Elder People Falling Detection

First Author: Haotian Xiang
UNI: hx2341

Second Author: Jirong Ren
UNI: jr4248

Abstract

Eldering caring is crucial issues in the whole world as the aging problem is gradually an increasing problem for all countries. The human activities recognition is the core of the eldering caring system. However, the human activity is recorded by a FMCW radar which produce a giant signal data for one human action (which can be split into over 962 slices for one signal). And the whole training process will contain over 1636 activities which will generate a dataset with over 962300000 samples. It can easily exceed the memory the ram size of single computer even a high-end working station (the dataset is over 128GB.) Thus, big data processing is required to tackle this problem.

The report aims to adopt the MATLAB to processing raw radar signal and generate the dataset to train the classifier. The training is performed on the frame of Pyspark and RDD-Machine Learning to cope with the heavy dataset. To classify different human activities, micro-Doppler features are extracted from the Doppler signal. And in total, 20 micro-Doppler features are used to training the classifier.

The statistic learning method in RDD and Pyspark are used to develop the human activity classifier. SVM, Decision Tree, XGBoost, NaiveBayes, and KNN. In the trail stage, the best classification accuracy is 78.4% by the quadric SVM. In conclusion, the Pyspark and RDD can handle these amounts of size of samples. But the parameters in statistic learning needs to be adjusted and a more sophistic method is required to process the Doppler signal.

1. Introduction

The eldering care is becoming a gradually heating problem and issue all over the world. As we grow older, most of us want to continue to live in the homes we cherish and that are full of happy memories.

The challenge for many families is providing the level of support our loved ones need to do so safely and independently. Often, we cannot be with our loved ones as much as we would like. It can be worrying not knowing what is happening day-to-day. For many families, the Covid pandemic has highlighted this concern.

Smart devices, home automation and voiced controlled entertainment systems are now commonplace in our homes. Smart home sensors help nurses and family members identify and predict problems by monitoring changes in routine. With the aging issue is more and more critical globally, a smart in-home monitoring system is required for years to cope with the increasing aging issues and taking care the elder people at home [1]. The human activity monitoring and recognizing is a crucial and core part of the eldering caring system

2. Related Work

Training methods and the choice of training samples vary in different related papers. In most of projects, Doppler Spectrums are used and are trained on cloud network. According to their work, the average accuracy of classification is between 92% to 100%. [2]

2.1 Choice of training samples

Three categories of signals are mainly adopted in currently human activities recognition are: Range-Time (RT) [3], Doppler and spectrum (RD) [4], and micro-Doppler signals (MD) [5]. Among these training samples, Range-Time spectrum was used by few works. Range-Time spectrum was treated as the input to train the neural network combined with the training results, which are from another trained network. The average accuracy achieved by those networks can reach approximately 93% [6].

Another mainstream is using the Doppler signature to generate the Point Cloud after RDS Mesh. Then consider information on the surface of the range Doppler signal and the grey image in phase domain. The average accuracy achieved by this type of methods can reach to approximately 94.3% [7]. The last mainstream of training sample choices is adopting the Micro-Doppler signature. The feature extraction of micro-Doppler signal from Energy features and physical-combined features are applied to aid the statistic leaning progress. The average accuracy achieved by those methods can reach to approximately 99%, with advanced feature extraction progress.

According to those related works, it is concluded that feature extraction progress is a worthwhile section to improve the performance of the designed classifier. Meanwhile, some projects applied all the micro-Doppler features (up to 24 types) [8], while others applied only 9-15 features and achieved great accuracy as well, around 99% [9].

2.2 Classifier training with Statistic Learning

In industry, statistic learning is the most popular method to train the classifier, including Decision trees, Bayes classifier, K-nearest-neighbor (KNN), XGBoost and Support Vector machine（SVM）. Among those statistic learning methods, SVM is the most promising method. The SVM with micro-Doppler feature extraction had successfully reached the accuracy of 92% [8]. Meanwhile, some hierarchy classifiers and more specified in a secondary classification, and clusters are adopted to improve the recognition accuracy.

2.3 Classifier training with network

Neural network is the first choice for most recent projects. Large number of projects use classifiers trained by Convolutional Neural Network. Lightweight neural networks are deployed, VGG-19 [11], VGG-16 [9], Mobile Net, and Shuffle Net, with the best accuracy of 95.4% [8].

The traditional computer-based neural network is applied as well. The Convolutional Neural Network (CNN) and two CNN combined neural networks are deployed, which gets the best accuracy of 93.9% [12]. Besides, some more sophisticated fusion methods are applied, which boosts the accuracy to 94.27% [13]. Point Net with CNN is used as the classifier, which gets an accuracy of 93.4% [12]. Furthermore, a deep CNN with micro-Doppler features is deployed to build the classifier, getting an accuracy of 99.24% [15].

RNN and LSTM are applied to train the classifier as well. The Gated Recurrent Unit (GRU), derived from RNN, with micro-Doppler features, gets an accuracy of 94.3% [12]. LSTM neural network with micro-Doppler features achieves 93.9% [16].

To sum up, the human activity classifier based on CNN can achieve an accuracy of nearly 100%, while for RNN-based classifier, the average accuracy is only approximately 94.3%.

3. Methodology

3.1 Source data processing

The original data is collected from human activities radar signatures[19]. As the raw data collected from the FMCW radar is usually the time domain output power, it is a one-dimensional array (Complex or Real). To get more information and features from the signal, the time frequency is required, which enables study and parameter collecting from both frequency and time domain [9]. The mainstream time-frequency transformation formula is Fast Fourier Transform (FFT).

The first four elements of the raw samples are the carrier frequency (5.8GHz in C-Band), the duration of the chirp (1ms), the length of the samples every recording cycle (128 samples), and the bandwidth of the chirp (400MHz). The rest of the elements of the raw samples are the recording data 1×128000 (Act1) and 1×64000 (Other human activities). Besides, the recording length for Act1 is 10ms, and for rest of human activities are 5ms.

The verification of the raw data length:

$$Sample\ Length = \frac{128 \times recording\ length}{duration\ of\ the\ chirp}$$

3.2 ROI Extraction and Slices Labeling

The slices in the spectrum are recategorized as original activities and Idle slices. The steps to determine the region of interest are shown below. The first step is to justify the vertical boundary of the ROI. The rows containing all zero pixels is deleted, and the rest rows are vertical endpoint.

Then, the second step is to determine the horizontal boundary. The slices containing all-zero value are eliminated. Next, all the lengths and widths of ROIs are put into comparison, and only the maximum width, and length are left. The global region of interest is set.

Then, the slices in the are classified into activities slices and idle slices. So, the new label Act7 (Idle) is generated. All the slices are gathered according to its label.

3.3 Specification of dataset

The raw radar signal is process by MATLAB. The final Doppler spectrum is 981*1063.

The dataset is upload to the Bucket of GCP. The total amount of dataset is shown below:

Activity	NO. of Activities
Act1	298191
Act2	209166
Act3	228920
Act4	263691
Act5	262042
Act6	102253

Table1. Amount of dataset

The construction of training dataset is constructed by Pyspark RDD and Pandas. The slice of features is evenly split and reallocate to corresponding features. The dataset is constructed with both the micro-Doppler signals and their corresponding features.

Besides, all the labels are added manually. But as we have over 9 million samples, the original VM's ram is consumed up quickly. Thus, all the data is transformed into H5 File [1] which dramatically reduce the storage consumption. Then, the whole H5 structured samples are processed by Spark sql and processed dataset is stored in the GCP cloud bucket.

The construction of the training dataset is shown below：

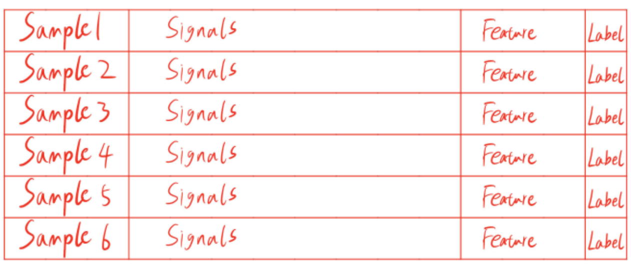

Fig1. Structure of dataset

3.4 Statistical analysis

Pyspark is used to read the dataset in the cloud storage, and following Pyspark Statistic-Learning methods, SVM, XGBoost, NaïveBayes and KNN are used to train the Human Activities Classifier.

The statistical learning methods in Pyspark MLLIB is all used from Trees, KNN SVM, Naïve Bayes to XGBoost. The dataset is split into initially 7:3 for the training and test set. And both CrossValidator and TrainValidationSplit in Spark MLLIB are used to optimize the hyperparameters. The final classification result is exhibited by MATLAB to get a better illustrating result.

The stack of the software is shown below:

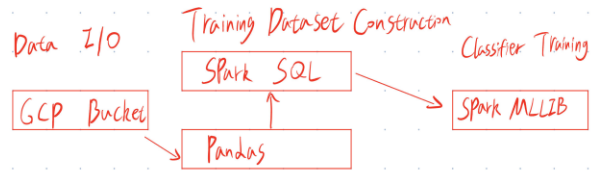

Fig2. Stack of software

The constructed dataset will be upload to GCP bucket. The training progress will be proceeded by Spark SQL. Then, Spark MLLIB will train the cluster model.

3.5 Statistic learning methods

3.5.1 SVM

The Support Vector Machine (SVM) is a linear classifier that can be viewed as an extension of the Perceptron developed by Rosenblatt in 1958. The Perceptron guaranteed that you find a hyperplane if it exists. The SVM finds the maximum margin separating hyperplane. We define a linear classifier:

$$h(x) = \sin\,(w^T x + b)$$

We assume a binary classification setting with labels $\{+1,-1\}$ $\{+1,-1\}$. A hyperplane is defined through w, bw, b as a set of points such that

$$H = \{x|w^T x + b = 0\}$$

Let the margin γ be defined as the distance from the hyperplane to the closest point across both classes.

3.5.2 Decision tree

A decision tree is a non-parametric supervised learning algorithm, which is utilized for both classification and regression tasks [18]. It has a hierarchical, tree structure, which consists of a root node, branches, internal nodes and leaf nodes.

a decision tree starts with a root node, which does not have any incoming branches. The outgoing branches from the root node then feed into the internal nodes, also known as decision nodes. Based on the available features, both node types conduct evaluations to form homogenous subsets, which are denoted by leaf nodes, or terminal nodes. The leaf nodes represent all the possible outcomes within the dataset.

Decision tree learning employs a divide and conquer strategy by conducting a greedy search to identify the optimal split points within a tree. This process of splitting is then repeated in a top-down, recursive manner until all, or most records have been classified under specific class labels. Whether or not all data points are classified as homogenous sets is largely dependent on the complexity of the decision tree.

However, as a tree grows in size, it becomes increasingly difficult to maintain this purity, and it usually results in too little data falling within a given subtree. When this occurs, it is known as data fragmentation, and it can often lead to overfitting.

To reduce complexity and prevent overfitting, pruning is usually employed; this is a process, which removes branches that split on features with low importance. The model's fit can then be evaluated through the process of cross-validation. Another way that decision trees can maintain their accuracy is by forming an ensemble via a random forest algorithm; this classifier predicts more accurate results, particularly when the individual trees are uncorrelated with each other.

The strengths of decision tree methods are:

a. Decision trees are able to generate understandable rules.
b. Decision trees perform classification without requiring much computation.
c. Decision trees are able to handle both continuous and categorical variables.
d. Decision trees provide a clear indication of which fields are most important for prediction or classification.

The weaknesses of decision tree methods:

a. Decision trees are less appropriate for estimation tasks where the goal is to predict the value of a continuous attribute.
b. Decision trees are prone to errors in classification problems with many classes and a relatively small number of training examples.
c. Decision tree can be computationally expensive to train. The process of growing a decision tree is computationally expensive. At each node, each candidate splitting field must be sorted before its best split can be found. In some algorithms, combinations of fields are used, and a search must be made for optimal combining weights. Pruning algorithms can also be expensive since many candidate sub-trees must be formed and compared.

3.5.3 XGBoost

XGBoost is an implementation of Gradient Boosted decision trees. XGBoost models majorly dominate in many Kaggle Competitions [21].

In this algorithm, decision trees are created in sequential form. Weights play an important role in XGBoost. Weights are assigned to all the independent variables which are then fed into the decision tree which predicts results. The weight of variables predicted wrong by the tree is increased and these variables are then fed to the second decision tree. These individual classifiers/predictors then ensemble to give a strong and more precise model.

XGBoost can work on regression, classification, ranking, and user-defined prediction problems.

3.5.4 KNN

KNN is widely known as an ML algorithm that doesn't need any training on data. This is much different from eager learning approaches that rely on a training dataset to perform predictions on unseen data.

KNN relies on observable data similarities and sophisticated distance metrics to generate accurate predictions. This technique may seem a bit counterintuitive and not trustworthy at first, but it's actually very reliable. It's popular in many fields. In computer vision field, KNN performs classification tasks. It handles image data well, and it's considered a fine option for classifying a bunch of diverse images based on similarities.

3.5.5 Naïve Bayes

Naive Bayes is among one of the very simple and powerful algorithms for classification based on Bayes Theorem with an assumption of independence among the predictors [22]. The Naive Bayes classifier assumes that the presence of a feature in a class is not related to any other feature. Naive Bayes is a classification algorithm for binary and multi-class classification problems.

3.6 Transformer model

3.6.1 Structure of transformer

The neural network is based and modified by the original transformer network [17].

The structure of the Transformer Neural Network is shown below:

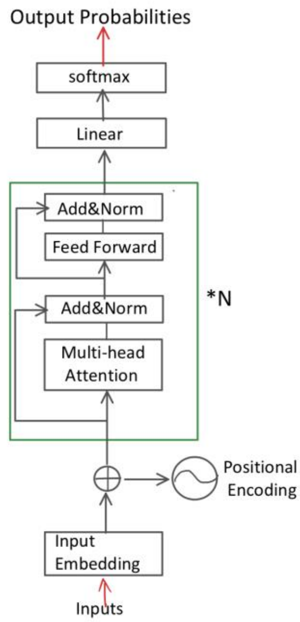

Fig3. Structure of transformer

3.6.2 Input Embedding Section

The embedding mechanism is based on the Sine and Cosine function. The positional encoding is applied to the input samples.

The positional encoding formula is shown below:

$$\begin{cases} PE_{(pos,2i)} = \sin\left(\dfrac{pos}{10000^{\frac{2i}{d_{model}}}}\right) \\ PE_{(pos,2i+1)} = \cos\left(\dfrac{pos}{10000^{\frac{2i}{d_{model}}}}\right) \end{cases}$$

3.6.3 Add & Norm Section

The add layer is based on the residue connection which is used to prevent the gradient vanishing. The input is put into a weight layer and sent to the ReLu activation function. The input will be processed by the following formula:

$$F(x) = F(x) + x$$

The layer normalization is to normalize the input of attention score to reduce the dimension and aids calculation effect. The formula is shown below:

$$LN(x_i) = \alpha \times \frac{x_i - u_L}{\sqrt{\alpha_L^2 + \epsilon}} + \beta$$

3.6.4 Attention Mechanism

The attention mechanism is the mimic of human attention system. The input will be divided into three different inputs Q, K and V. The similarity between Q and K inputs will be calculated and sent through the SoftMax activation function. The formula is shown below:

$$Weight(Q,K) = SoftMax\left(\sqrt{d_{model}} \times similarity(Q,K)\right)$$

$$Attention\ Score(Q,K,V) = SoftMax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

3.6.5 Input Samples Choice

The input sample used to train the Transformer neural network are masked slices, Range-Time slices, and Raw Data slices. To aids the classification accuracy, the input Q is used to send micro-Doppler features into the network, and the similarity measures the similarity between radar samples and micro-Doppler features.

3.6.6 Transformer Neural Network aided by micro-Doppler Features

The training data is limited due to the few numbers of radar recordings and the depth of the Transformer network which is restricted by the computation ability of the computer. As referencing the human beings' mechanism, people pay attention and focus more on the place which contains some specified features of the object to memorize and recognize some vital details.

Thus, the self-attention-based neural network can adopt this characteristic as well by adding features and comparing similarities between input objects and their corresponding features [20].

4. Results and Analysis

4.1 Improvements and best run

The best accuracy model is Quadric SVM. The original confusion matrix is shown below:

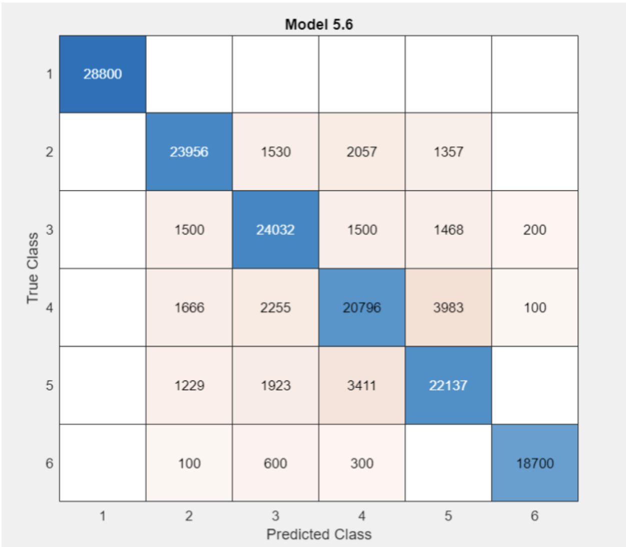

Fig4.

The total accuracy is 84.6%. According to the confusion matrix, Act1 and Act6 is recognized properly but for Act2 to Act5 a more detailed parameter adjustment is required in the RDD-SVM frame. Thus, a sophistic optimizer in Spark is used and adjusted according to the elbow method.

The elbow method is applied to improve the accuracy of the classification. The original plot of the elbow is shown below:

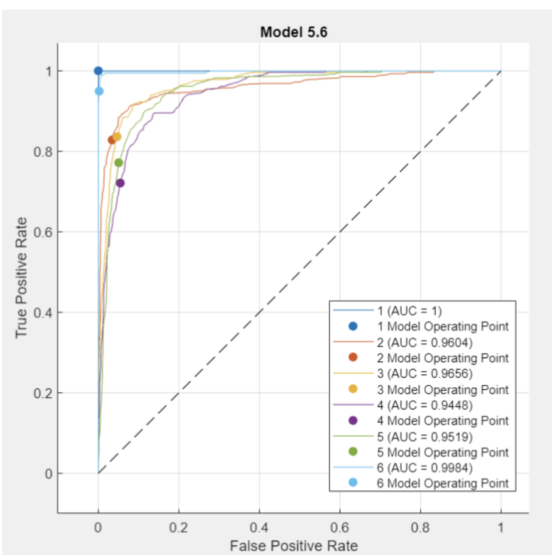

Fig5.

The parameters in the Cross Validator are adjusted and optimized as well. The final optimized parameters are produced after 100 epochs.

The final optimized accuracy is increased to 96.4%. The confusion matrix is shown below:

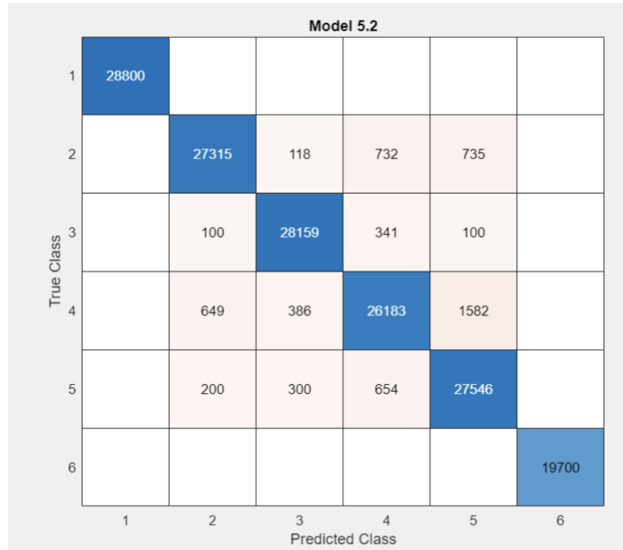

Fig6.

The classifier can clearly classify all six human activities with almost no classification error. The classifier can learn almost all the features of the six human activities which is capable to be used as the core part of older people's detection system.

The following parts illustrate the accuracy and confusion matrix of other statistic learning methods are shown in the following figures.

4.2 Results of decision Trees

For Decision Trees, the accuracy is 73.8% . The confusion matrix is shown below:

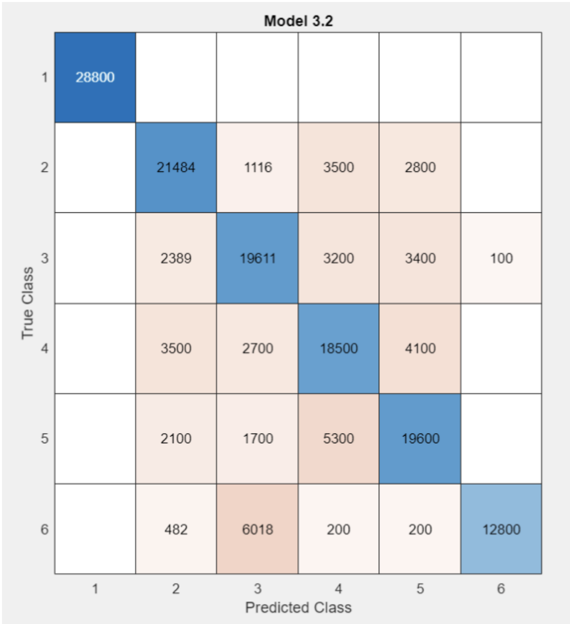

Fig7.

With the modification the accuracy reaches to 85.9%. The newly confusion matrix is shown below:

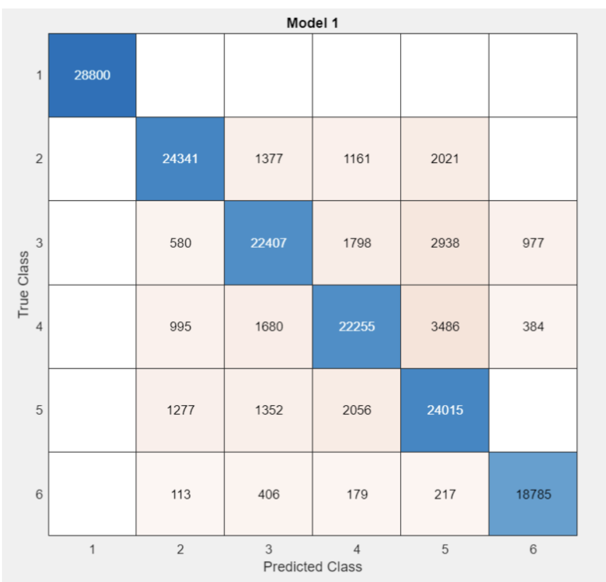

Fig8.

The decision tree cannot clearly differentiate between act2 to act7 due to the limitation of the depth of the tree to handle the overwhelm amount of data.

4.3 Results of Naïve Bayes

The accuracy of naïve Bayes model is 76.0%. The confusion matrix is shown below:

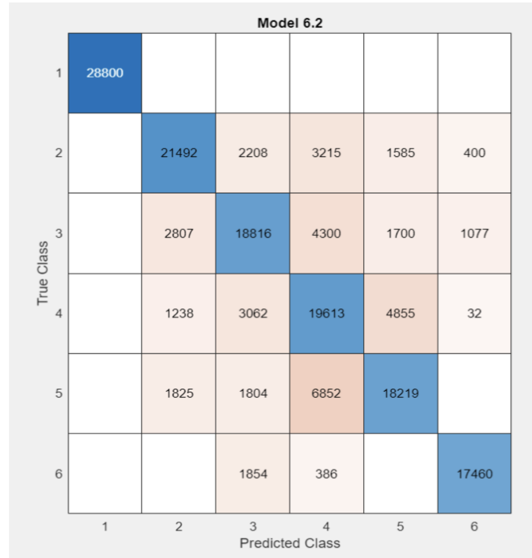

Fig9.

The optimization does not work on the naïve bayes.

4.4 Results of KNN

For KNN, the accuracy is 81.2%. The confusion matrix is shown below:

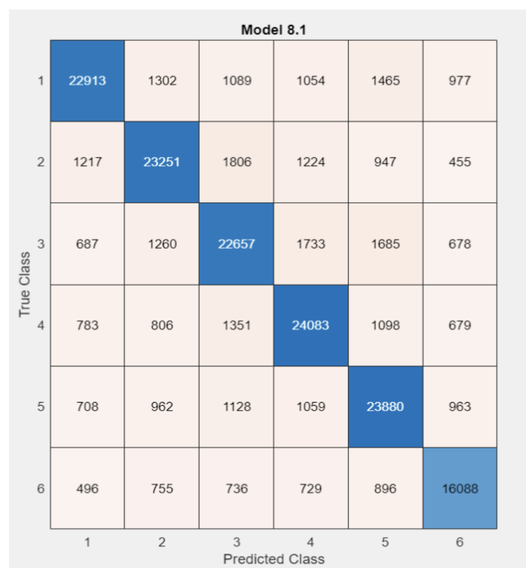

Fig10.

The classifier cannot distinguish all six human activities. The optimization method is increasing the neighbor numbers of the whole KNN model. Then, the optimized result has an accuracy of 84.6%.
The confusion matrix is shown below:

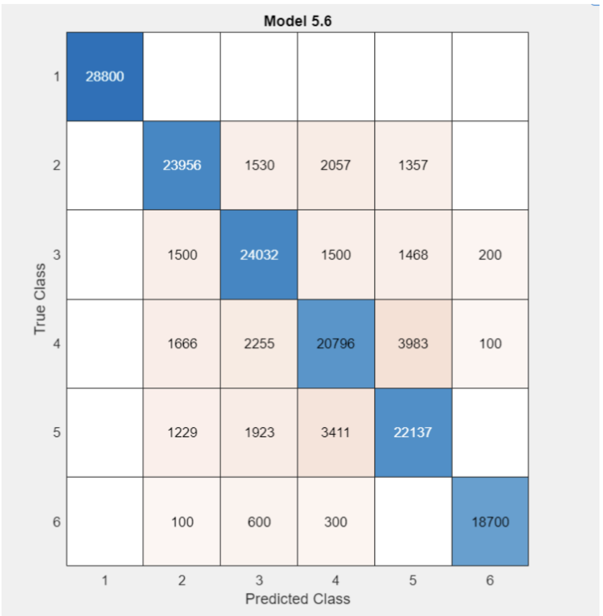

Fig11.

4.5 Results of Logistic Regression

For Logistic Regression, the accuracy is 76.2%. The confusion matrix is shown below:

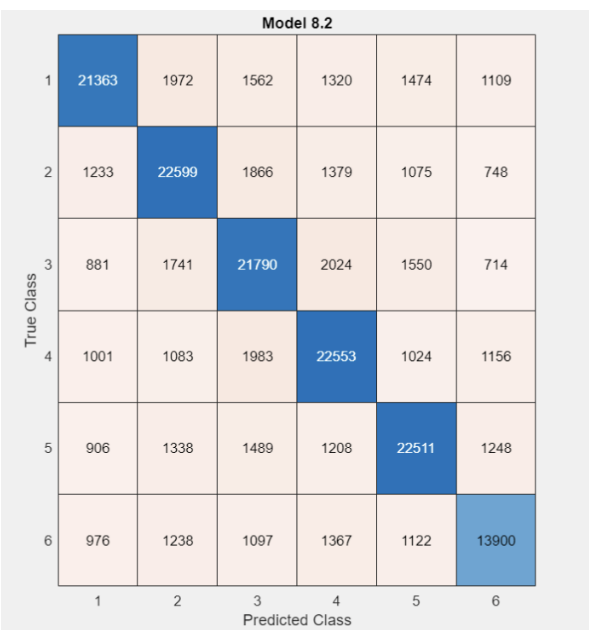

Fig12.

With the optimization, the accuracy reaches to 85.0%. The confusion matrix is shown below:

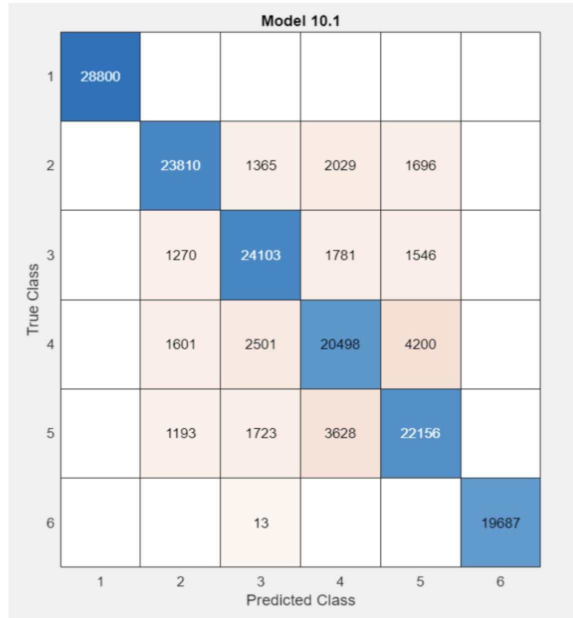

Fig13.

4.6 Results of XGBoost

For XGBoost, the best accuracy is 95.9%: The confusion matrix is shown below:

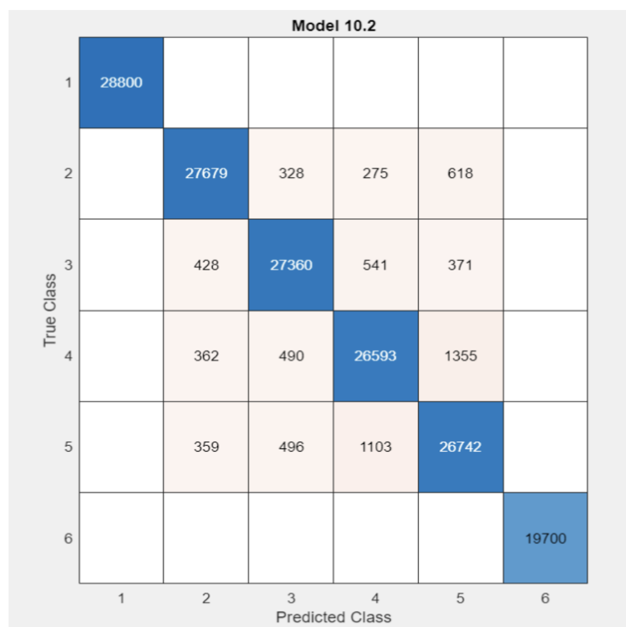

Fig14.

5. Overview and discussion

5.1 Limitation

The amount of sample size is exceeding the memory of the VM instance as the we use the Spark to construct the training dataset. The micro-Doppler features cannot be allocated to every micro-Doppler signal as original plan. The features cannot to split into slices according to different signals inch by inch. Because the ram limits these operations.

Thus, all features are just evenly split and attached to every signal. This method will not help the model to aid the accuracy and reduce the training time.

The second limitation is the training time. Every epoch consumes over 10 minutes due to limitation of the statistical learning method and the I/O speed of the cloud bucket.

5.2 Future Modification

The larger Ram and higher processing speed CPU is a must. Besides, a less computation complexity data processing methods are needed to reduce the ram usage in data reading, copying and hsatck. Besides, with improved RAM and CPU, a more sophistic method in allocating micro-Doppler features to the signals. (i.e. according the position of the signal in the whole micro-Doppler features to distribute the signal.)

References:

[1] G. Turner, “Security.org,” 2022. [Online]. Available: https://www.security.org/home-security-systems/best/smart-home/.

[2] S. Y. D, “WIDEBAND RADAR (ADVANTAGES AND PROBLEMS),” Ultrawideband and Ultrashort Impulse Signals, pp. 71-76, 2004.

[3] Y. Kim, “Human Activity Classification Based on MicroDoppler Signatures Using a Support Vector Machine,” IEEE Transactions on Geoscience and Remote Sensing, 2009.

[4] R. Basics, “Frequency-Modulated Continuous-Wave Radar (FMCW Radar),” 2020. [Online]. Available: https://www.radartutorial.eu/02.basics/Frequency%20Modulated%20Continuous%20Wave%20Radar.en.html.

[5] J. J. M. d. Wit, “Radar Micro-Doppler Feature Extraction Using the Singular Value Decomposition,” International Radar Conference, 2014.

[6] L. Cohen, Time-Frequency Analysis, Pearson College Div, 1995.

[7] V. Loan, Computational Frameworks for the Fast Fourier Transform., SIAM, 1992.

[8] M. Tom, Machine Learning, New York: McGraw Hill, 1997.

[9] R. Ingrid, Neural Networks Module, 2014.

[10] L. Floridi, “GPT 3: Its Nature, Scope, Limits, and Consequences,” Minds and Machines, 2022.

[11] V. Vapnik, The Nature of Statistical Learning Theory, New York: Springer, 1995.

[12] J. Soldatos, “The Embedded Machine Learning Revolution: The Basics You Need to Know,” wevolver, 04 feb 2021. [Online]. Available: https://www.wevolver.com/article/the-embedded-machine-learning-revolution-the-basics-you-need-to-know.

[13] D. Samuel, “A thorough review on the current advance of neural network structures,” Annual Reviews in Control, vol. 14, pp. 200-300, 2019.

[14] S. Hochreiter, “Long short-term memory,” Neural Computation, vol. 9, pp. 1735-1780, 1997.

[15] S. Yang, “The Inaugural Radar Challenge 2020: Survey on Machine Learning Based Models for Human Activity Classification,” IET International Radar Conference, 2021.

[16] Z. Li, “Multi-domains based Human Activity Classification in Radar,” IET International Radar Conference, 2020.

[17] A. Vaswani, “Attention Is All You Need,” *arXiv*, 2017.

[18] A. Khairuddin and I. Saeh, "Decision Tree for Static Security Assessment Classification," in 2009 International Conference on Future Computer and Communication (ICFCC), Kuala Lumpar, 2009 pp. 681-684.

[19] Fioranelli, F.,Shah, S. A.,Li, H.,Shrestha, A., Yang, S.andLe Kernec, J., “Radar signatures of human activities”, (2019)

[20] L. Zhang, “Polarimetric HRRP recognition based on feature guided Transformer model,” Electronics Letters, 2021.

[21] Tianqi Chen, Carlos Guestrin, “XGBoost: A Scalable Tree Boosting System”, arXiv, 2016.

[22] 1.9. Naive Bayes — scikit-learn 1.2.0 documentation