Navigation - Indoor Localization systems including the concept of 2D wireless indoor positioning helps to navigate autonomous vehicles based on Radio Frequency Identification (RFID) technology [8]. Recently in research, it has shown that a localization scheme for drones is called PILOT (High-Precision Indoor Localization for Autonomous Drones), which is specifically designed for indoor environments [9]. PILOT responses on ultrasonic acoustic signals to evaluate the drone's target location. Flying autonomously, drones need to be informed of their location constantly. Based on their current position and the final destination, navigation commands will be generated and drones will be guided to their destination. Localization can be easily conducted in outdoor environments, through GPS signals and drone inertial measurement units (IMUs). Other research has shown that UWB localization paired with inertial-based dead reckoning algorithms is more accurate and reliable in tracking and navigating robots in indoor environments [10]. The robots are equipped with an inertial measurement unit. Multiple UWB anchors are embedded in a closed environment and communicated within the anchor network. At the same time, the UWB tag on the robot overhears the communication and locates itself within the building using Time Difference of Arrival (TDoA) calculations.

1.2 Motivation

Although fingerprint-based localization is a promising way for indoor localization, it suffers from the challenge of repetitive site surveys. Collecting data from each location point is a tedious task. Data needs to be collected during a long period, to cover the dynamicity of the data over time. Every location point may not be accessible every time, thus resulting in a class imbalance in the dataset. Class imbalance often affects the performance of ML models. In general, ML classifiers typically have more bias towards the major class and cause bad classification of the minor classes, which in turn reduces the system's accuracy. This work is aimed to address this research challenge by analyzing the dataset's sample distribution concerning class labels and augmenting synthetic fingerprint data for the classes with lower sample counts, generated by the autoencoder model.

1.4 Contribution

In this work, our contributions are as follows.

– An algorithm has been proposed based on an autoencoder to generate synthetic data for the location points with the lower number of sample counts to reduce the class imbalance in localization datasets.

– Similarity between original and generated data is checked using *KL-Divergence* and Euclidean distance measurements.

– The proposed algorithm is validated through experiments on real-life benchmark datasets.

1.5 Scope of the Work

The autoencoder-based indoor localization approach has an extent to reduce the class imbalance problem in fingerprint data. The research work has been completed, with the designing of a deep neural network architecture (Autoencoder). The experiment can be performed both in large and small datasets It also integrates different mathematical algorithms such as *KL-Divergence* and *Euclidean-Distance* to calculate a symmetrical measure of the relative difference between two probability distributions. Further helps to make localization accuracy more accurate.

Chapter 2

Related Work

2.1 Machine Learning Framework for ILS

The machine learning framework, for Indoor Localization, is mentioned below;

Data Collection - The very first step was data collection, where we gathered two datasets such as - ***1.*** *JU-IndoorLoc* and ***2.*** *UJI-IndoorLoc* datasets. For the *JU-IndoorLoc* datasets RSSI signal values were collected from every possible training point of the university building of Jadavpur Building. With the help of a mobile application, data were collected from the RSSI signal of WiFi access points (APs). Besides that *UJI-IndoorLoc* dataset was downloaded from the internet resources.

2.2 Survey of deep learning for ILS

Autoencoder is an autonomous artificial neural network which determines how to compress and encode data efficiently and then learns how to rebuild the data again from the diminished encoded portrayal to a representation that is as similar to the original input as possible [11]. An autoencoder network is integrated into a pair of two connected networks such as an encoder and a decoder. The process initiates with the encoder compressing the original data. Then, the decoder decompresses the data to regenerate data as close as possible to the original data input. Researchers have used autoencoders in the domain of indoor localization for various purposes. Wi-Fi-based indoor localization has attracted great interest due to its ubiquitous access in many indoor environments. Although the accuracy deteriorates in complex indoor propagation environments, that results in variable received signal strength (RSS). Here autoencoder improves the accuracy of indoor localization by omitting the noisy signal RSS. The **AutLoc** system includes both an offline training phase and an online localization phase. In the offline training phase, the deep autoencoder is edified to denoise the measured data and then build the RSS fingerprints according to the trained weights. For the online localization phase, three machine learning algorithms are adopted, which are random forest regression, multi-layer perceptron classification and multi-layer perceptron regression, to estimate the location. Averaging over the results of three algorithms, we then obtain the final estimated location. Simulation results justify the superiority of the proposed AutLoc system over previous indoor location schemes in vast scenarios [12]. In another research, RSS measurements from available transmitter sensors are used in the fingerprint localization method work based on received signal strength (RSS) in wireless sensor networks [13]. This method collects data from electronic gadgets with internal sensors and implements a novel algorithm that takes advantage of deep learning and extreme learning machines. With the help of an autoencoder, it extracts high-level features that help in improving the localization execution in the feature extraction and the classification [14]. Another work has proposed the development of a semi-supervised deep learning method that will be able to train a prediction model from a small set of annotated Wi-Fi observations and a massive set of non-annotated ones [15]. The method is based on the variational autoencoder deep network that is complemented with an additional component of structural projection that will be able to further improve the localization accuracy in a complex, multi-building and multi-floor environment. The author considered several different network compositions which combine the classification and regression subtasks to achieve optimal performance. The evaluation of the method was accomplished on the public IndoorLoc dataset. It was found to maintain state-of-the-art localization accuracy with a very limited amount of annotated data.

2.3 Comparison of DL Techniques

To learn from and generate predictions from complicated data, a subsection of machine learning called deep learning focuses on training artificial neural networks with several layers. Here are a few prevalent deep-learning strategies:

1. Convolutional Neural Networks (CNNs): CNNs are widely used in computer vision tasks, such as image classification, object detection, and image segmentation [16]. They leverage convolutional layers to automatically learn hierarchical representations of input data, capturing local patterns and spatial relationships.

2. Recurrent Neural Networks (RNNs): RNNs are made to handle sequential data, such as time series or text. They can capture temporal interdependence and model context over time. Researchers have developed a pavement deterioration prediction model using RNNs in which rutting depth was predicted utilizing periodic survey data ranging from 1987 to 2016 [17]. They found that the RNN model produced higher prediction accuracy than the conventional linear multiple regression and multi-layer perceptron methods. Popular RNN variations that deal with the vanishing gradient problem include Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU).

3. Generative Adversarial Networks (GANs): A generator and a discriminator are the two neural networks that makeup GANs. While the discriminator tries to tell the difference between actual and fake data, the generator aims to produce artificial data that mimics real data. GANs have been used to create images, enhance data, and transfer styles with success [18].

4. Autoencoders: Neural networks that have been taught to reconstruct their input data are autoencoders [19]. They are made up of an encoder that converts the input into a representation in a lower-dimensional latent space and a decoder that retrieves the input from the latent space.

For applications including data denoising, dimensionality reduction, and anomaly detection, autoencoders are employed.

5. Transfer Learning: Transfer learning involves leveraging pre-trained deep learning models trained on large datasets and fine-tuning them for specific tasks or domains with limited data [20]. By using pre-trained models as a starting point, transfer learning helps in improving performance and reducing the need for extensive training data.

6. Reinforcement Learning: Although not exclusive to deep learning, deep reinforcement learning combines deep neural networks with reinforcement learning algorithms. Agents learn to make sequential decisions by interacting with an environment and receiving rewards or penalties. Deep Q-Networks (DQNs) and Proximal Policy Optimization (PPO) are popular deep reinforcement learning algorithms [21].

7. Attention Mechanisms: Attention mechanisms are used to focus on relevant parts of the input during processing [22]. They have been widely adopted in natural language processing tasks, such as machine translation and text summarization. Attention helps models weigh different input elements based on their importance, improving performance on complex tasks.

8. Transformers: Transformers are a type of deep learning architecture that utilizes attention mechanisms. They have revolutionized natural language processing and achieved state-of-the-art performance in tasks like language translation, sentiment analysis, and question-answering [23]. Transformers allow for the parallel processing of input sequences and capture long-range dependencies effectively.

These are just a few examples of deep learning techniques. The field is rapidly evolving, and researchers continue to develop new models, architectures, and approaches to tackle a wide range of complex problems.

2.4 Summary

Each DL technique has its strengths and is suited to specific problem domains. CNNs excel at visual tasks, RNNs are ideal for sequential data, GANs generate synthetic samples, Transformers are powerful for natural language processing, and RL enables learning through interaction. Researchers and practitioners choose the technique based on the requirements and objectives of the task at hand.

Chapter 3

Proposed Methodology

3.1 Problem Definition

Collecting a large number of RSSI signal values, for any specific indoor location is challenging, due to long time-consuming, environmental hindrances, the erratic behaviour of signal propagation and random variations of RSSI samples for any fixed location. Wifi positioning indoors is difficult because the relation between Received Signal Strength (RSS) and the position does not follow any specific parametric forms. Even large RSSI signal values reflect a class imbalance when there are many more instances of some specific classes'(majority classes) than others. In such circumstances, standard classifiers tend to be overwhelmed by the large classes and ignore the small ones. In our scenario, for any specific location $X_{i,k}$ where ***i*** is the class label and ***k*** the number of instances or samples is not the same for ***n*** number of access points. So, to construct a solution to this class imbalance problem, we proposed a methodology that is Autoencoder-based Data Augmentation of RSSI fingerprinting for Indoor Localization.

3.2 Autoencoder-Based Framework

To reduce class imbalance present in a dataset, at first we compute the number of samples for every location. The sample count ***m*** which is most common among all the locations is the target sample count. We aim to augment data samples to each location so that its sample count becomes equal or equal to ***m.***

An autoencoder-based framework has been proposed in this work to generate label-specific synthetic data having distributions similar to the original data. To deal with the class imbalance problem, we select a particular sample count $\boldsymbol{x}$ and consider all the locations having sample count $\boldsymbol{x}$. RSSI vectors of all such locations form the input data for AE.

To understand the complex and crucial pattern among the labels and RSSI vectors, The AE has been trained with the required number of layers and the necessary numbers of neurons in each layer. The structure of the AE is represented in *Fig1*. The AE has a total 7 number of layers, among which fast three layers act as an encoder, and the last three layers act as a decoder. The innermost layer, which is the middle layer is known as the bottleneck layer, here the features are compressed maximum. The structure of AE is designed in such a way, that the intensity of compression is equal, from each previous to the next layer. To obtain this, for each dataset, the total number of features $\boldsymbol{n}$ is divided by 4 since both the encoder-to-bottleneck and bottleneck-to-decoder have 4 layers each. The first encoding layer contains $\boldsymbol{n}$ neurons. After that in each layer $\frac{n}{4}$ neurons get reduced from their corresponding previous layer, up to the bottleneck layer. After that starting from the decoding layer $\frac{n}{4}$ neurons are added in each layer. As an output, the output layer contains n neurons generating data having $\boldsymbol{n}$ attributes and \ the number of samples as input data. After we get the unlabelled data generated by the AE, the completed data is divided into $\boldsymbol{b}$ batches each containing $\boldsymbol{x}$ samples. For $\boldsymbol{i}$ in the range 1 to $\boldsymbol{b}$ the similarity between the original data of label $\boldsymbol{i}$ and the generated data of batch i is investigated with *KL-divergence*. The detailed findings of the results are mentioned in Chapter 5.

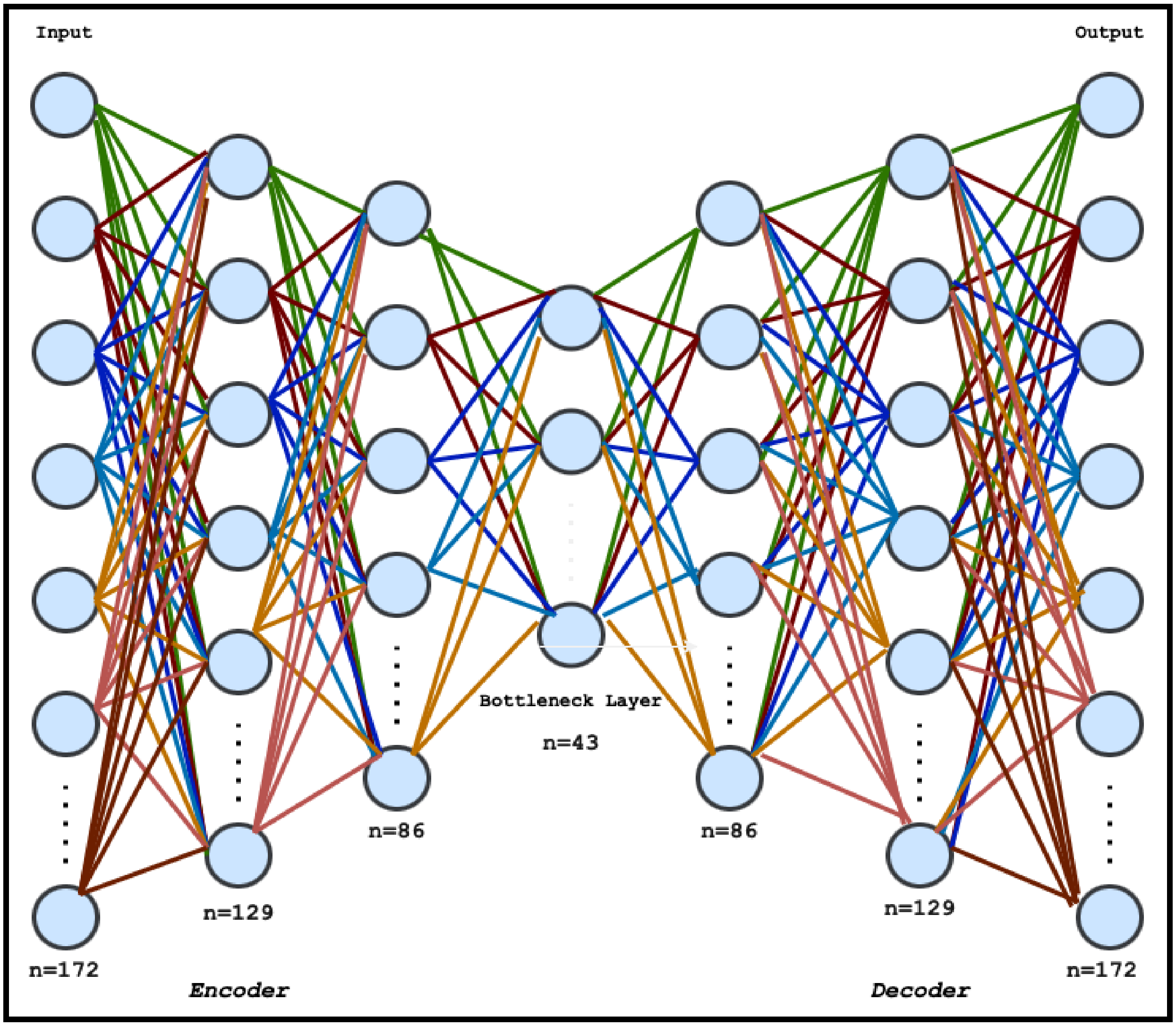

Fig 1:- The Architecture of the Autoencoder

3.3 Algorithm

Training Phase

1. **Input:** Raw Input Data ie; C_i the number of classes, C_k number of samples, and *n* number of AP.

   1.1 Label-wise sample count. (C_i number of classes for C_k max and *n* number of AP).

   C_i -> $C_{k_{max}}$ for *n* no AP

   1.2 Normalization of Dataset. (x the values for all *l* number of corresponding locations)

   x_{min} —— min(x1, x2, ..., xn)

   x_{max} —— max(x1, x2, ..., xn)

   $x_{l\ normalized}$ ———— (x*l* - min_val) / (max_val - min_val)

   1.3 **Autoencoder:**

   1.4 Denormalization of Dataset.

   x_{min} —— min(x1, x2, ..., xn)

   x_{max} —— max(x1, x2, ..., xn)

   $x_{l\ Denormalized}$ ———— (x*l*_normalized * (max_val - min_val)) + min_val

   1.5 Kl-Divergence: To assign a label to the generated dataset.

   1.6 Euclidean Distance: To assign a label to the generated dataset.

   1.7 Data Augmentation (Original Dataset+Generated Dataset)

2. **Classifier:-** Checking the accuracy to predict location from Augmented and Test data.

3.4 Workflow

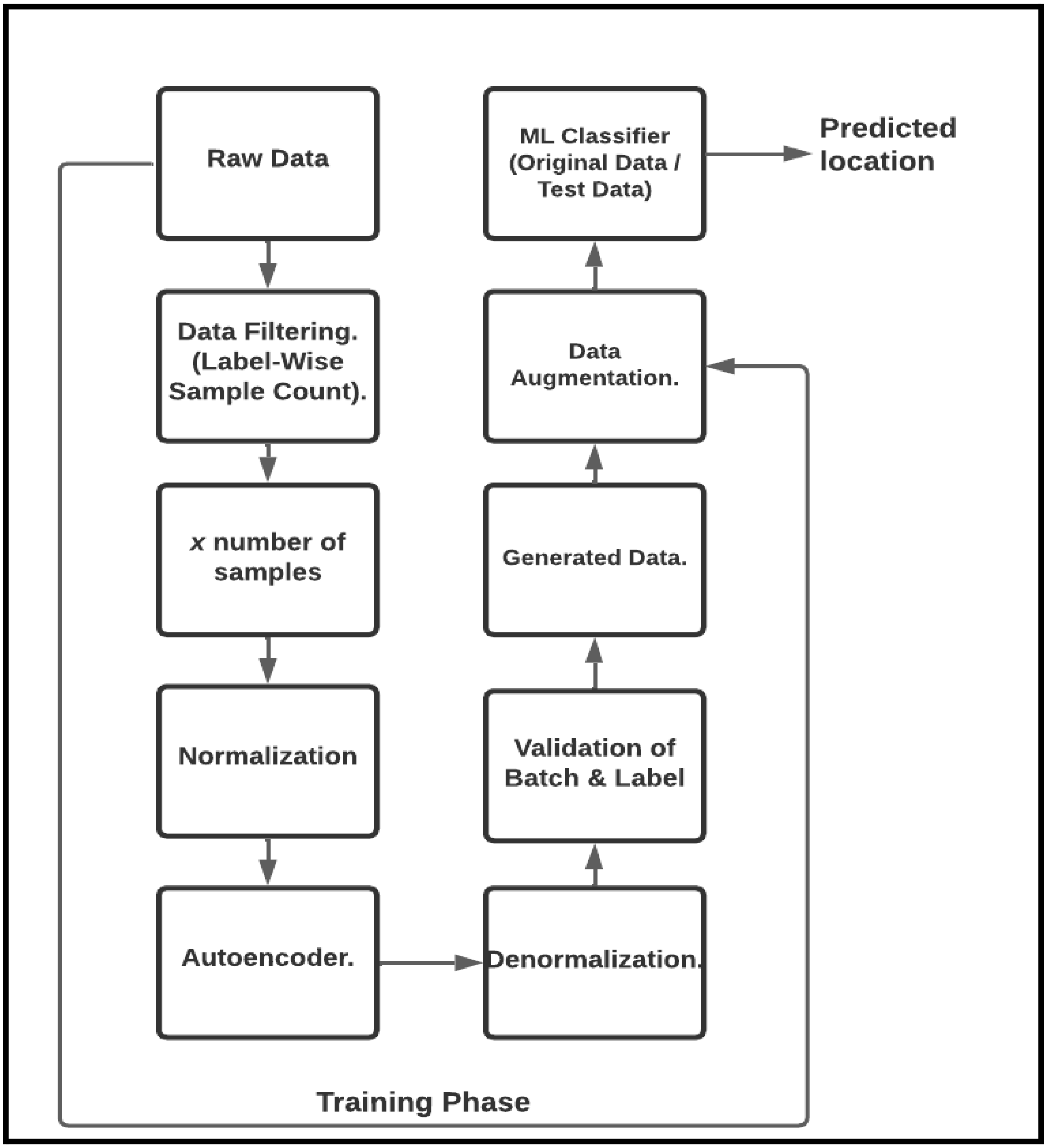

Fig2:- Workflow Model

Before augmenting the dataset we used two statistical measurement theories, to calculate the distance of the location point between the input dataset and the generated dataset. The two theories are 1) *KL-Divergence* and 2) *Euclidean Distance.*

- *KL - Divergence:* The relative statistical evaluation that is often used to quantify the difference between the location point of the input dataset and the generated dataset distribution [24]. Let us understand the equation below;

$$D_{KL}\ \boldsymbol{(p(x)\ |\ |q(x)) = \sum_{x \in X} p(x)\ ln}\ \frac{p(x)}{q(x)}\ \textbf{--------- } \boldsymbol{eq1} \qquad \textbf{(Joyce, 2011)}$$

If, D is the dataset which is having the values of KL Divergence of the location points, then p(x) is the location point probability distribution of the input dataset and q(x) is the probability distribution of the generated dataset. Though KL Divergence measures the distance between two distributions, it is not a distance measure. It is because KL Divergence is not a metric measure. It is not symmetrical: the KL from input p(x) to generated q(x) is generally not the same as the KL from generated q(x) to input p(x).

- *Euclidean Distance:* The length of a line segment connecting two location points in a dataset is the Euclidean distance between them [25]. Let's examine the following equation.;

$$D_{Eu}\ \boldsymbol{(p, q)} = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2}\ \textbf{--------- } \boldsymbol{eq2} \quad \textbf{(Kannoujia et al., 2022)}$$

Here p denotes the location points probability of the input dataset and q denotes the locations points probability, of the generated dataset. It depicts the distance between any two points on the real line as the absolute value of the numerical difference of their coordinates of, their absolute difference in the dataset. Thus p and q are on a real line.

Summary:- The methodology helps in reducing the class imbalance using data augmentation. Throughout collecting the raw input and filtering it by taking the max samples C_k for the label C_i.

The method consists of two-phase the training phase and the other location prediction phase. In the autoencoder, the training phase reduces the number of neurons and reconstructs the data. Finally, the data are augmented and checked for location estimation with the test data.

Chapter 4

Dataset Details

4.1 JU-IndoorLoc

The *JU-IndoorLoc*[1][31] the dataset used in the experiment *RSSI* value has been collected with cell grid sizes of 1m×1m from three floors of our University building using an Android application such as the 3rd, 4th and 5th floors. JUIndoorLoc dataset, containing 25,364 samples, is divided into two datasets: Such as 1) Training_Data which consists of 23,904 samples and Test_Data 1,460 samples. Both the dataset consist of have177 attributes including 172 AP.

4.2 UJI-IndoorLoc

The *UJI-IndoorLoc*[2][32] dataset was the official database used in the IPIN2015 competition. The dataset is having 529 attributes including 520 AP. The Universitat Jaume has three buildings with four or more floors and roughly 110.000 metres square total integrated into the UJIIndoorLoc dataset. It can be applied for classification, such as identifying actual buildings and floors, or regression, such as estimating actual longitude and latitude. It was developed in 2013 using 25 Android smartphones and more than 20 distinct users.

[1] https://drive.google.com/open?id=1_z1qhoRIcpineP9AHkfVGCfB2Fd_e-fD
[2] https://www.kaggle.com/datasets/giantuji/UjiIndoorLoc

Chapter 5

Experimental Results & Discussions

The result of the methodology for different datasets are mentioned below. The impact for different datasets varies due to their other parameters such as sample instances of the labels, number of AP and overall input shape. Further, the Autoencoder is tuned according to the input of the datasets, from assigning the neuron in the hidden layer to the bottleneck.

5.1 Set Up

To complete the experiment, the two datasets were used ie; *JU-IndoorLoc and UJI-IndoorLoc.* The *JU-IndoorLoc* dataset used in the experiment was collected, through the mobile application in the building of Jadavpur University. We filtered the data and selected the 4th-floor dataset whose label instance was 8. The dataset consists of 51 labels and 172 AP for training the Autoencoder model.

We trained the *UJI-IndoorLoc* data set was the official database used in the IPIN2015 competition. The dataset consists of 520 AP and for training the autoencoder our Train data was the label whose sample instance was 10 and we found 6 labels to have 10 instances. Our Test data was the label whose instance count was 30, we found that 6 labels have 30, so we took 5 samples from every 6 labels.

5.2 Convergence Analysis

Fig 3 & *Fig* 4 show our model's training loss and validation loss. It also points out how well the Autoencoder model has fitted the training data, further the validation loss demonstrates how well the model fits new data.

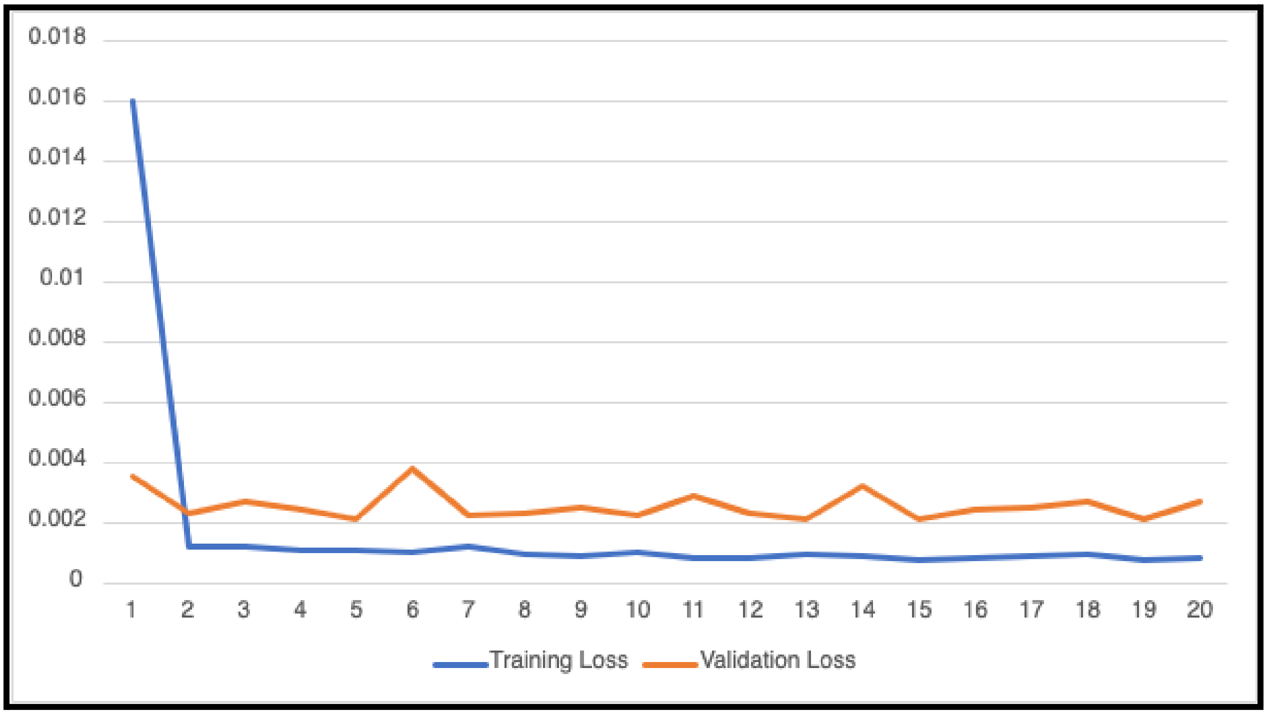

Fig 3: Training Loss & Validation Loss for JU-IndoorLoc Data

Fig 3 is the graphical representation of both Training Loss and Validation Loss of the Autoencoder model that has been trained, with the *JU-IndoorLoc* dataset. Here we can see that the model has been trained, with 20 Epochs and the training loss has gradually decreased within every epoch, which indicates, there are fewer errors in the model and the model fits the training data effectively. Further validation loss has been also gradually low, but that is comparatively higher than the training loss, which resembles the model that has fitted well and can generalize on new data.

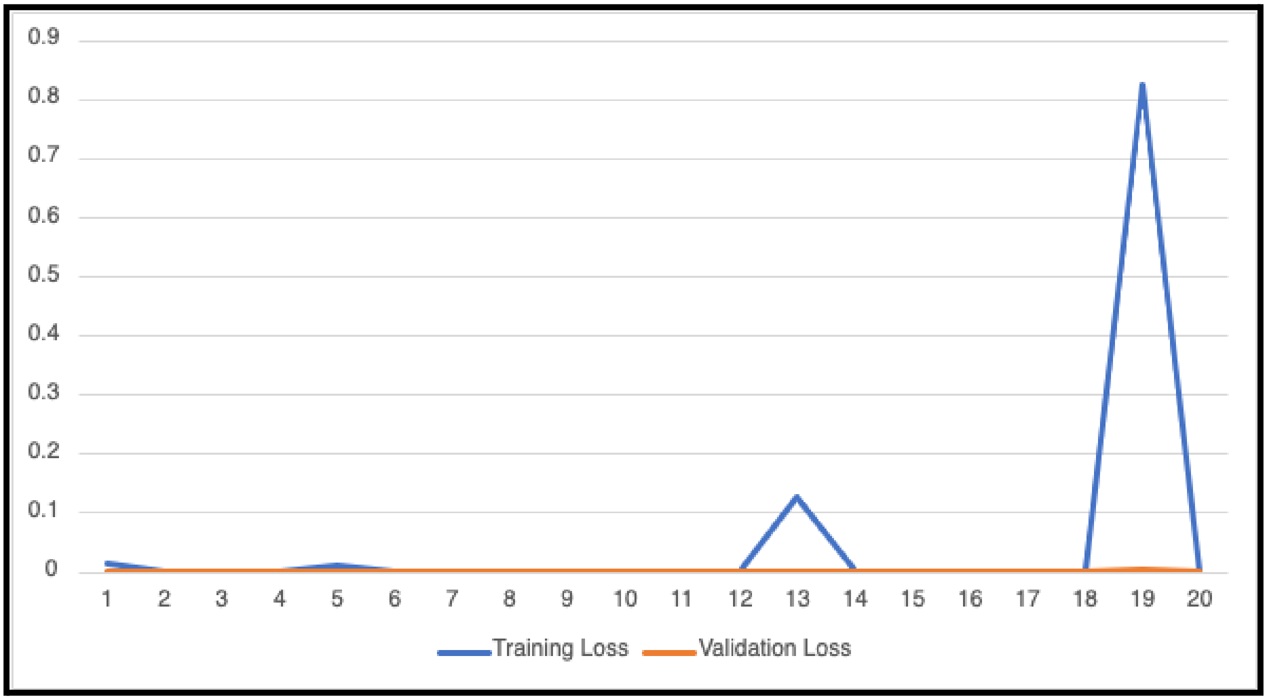

Fig 4: Training Loss & Validation Loss for UJI-IndoorLoc Dataset

Fig 4 is the graphical representation of both Training Loss and Validation Loss of the Autoencoder model that has been trained with the *UJI-IndoorLoc* dataset. Here we can see similar findings that the model has been trained with 20 Epochs and the training loss has gradually decreased within every epoch, which indicates there are fewer errors in the model and the model fits the training data effectively. Further validation loss has also gradually been low but that is comparatively higher than the training loss which resembles the model, has fitted well and can generalize on new data.

5.3 Handling Class Imbalance

To handle the class imbalance problem, we have used the autoencoder to augment the data and later calculate the *KL-Divergence* and Euclidean distance to its corresponding original input data. In the beginning accuracy of both dataset are evaluated with the classifier. And further by completing data filtering, and augmenting to label assigning the whole process is completed. Finally, the accuracy of the generated dataset is also checked. values for the different classifiers are presented below in Table 1.

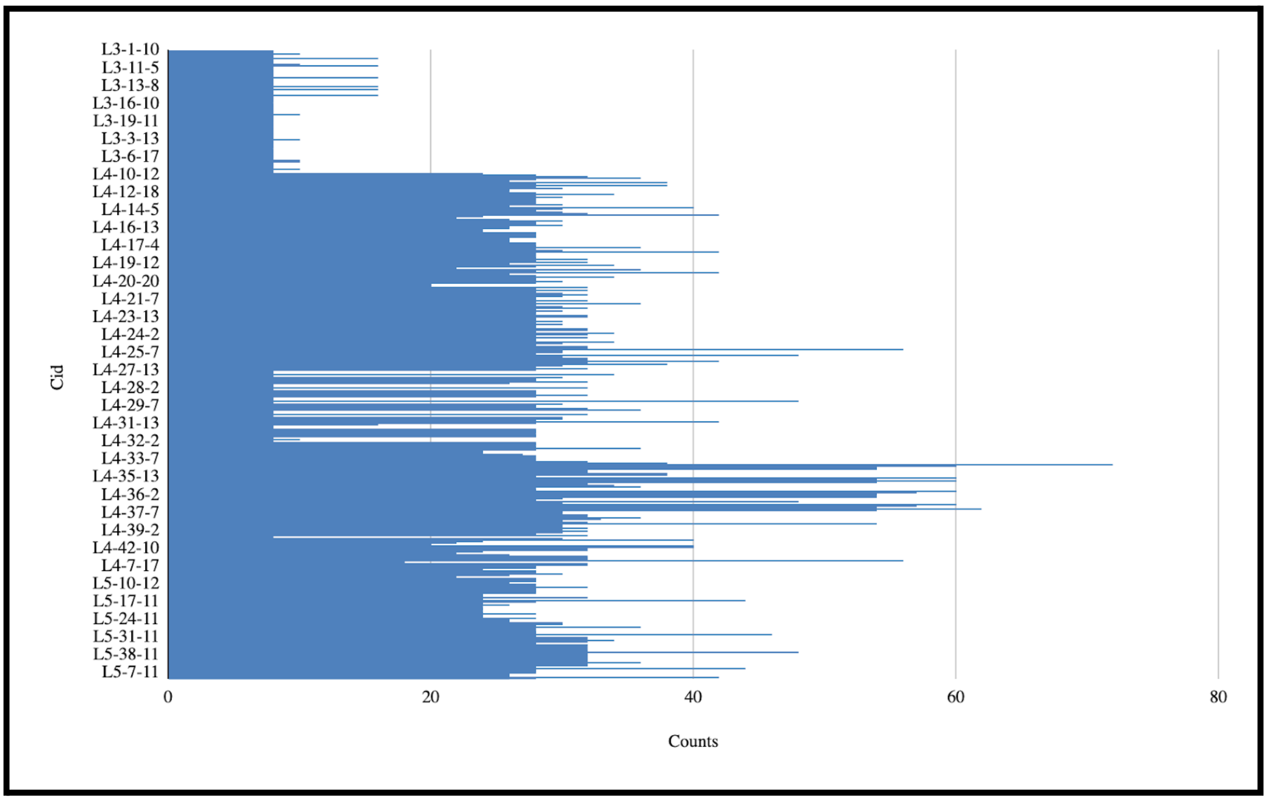

Fig 5:- Counts of class instances of JU-IndoorLoc Dataset for class imbalance.

Fig 5 is the graphical representation of instances counts of the classes of the JU-IndoorLoc dataset. Here we find that every class is having a different number of instance counts, which clearly states, that the dataset is having class imbalance.

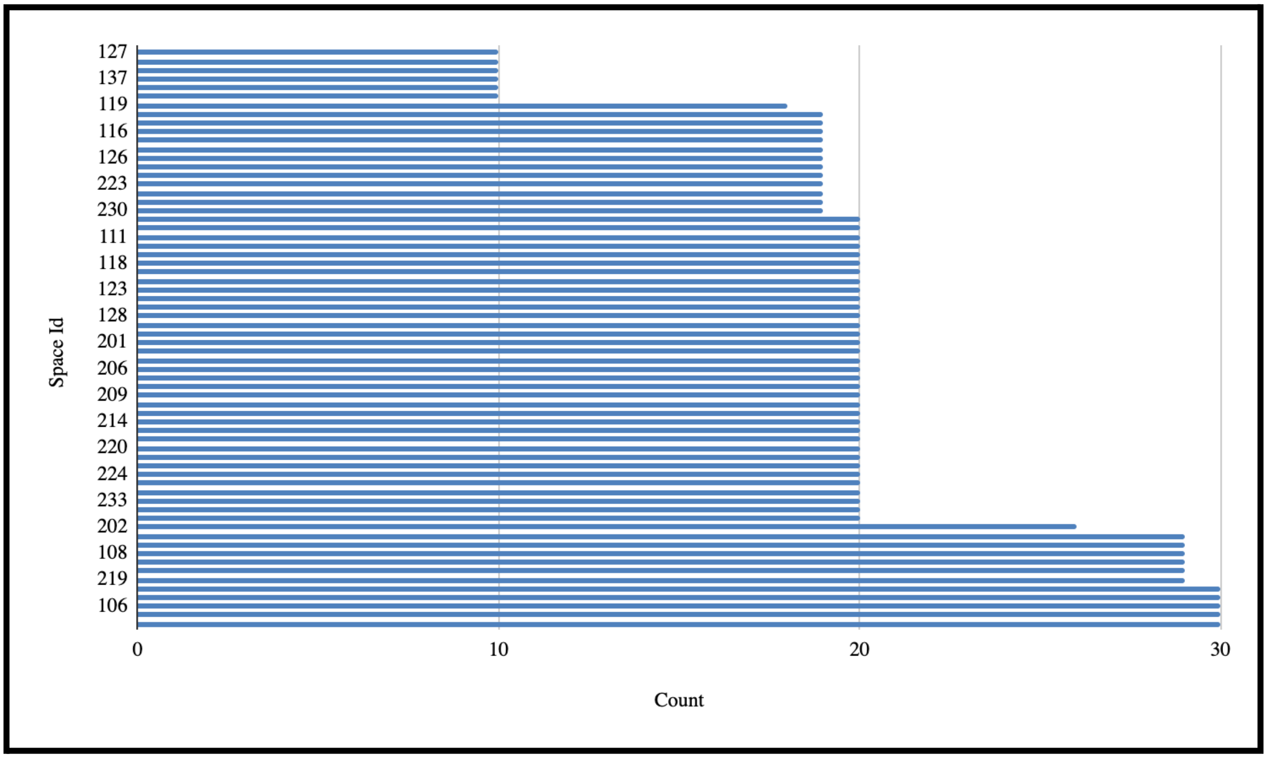

Fig 6:- Counts of Label in UJI-IndoorLoc Dataset for class imbalance.

Fig 6 is the graphical representation of instances counts of the classes of the JU-IndoorLoc dataset. Here we find that every class is having a different number of instance counts, which clearly states, that the dataset is having class imbalance.

Accuracy of JU-IndoorLoc Dataset

Train Data	Test Data	KNN	SVM	RF
JUIndoorLoc-Training-data	JUIndoorLoc-Test-data	98.836 %	94.931%	99.726 %
JUIndoorLoc-Training-data + Generated_Data_1	JUIndoorLoc-Test-data	98.836 %	94.931%	99.178 %
JUIndoorLoc-Training-data + Generated_Data_1 + Generated_Data_2	JUIndoorLoc-Test-data	98.836 %	94.931 %	99.315 %
JUIndoorLoc-Training-data + Generated_Data_1 + Generated_Data_2 + Generated_Data_3	JUIndoorLoc-Test-data	98.836 %	94.931 %	99.315 %

Table 1:- Accuracy of Original and Augmented JU-IndoorLoc Datasets

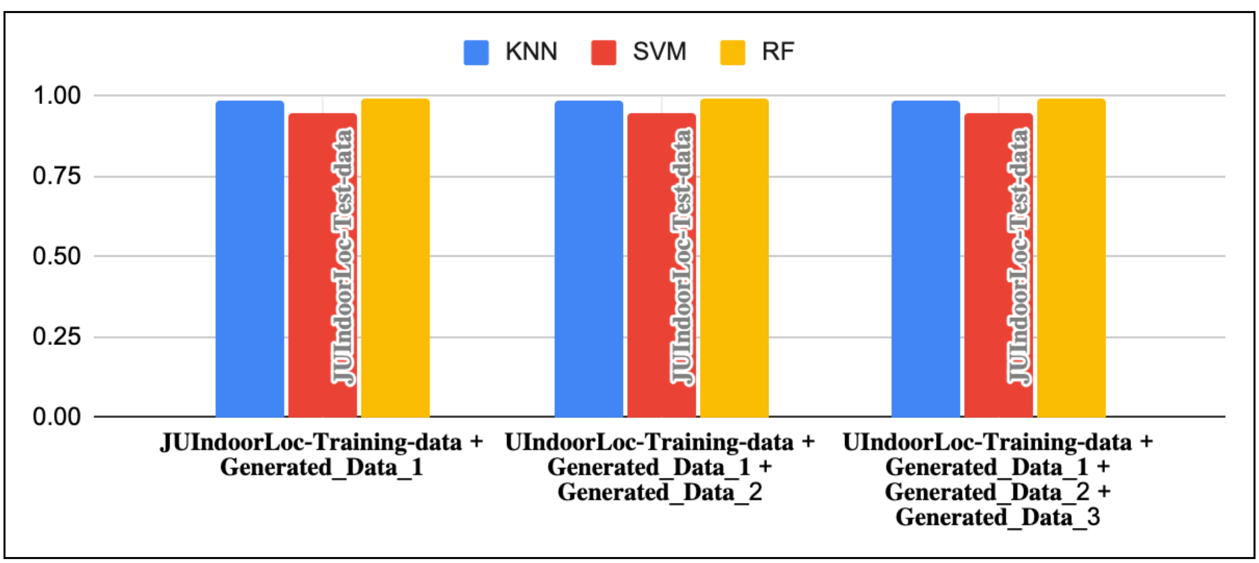

Fig 7: Accuracy of Original and Augmented JU-IndoorLoc Datasets.

As we trained the model with the Training data and Test data for ***JU-IndoorLoc Dataset*** that has been provided to us. The label presented in the Test dataset, are not present in the Train dataset. We didn't manipulate or edited the dataset to check the accuracy at its original. Later for every 4th floor 8 samples augmented generated data our Test data was the same. So the accuracy seems to be almost the same.

Accuracy of UJI-IndoorLoc Dataset

Train Data	Test Data	KNN	SVM	RF
UJI_Train	UJI_Test	62.868 %	58.823 %	83.456%
UJI_Train_minus	UJI_Test_minus	81.25 %	56.61 %	90.44%
UJI_Train_minus + Denormalised_generated_set1	UJI_Test_minus	81.25 %	58.82 %	90.44 %

Table 2:- Original and Augmented UJI-IndoorLoc Datasets

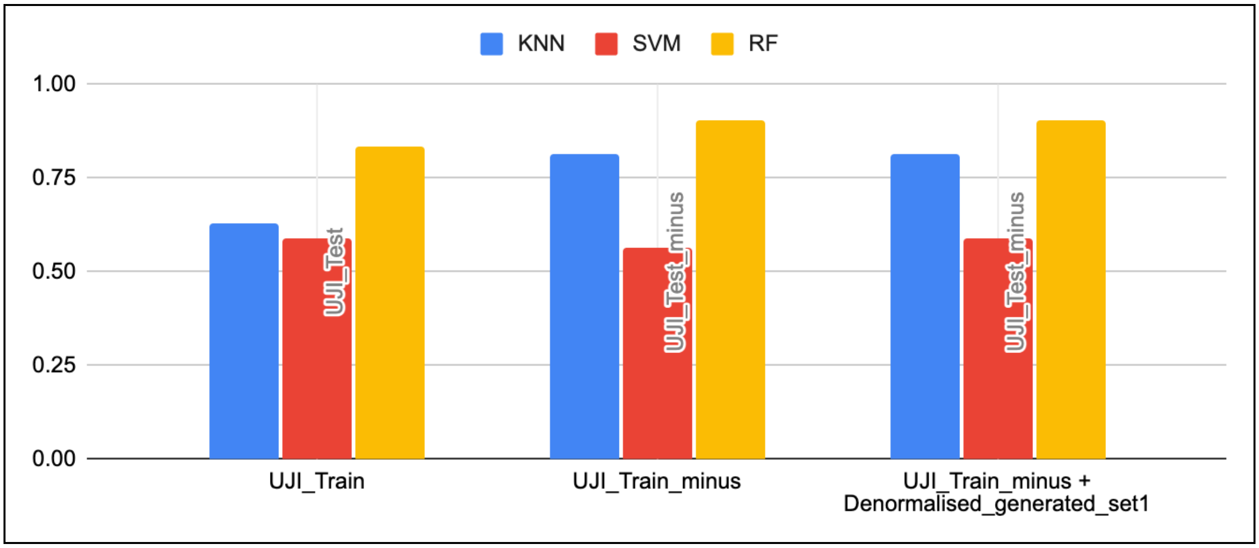

Fig 8:- Original and Augmented UJI-IndoorLoc Datasets

In Table 2 the UJI_Train_minus dataset depicts all the same values as the UJI_Train dataset with minus assigned. The dataset has been created for training the autoencoder model in the process of normalization, where a positive value depicts a dummy value or no signal of the fingerprint dataset. So to overcome the problem we integrated minus to all the values.

For JU-IndoorLoc Dataset

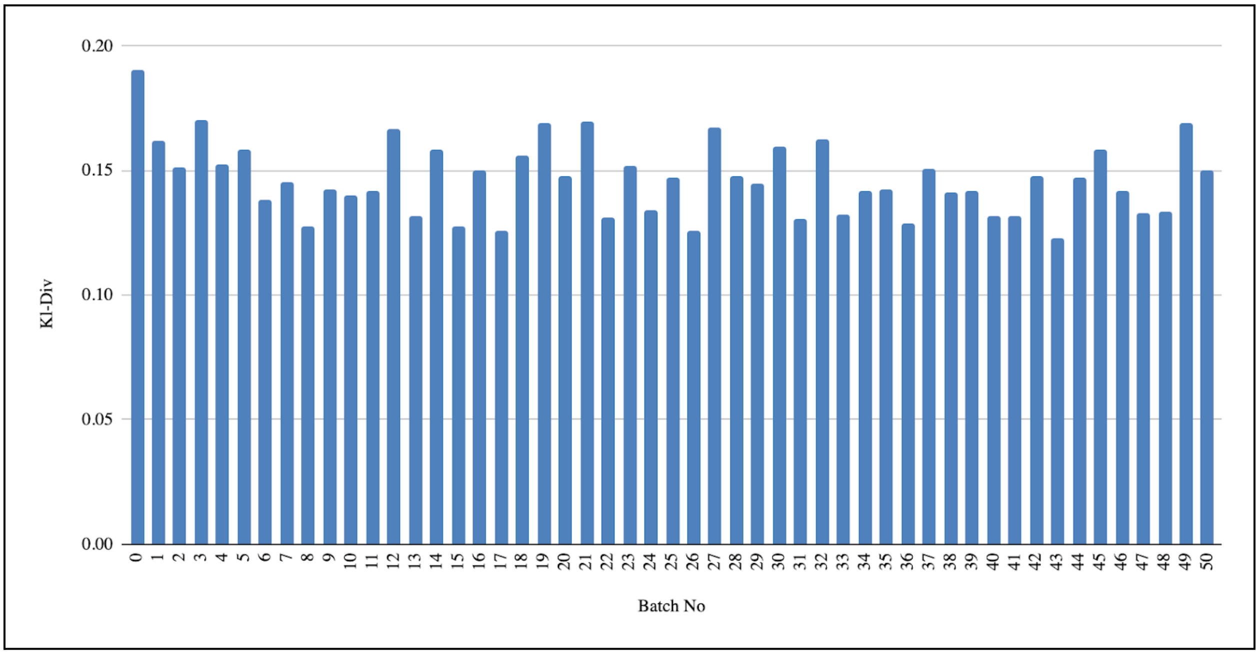

Fig 9: KL-Divergence of JU-IndoorLoc Dataset

Fig 9 represents the *KL-Divergence* of the 50 batches of input and generated dataset of the *JU-IndoorLoc Dataset.* It shows that Batch 0 has the highest value of *KL-Divergence* and Batch 17 has one of the lower values among other batches. So we plotted another graph regarding Batch 0 and Batch 17 for all labels presented in *Fig 10.*

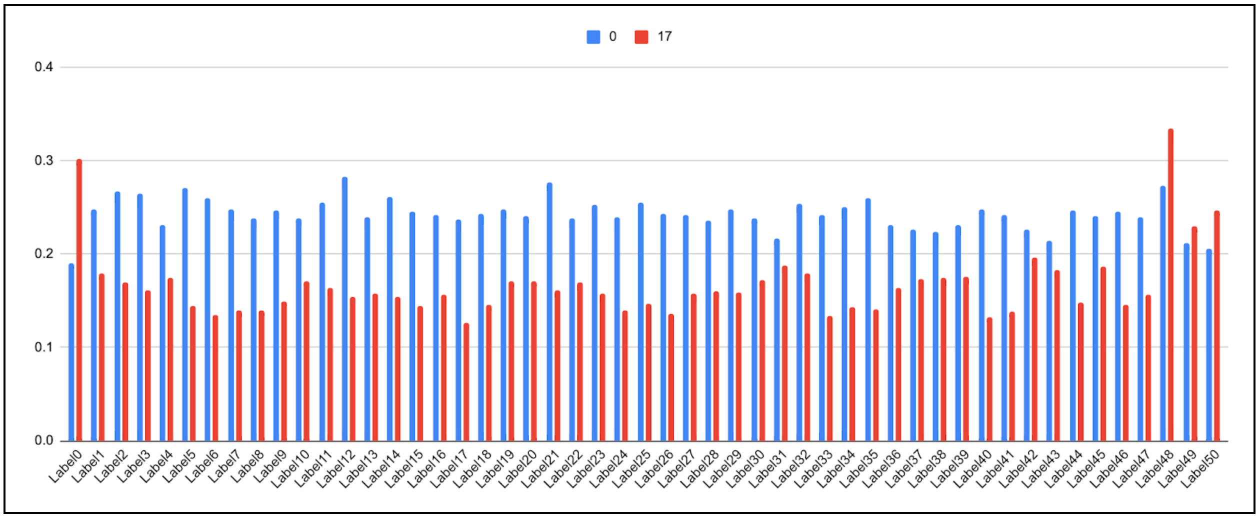

Fig 10: KL-Divergence of Batch 0 & Batch 17 of JU-IndoorLoc Data

Fig 10 represents the graphical representation of *KL-Divergence* of all the labels referencing Batch 0 and Batch 17 and it shows that, for every label the maximum value is associated with Batch 0 and the minimum with Batch 17. Here the maximum value of the KL- Divergence lies between 0.4

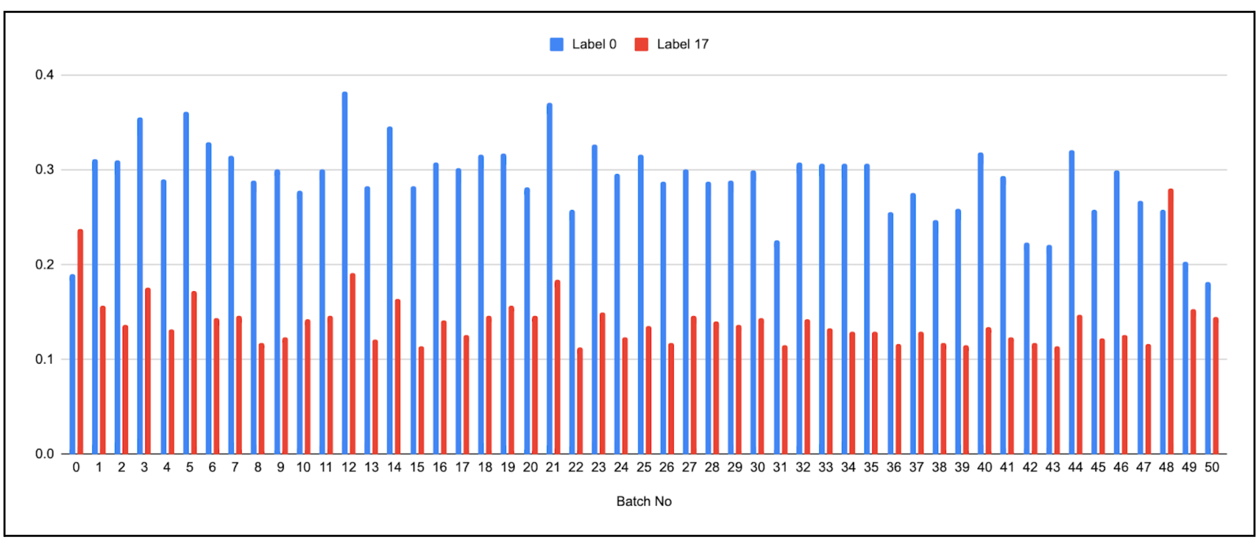

Fig 11: KL-Divergence of Label 0 & Label 17 of JU_IndoorLoc Data

Fig 11 is the graphical representation of *KL-Divergence* of all the batches after assigning labels, referencing Label 0 and Label 17. It shows that for every label the maximum value is associated with Label 0 and the minimum with Label 17. This signifies that the graph is similar to the pre-labelled graph ie presented in *Fig 10*. So we can say the class labels are assigned properly to all the batches.

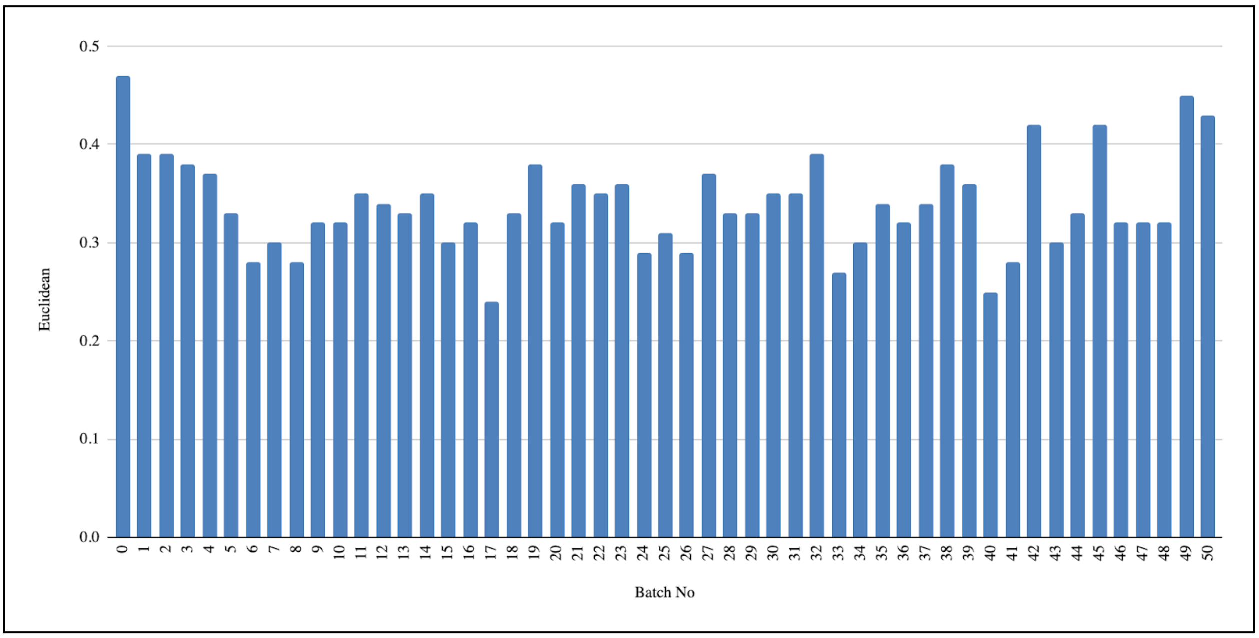

Fig 12: Euclidean Distance of JU-IndoorLoc Data

Fig 12 represents the *Euclidean Distance* of the 50 batches of input and generated dataset of the *JU-IndoorLoc Dataset.* It shows that Batch 0 has the highest value of *Euclidean Distance* and Batch 17 is having lower value than other batches. So we plotted another graph reference to Batch 0 and Batch 17 presented in *Fig 13.*

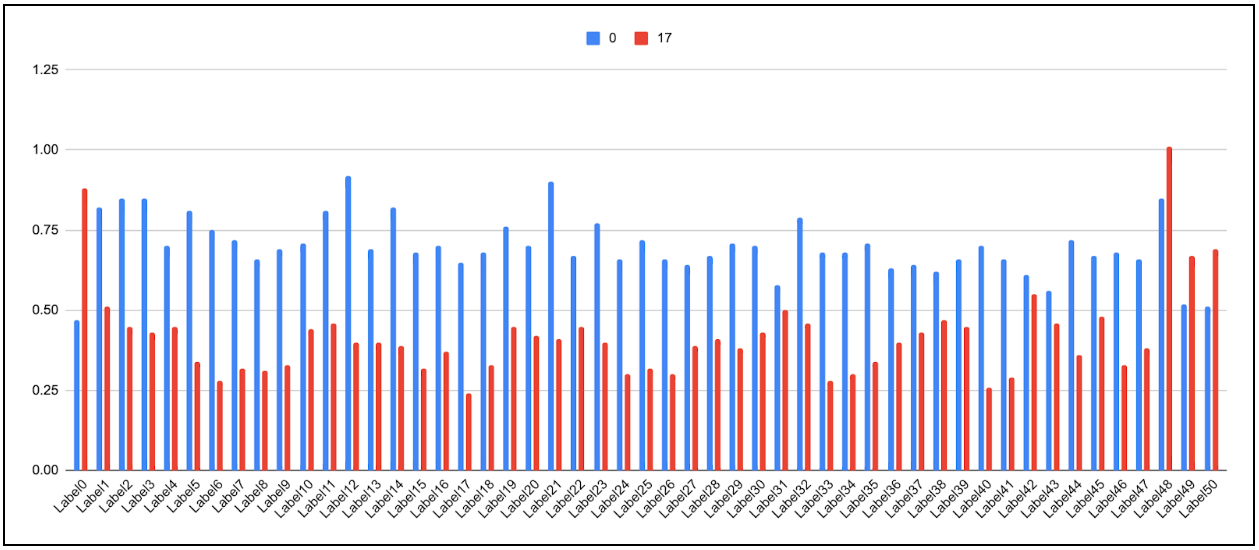

Fig 13: Euclidean Distance of Batch 0 & Batch 17 of JU-IndoorLoc Data

Fig 13 represents the graphical representation of *Euclidean Distance* of all the labels referencing Batch 0 and Batch 17 and it shows that for every label the maximum value is associated with Batch 0 and the minimum with Batch 17. Here the maximum value of the *Euclidean Distance* lies within 1.

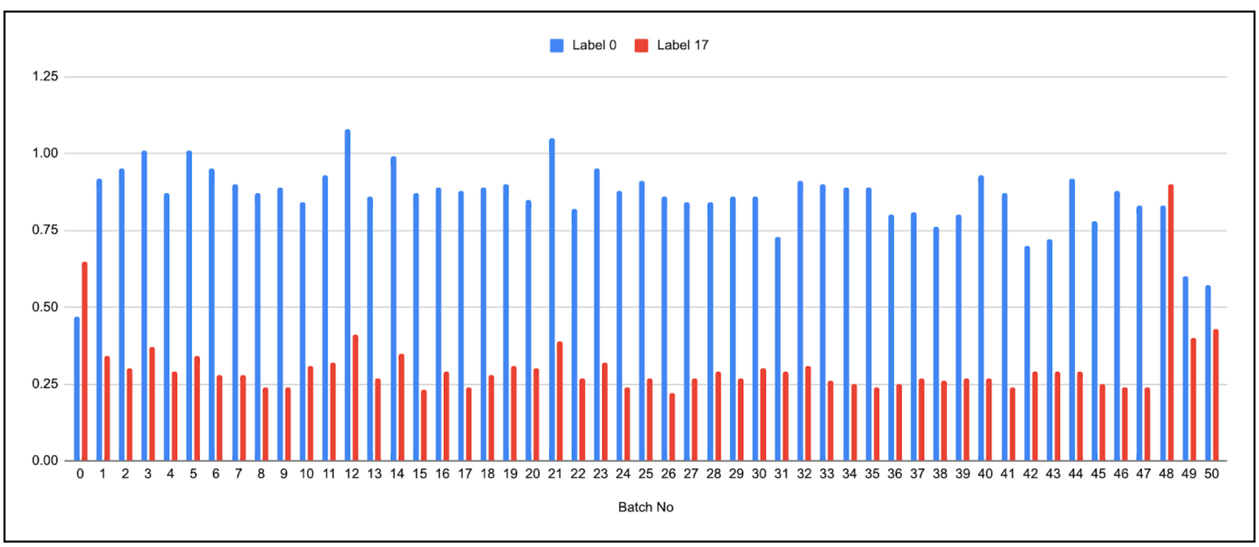

Fig 14: Euclidean Distance of Label 0 & Label 17 of JU-IndoorLoc Data

Fig 14 is the graphical representation of the *Euclidean Distance* of all the batches after assigning labels, referencing Label 0 and Label 17. We find that for every label the maximum value is associated with Label 0 and the minimum with Label 17. This signifies that the graph is similar to the pre-labelled graph ie presented in *Fig 13*. So we can say the class labels are assigned properly to all the batches.

For UJI-IndoorLoc Dataset

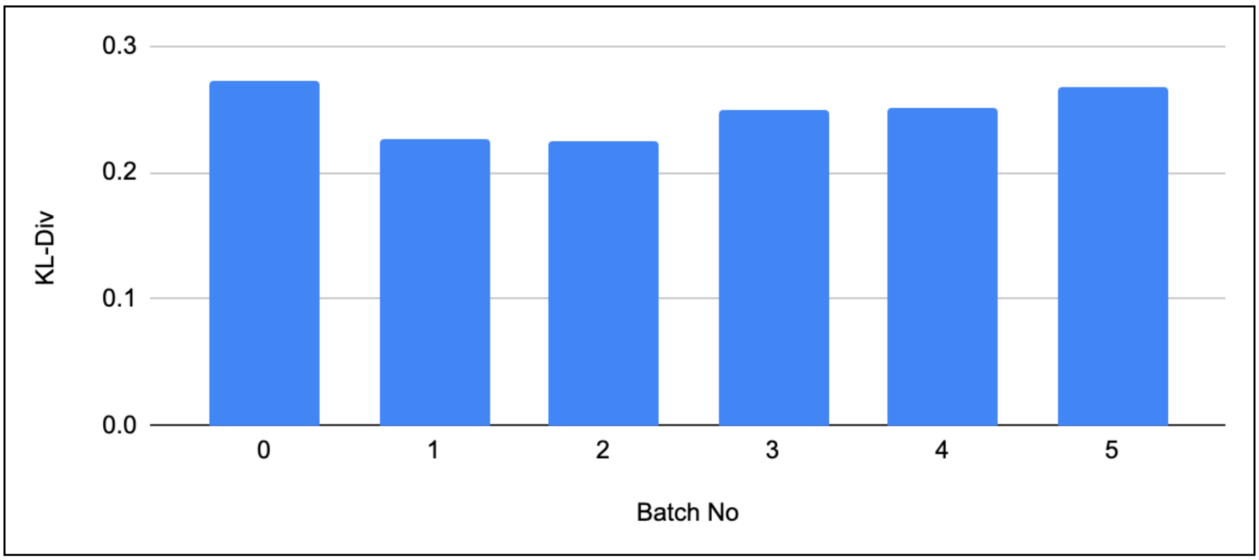

Fig 15: KL-Divergence of UJI-IndoorLoc Data

Fig 15 represents the *KL-Divergence* of the 5 batches of input and generated dataset of *UJI-IndoorLoc Dataset.* Here we can find that Batch 0 has the highest value of *KL-Divergence* and Batch 2 is having lower value among other batches. So we plotted another graph regarding Batch 2 for all labels presented in *Fig 16.*

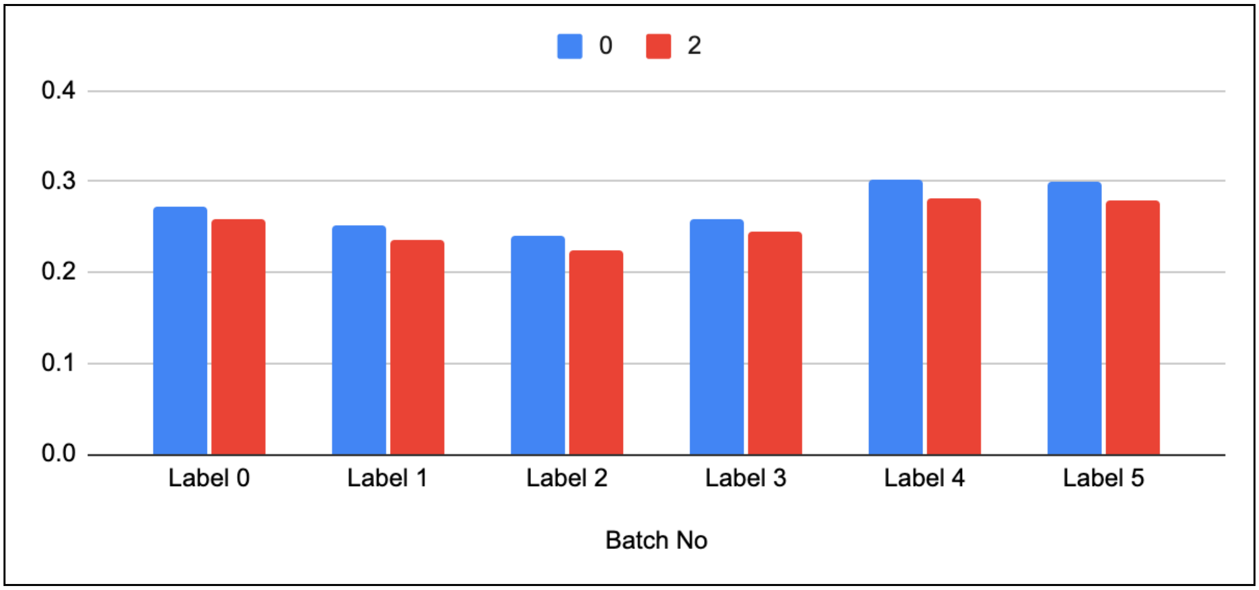

Fig 16: KL-Divergence of Batch 0 & Batch 2 of UJI-IndoorLoc Data

Fig 16 represents the graphical representation of *KL-Divergence* of all the labels referencing Batch 0 and Batch 2 and we find that for every label the maximum value is associated with Batch 0 and the minimum with Batch 17. Here the maximum value of the *KL- Divergence* lies between 0.3.

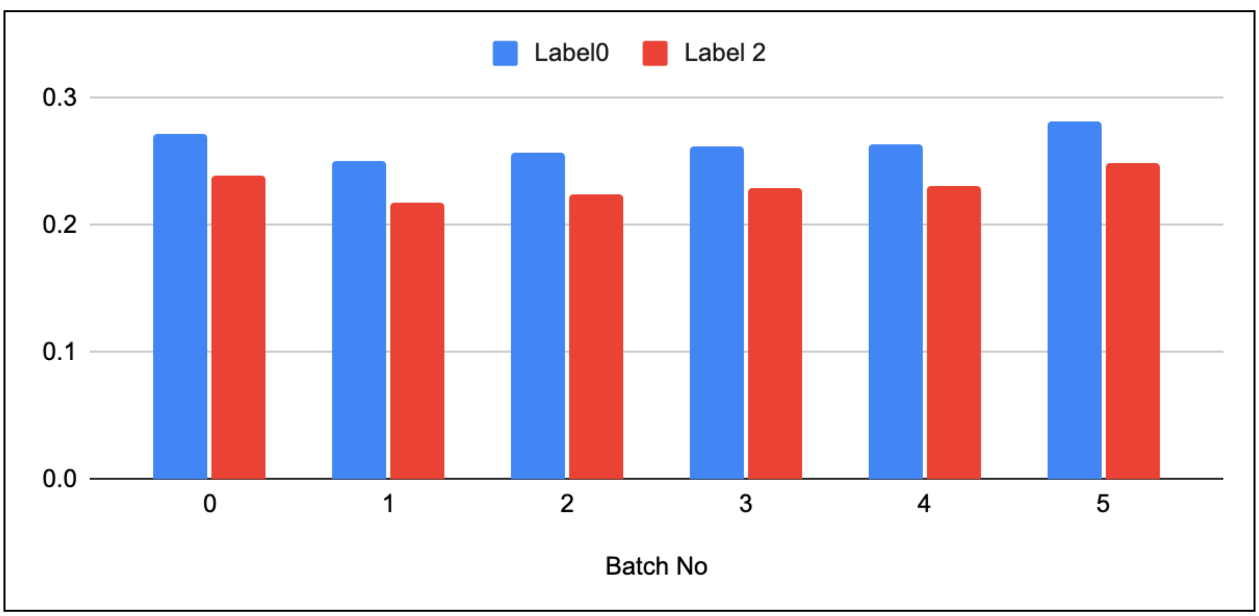

Fig 17: KL-Divergence of Label 0 & Label 2 of UJI-IndoorLoc Data

Fig 17 is the graphical representation of *KL-Divergence* of all the batches after assigning labels, referencing Label 0 and Label 2. We find that for every label the maximum value is associated with Label 0 and the minimum with Label 2. This signifies that the graph is similar to the pre-labelled graph ie presented in *Fig 16*. So it can be said that the class labels are assigned properly to all the batches.

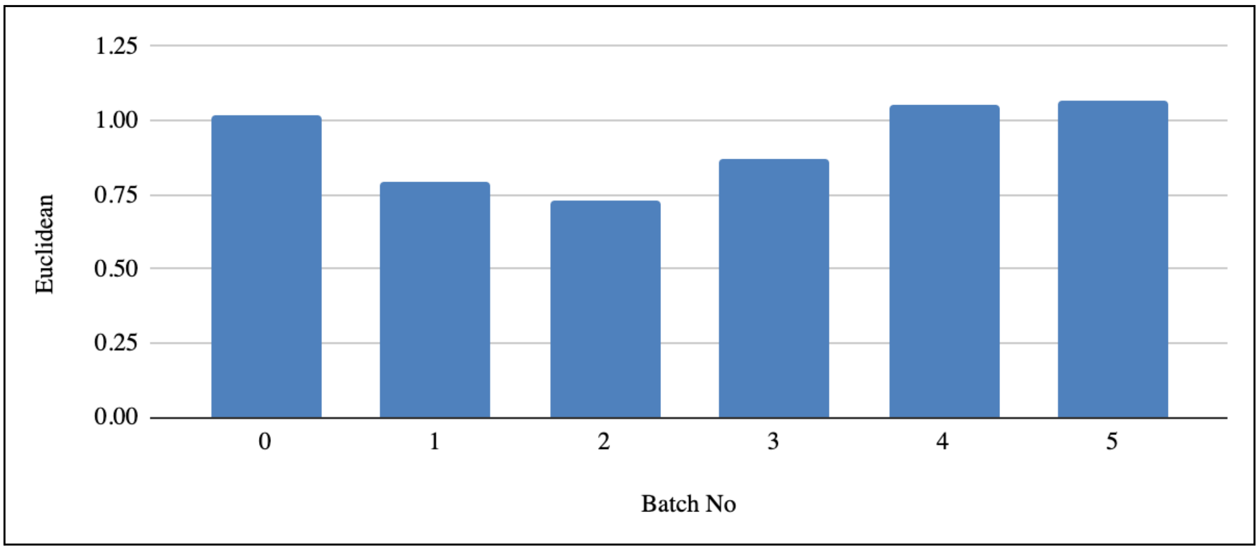

Fig 18: Euclidean Distance of UJI-IndoorLoc Data

Fig 18 represents the *Euclidean Distance* of the 5 batches of input and generated dataset of the *JU-IndoorLoc Dataset.* Here also we can find that Batch 0 has one of the highest values of *Euclidean Distance* and Batch 2 is having lower value among other batches. So we plotted another graph reference to Batch 0 and Batch 2 presented in *Fig 19.*

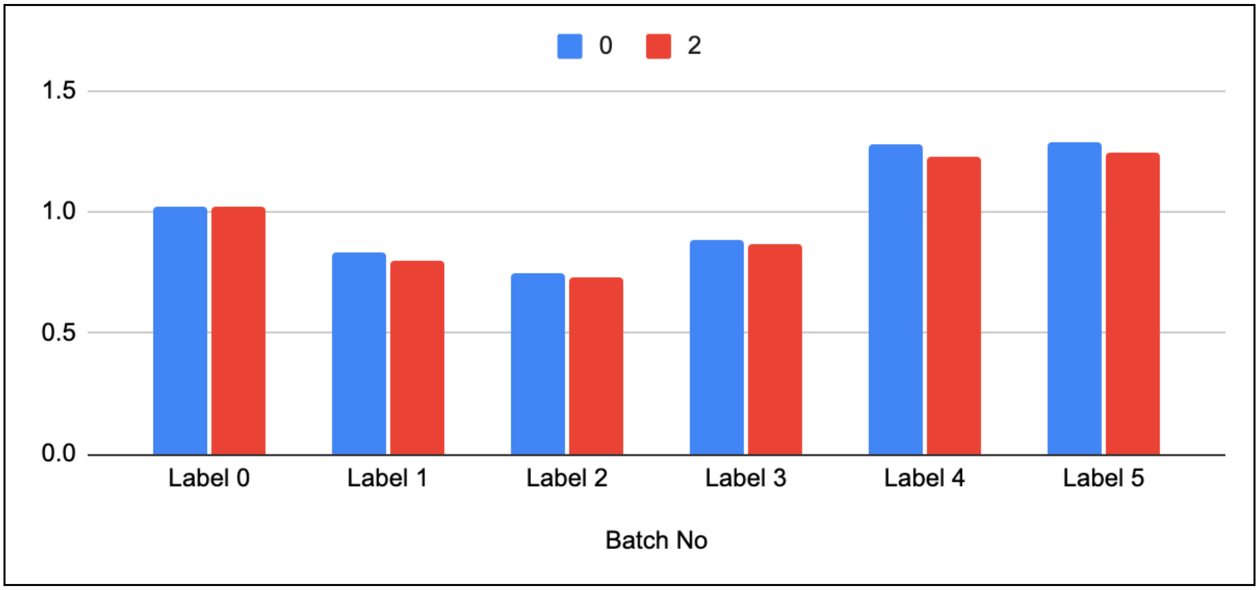

Fig 19: Euclidean Distance of Batch 0 & Batch 2 of UJI-IndoorLoc Data

Fig 19 represents the graphical representation of *Euclidean Distance* of all the labels referencing Batch 0 and Batch 17 and we find that for every label the maximum value is associated with Batch 0 and the minimum with Batch 17. Here the maximum value of the *Euclidean Distance* lies within 1.

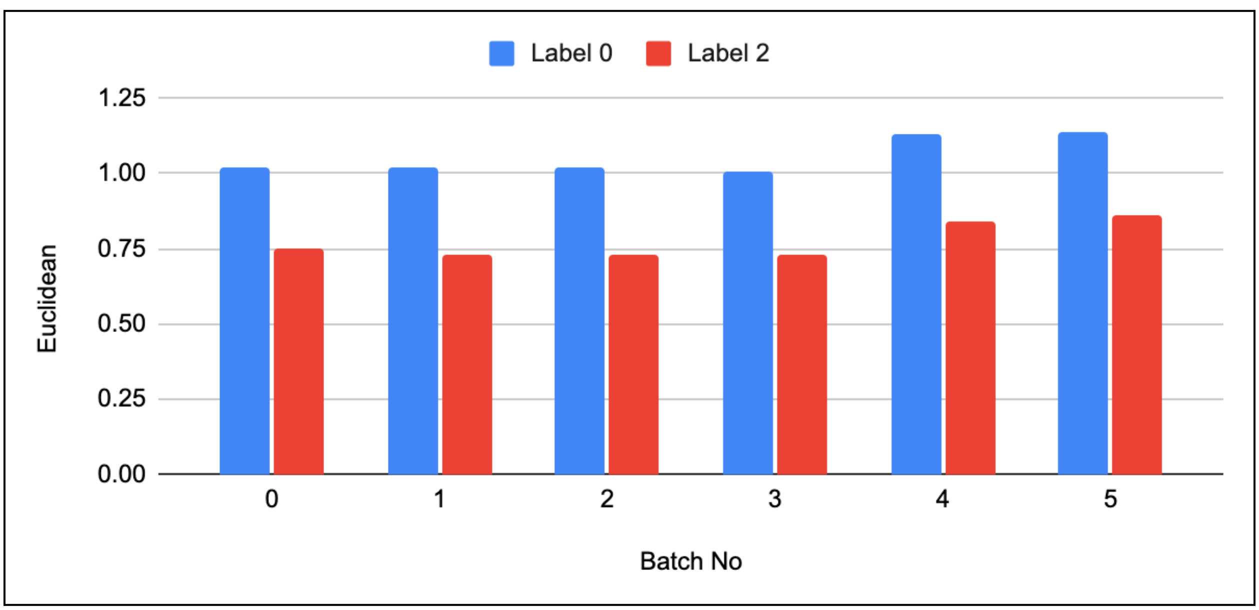

Fig 20: Euclidean Distance of Label 0 Label 2 of UJI-IndoorLoc Data

Fig 20 is the graphical representation of the *Euclidean Distance* of all the batches after assigning labels, referencing Label 0 and Label 17. We find that for every label the maximum value is associated with Label 0 and the minimum with Label 17. This signifies that the graph is similar to the pre-labelled graph ie presented in *Fig 19*. So we can say the class labels are assigned properly to all the batches.

5.4 Comparison with other CNN Models

AlexNet[29] and ZFNet[30] are both deep convolutional neural network architectures that have made significant contributions to the field of computer vision. While they share similarities in their basic structure, ZFNet is a modified version of AlexNet which gives better accuracy. There are some notable architectural differences between the two models. The key differences are as follows: • *Deeper architecture*: ZFNet has a slightly deeper architecture with an increased number of layers compared to AlexNet. Specifically, ZFNet has 8 layers of learnable weights (convolutional and fully connected layers) compared to AlexNet's 5 layers. • *Filter size*: ZFNet introduced a novel concept by using smaller filter sizes in the initial layers compared to AlexNet. While AlexNet used an 11x11 filter size in the first layer, ZFNet reduced it to a 7x7 filter size. This change was motivated by the observation that smaller filters can capture more fine-grained details in the early layers of the network. • *Convolutional layer configuration*: ZFNet introduced a specific configuration of convolutional layers to capture more detailed information. In particular, it used a combination of small filter sizes and a larger stride in the early layers to increase the receptive field while reducing the spatial dimensions. This configuration allowed ZFNet to capture both local and global features effectively. • Visualizing network activations: Another notable difference is that ZFNet introduced a technique for visualizing network activations, which was not explicitly presented in the original AlexNet. This visualization technique helped me understand how the network learns and what features it captures at different layers.

5.5 Summary

Overall the experiment results, from both datasets, *JU-IndoorLoc and UJI-IndoorLoc* presented an excellent percentage of accuracy for the localization. The graphical representation from - to - showed from predicting the distribution points for the input and generated dataset with the help of *KL-Divergence* and *Euclidean distance* has been effective. But the main observation that has to be noted is that the *JU-IndoorLoc* dataset accuracy was wonderful both before and after augmentation, their accuracy seems to have a slighter change in the accuracy. But for the *UJI-IndoorLoc* it looks different, the accuracy before and after the augmentation has a great change where the dataset after the augmentation gives an accuracy of 90%.

Chapter 6

Conclusion & Future Work

6.1 Conclusion

In this thesis, we proposed an autoencoder-based indoor localization approach to address the class imbalance problem in fingerprint data. The objective was to improve the accuracy and reliability of indoor localization by effectively handling the imbalance in the datasets. Our experimental results demonstrated the effectiveness of the proposed approach in reducing class imbalance and improving indoor localization accuracy. By training the autoencoder on the fingerprint dataset, we successfully learned robust and discriminative features that captured the underlying patterns in the data. The autoencoder's ability to reconstruct the input data helped in mitigating the impact of class imbalance and enhancing the performance of the localization model. We implemented and evaluated two Algorithms to handle the class imbalance problem, including *KL-Divergence* and *Euclidean Distance*. Our experiment showed that original data and generated data led to improved localization accuracy. In conclusion, this thesis successfully addressed the class imbalance problem in fingerprint data for indoor localization through an autoencoder-based approach. The results demonstrated the significance of leveraging autoencoders for feature learning and dimensionality reduction, leading to enhanced accuracy and robustness. By effectively reducing the class imbalance, our approach has the potential to improve the performance of indoor localization systems and contribute to the advancement of location-based services in various domains.

6.2 Future Work

Researchers can investigate and create more sophisticated autoencoder architectures created specifically for RSSI fingerprinting. To more effectively capture the spatial and temporal properties of the RSSI data, this could entail novel network topologies like convolutional autoencoders or recurrent autoencoders. Other sensor modalities, such as WiFi, Bluetooth, or interior data, can also offer important information for indoor localization with RSSI data in addition to multi-modal fusion [26]. To increase, the precision and robustness of indoor localization systems, future research can concentrate on combining several senses utilizing autoencoder-based techniques. Additionally, by creating realistic and diverse synthetic samples with GANs and autoencoders, one may balance the dataset and enhance the effectiveness of the localization system [27]. In different control and decision-making tasks, deep reinforcement learning (DRL) algorithms have demonstrated success [28]. To enable intelligent decision-making, such as dynamic localization, researchers might investigate the integration of autoencoder-based localization systems with DRL approaches. Finally, the availability of benchmark datasets that are standardized and assessment measures specific to autoencoder-based indoor localization can enable fair comparisons between various approaches and spur additional study. In the future, developing such datasets and establishing suitable evaluation metrics will enable thorough analysis and comparison of various methodologies.

References

[1] Shang, S., Wang, L. Overview of WiFi fingerprinting-based indoor positioning. *IET Commun.* 16, 725– 733 (2022). https://doi.org/10.1049/cmu2.12386

[2] R. Joseph and S. B. Sasi, "Indoor Positioning Using WiFi Fingerprint," 2018 International Conference on Circuits and Systems in Digital Enterprise Technology (ICCSDET), Kottayam, India, 2018, pp. 1-3, doi: 10.1109/ICCSDET.2018.8821184.

[3] Woo, S., Jeong, S., Mok, E., Xia, L., Choi, C., Pyeon, M. and Heo, J., 2011. Application of WiFi-based indoor positioning system for labor tracking at construction sites: A case study in Guangzhou MTR. *Automation in Construction*, *20*(1), pp.3-13.

[4] Calderoni, L., Ferrara, M., Franco, A. and Maio, D., 2015. Indoor localization in a hospital environment using random forest classifiers. *Expert Systems with Applications*, *42*(1), pp.125-134.

[5] M. Poggi and S. Mattoccia, "A wearable mobility aid for the visually impaired based on embedded 3D vision and deep learning," 2016 IEEE Symposium on Computers and Communication (ISCC), Messina, Italy, 2016, pp. 208-213, doi: 10.1109/ISCC.2016.7543741.

[6] C. K. M. Lee, C. M. Ip, T. Park and S. Y. Chung, "A Bluetooth Location-based Indoor Positioning System for Asset Tracking in Warehouse," 2019 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Macao, China, 2019, pp.

1408-1412, doi: 10.1109/IEEM44572.2019.8978639.

[7] W. Yong, L. Qing, W. Lei and S. Hao, "Improvement of RFID locating algorithm in warehouse security system," 2017 13th IEEE International Conference on Electronic Measurement & Instruments (ICEMI), Yangzhou, China, 2017, pp. 190-195, doi: 10.1109/ICEMI.2017.8265762.

[8] Regus, M., Talar, R., & Labudzki, R. (2019, October). Indoor positioning and navigation system for autonomous vehicles based on RFID technology. In IOP conference series: materials science and engineering (Vol. 659, No. 1, p. 012059). IOP Publishing.

[9] A. Famili, A. Stavrou, H. Wang and J. -M. Park, "PILOT: High-Precision Indoor Localization for Autonomous Drones," in IEEE Transactions on Vehicular Technology, vol. 72, no. 5, pp. 6445-6459, May 2023, doi: 10.1109/TVT.2022.3229628.

[10] K. Blake, R. Huey, D. Menezes, J. Root, L. Tran and H. Chen, "Robot Navigation Using Ultra-Wideband Indoor Localization and Dead Reckoning Algorithms," 2022 Opportunity Research Scholars Symposium (ORSS), Atlanta, GA, USA, 2022, pp. 12-15, doi: 10.1109/ORSS55359.2022.9806034.

[11] Q. Xu, Z. Wu, Y. Yang and L. Zhang, "The difference learning of hidden layer between autoencoder and variational autoencoder," 2017 29th Chinese Control And Decision Conference (CCDC), Chongqing, China, 2017, pp. 4801-4804, doi: 10.1109/CCDC.2017.7979344.

[12] J. Liu, N. Liu, Z. Pan and X. You, "AutLoc: Deep Autoencoder for Indoor Localization with RSS Fingerprinting," 2018 10th International Conference on Wireless Communications and Signal Processing (WCSP), Hangzhou, China, 2018, pp. 1-6, doi: 10.1109/WCSP.2018.8555665.

[13] Niu, Ruixin & Vempaty, Aditya & Varshney, P.K.. (2018). Received-Signal-Strength-Based Localization in Wireless Sensor Networks. Proceedings of the IEEE. PP. 1-17. 10.1109/JPROC.2018.2828858.

[14] Ezzati, Zahra & Hajihoseini, Amirhosein & Ghorashi, Seyed Ali & Ghavami, Mohammad. (2021). A fingerprint technique for indoor localization using autoencoder based semi-supervised deep extreme learning machine. Signal Processing. 181. 107915. 10.1016/j.sigpro.2020.107915.

[15] B. Chidlovskii and L. Antsfeld, "Semi-supervised Variational Autoencoder for WiFi Indoor Localization," 2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Pisa, Italy, 2019, pp. 1-8, doi: 10.1109/IPIN.2019.8911825.

[16] R. Xin, J. Zhang and Y. Shao, "Complex network classification with convolutional neural network," in Tsinghua Science and Technology, vol. 25, no. 4, pp. 447-457, Aug. 2020, doi: 10.26599/TST.2019.9010055.

[17] Mers, M., Yang, Z., Hsieh, Y.-A., & Tsai, Y. (James). (2023). Recurrent Neural Networks for Pavement Performance Forecasting: Review and Model Performance Comparison. Transportation Research Record, 2677(1), 610–624. https://doi.org/10.1177/03611981221100521

[18] D. Liao, S. Huang, Y. Tan and G. Bai, "Network Intrusion Detection Method Based on GAN Model," 2020 International Conference on Computer Communication and Network Security (CCNS), Xi'an, China, 2020, pp. 153-156, doi: 10.1109/CCNS50731.2020.00041.

[19] K. Wangi and A. Makandar, "Autoencoder for Image Retrieval System using Deep Learning Technique with Tensorflow and Kears," 2023 IEEE International Conference on Integrated Circuits and Communication Systems (ICICACS), Raichur, India, 2023, pp. 1-5, doi: 10.1109/ICICACS57338.2023.10099675.

[20] S. A. H. Minoofam, A. Bastanfard and M. R. Keyvanpour, "TRCLA: A Transfer Learning Approach to Reduce Negative Transfer for Cellular Learning Automata," in IEEE Transactions on Neural Networks and Learning Systems, vol. 34, no. 5, pp. 2480-2489, May 2023, doi: 10.1109/TNNLS.2021.3106705.

[21] L. Lyu, Y. Shen and S. Zhang, "The Advance of Reinforcement Learning and Deep Reinforcement Learning," 2022 IEEE International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA), Changchun, China, 2022, pp. 644-648, doi: 10.1109/EEBDA53927.2022.9744760.

[22] Z. Luo, J. Li and Y. Zhu, "A Deep Feature Fusion Network Based on Multiple Attention Mechanisms for Joint Iris-Periocular Biometric Recognition," in IEEE Signal Processing Letters, vol. 28, pp. 1060-1064, 2021, doi: 10.1109/LSP.2021.3079850.

[23] D. Martín-Gutiérrez, G. Hernández-Peñaloza, A. B. Hernández, A. Lozano-Diez and F. Álvarez, "A Deep Learning Approach for Robust Detection of Bots in Twitter Using Transformers," in IEEE Access, vol. 9, pp. 54591-54601, 2021, doi: 10.1109/ACCESS.2021.3068659.

[24] Joyce, J.M. (2011). Kullback-Leibler Divergence. In: Lovric, M. (eds) International Encyclopedia of Statistical Science. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-04898-2_327

[25] Kannoujia, Tarun Kumar and Kumar, Vineet and Jha, Nikhil Ranjan and Mishra, Dr. Gauri Shankar, Social Distancing Detection Using Euclidean Distance Formula (July 14, 2022). Proceedings of the Advancement in Electronics & Communication Engineering 2022, Available at SSRN: https://ssrn.com/abstract=4157334

[26] J. Yu, P. Wang, T. Koike-Akino and P. V. Orlik, "Multi-Modal Recurrent Fusion for Indoor Localization," ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, Singapore, 2022, pp. 5083-5087, doi: 10.1109/ICASSP43922.2022.9746071.

[27] Carreon, A., Barwey, S. and Raman, V., 2023. A generative adversarial network (GAN) approach to creating synthetic flame images from experimental data. *Energy and AI*, *13*, p.100238.

[28] Singh, V., Chen, S.S., Singhania, M., Nanavati, B. and Gupta, A., 2022. How are reinforcement learning and deep learning algorithms used for big data based decision making in financial industries–A review and research agenda. *International Journal of Information Management Data Insights*, *2*(2), p.100094.

[29] Krizhevsky, A., Sutskever, I. and Hinton, G.E., 2017. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), pp.84-90.

[30] Zeiler, M.D. and Fergus, R., 2014. Visualizing and understanding convolutional networks. In Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13 (pp. 818-833). Springer International Publishing.

[31] Roy, P., Chowdhury, C., Ghosh, D., & Bandyopadhyay, S. (2019). JUIndoorLoc: A ubiquitous framework for smartphone-based indoor localization subject to context and device heterogeneity. *Wireless Personal Communications*, *106*(2), 739–762. https://doi.org/10.1007/s11277-019-06188-2

[32] J. Torres-Sospedra et al., "UJIIndoorLoc: A new multi-building and multi-floor database for WLAN fingerprint-based indoor localization problems," 2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Busan, Korea (South), 2014, pp. 261-270, doi: 10.1109/IPIN.2014.7275492.