Comparative Classifier Model Approach on Human Activity Recognition from Ambient Intelligence Dataset

Mahbuba Tasmin
ID : 1610064042

Sharif Uddin Ruman
ID: 1611557642

N.M. Shihab Islam
ID: 1420339042

Email:
mahbuba.tasmin@northsouth.edu

Abdur Raufus Saleheen
ID: 1610472642

Sifat Jahan
ID: 1611702642

***Abstract*— Human activity Recognition (HAR) is a challenging time series classification task based on neural network modeling to classify the activity of new unseen subjects from the collected sensor data. It involves predict the movement/activities of a person based on time series data collected from accelerometer of a smartphone or motion sensors in indoor setup. In this paper, we present a comparative model approach on classification methods from the Ambient Sensor Dataset from UCI machine learning repository to recognize human activities. Before the classifier approach, we have executed extensive data preprocessing and feature selection to produce our selected dataset for the classification. The accuracy of the three classifier models (Decision Tree, Random Forest and Nearest Neighbor) shows different accuracy scores for datasets with and without feature selection. The research output of this paper presents the necessity of data preprocessing and significant feature selection for achieving greater accuracy score for noisy time-series data of HAR activity.**

Keywords—Human Activity Recognition, Time Series Data, Activity Classification, Feature Engineering

I. INTRODUCTION

Technology's advancement has blessed mankind with smart world that consists of smart living appliances namely smart home devices, smartphones, wearables and other forms of applications, which has tremendously influenced human lifestyle and is continuing to shape the futuristic lifestyle as well. These technologies has empowered independent lifestyle of an individual, thus significantly reducing dependency on other people. With these smart technologies, the concept of Ambient Assisted Living (AAL) emerged. Ambient Assisted Living presents a system consisting of smart devices, home appliances, wireless networks primarily for healthcare monitoring and smart home living. This concept provides the solution to ensuring a safe and quality life for older citizens through preventing, curing and improving wellness and health conditions of older adults by assisting them in living comfortably in their preferred environment. Another sister concept in this regard is Ambient Intelligence (AML). It presents the ability of a computing system to sense its surrounding and interact with people around.

The concept of Ambient Assisted Living (AAL) and Ambient Intelligence (AML) originates at the first place from the advancement of Human Activity Recognition (HAR) through wireless sensor network and the Internet of Things (IoT). Data records from different sensor readings has paved the way to identify human activities separately and is leading to smart home systems consequently. Most HAR systems are based on camera or computer vision or wearable sensors.

One major feature of activity recognition is change detection via detecting sudden change in statistical metrics (e.g. Mean and Covariance), which represents a change in time series data within an indoor environment. Precise manipulation of the derived metrics using a robust algorithm would decide the class of activity performed within a timeframe. In general, activity recognition is a vital component of context-aware systems, which provides the understanding of the smart home applications to understand user requirement and adapt to the various circumstances. Nevertheless, a real-time indoor HAR system in a real environment is often limited by the constraints of indoor environments and makes it difficult to build a robust and scalable system.

Computer vision based HAR systems are useful for large coverage and pedestrian activity recognition. To eliminate the potential privacy issue related to camera based computer vision system in an indoor environment setup, HAR solutions at recent years are based on wearable sensors or devices including smartphones. Wearable approach is sometimes rendered restrained and potential limitations since the user need to always equip the sensing device while recording data, which doesn't support seamless activity record process. In addition, the wearable approach requires transition between different positions of the user need to be perceived since the system depends on the target to determine the location of the wearable device with respect to the performed activity.

On the other hand, In the indoor environment, intelligent HAR system perceives the state of the physical environment and the interacting resident using sensors, reasons about the recorded data and applies Ambient Intelligence to take actions to achieve specified targets. During recording, embedded sensors in the home collects readings while residents independently perform their usual activities. Sensor-data is collected and stored in a database and later analyzed to generate target information such as patterns, predictions and transitions. The process of discerning relevant activity information from sensor streams is a non-trivial task and introduces many difficulties for traditional machine learning algorithms. These difficulties include spatio-temporal variations in activity patterns, sparse occurrences for some activities, and the prevalence of sensor data that does not fall into predefined activity classes.

To this end, the present work is motivated to classify five distinct activities (Watch TV, Read, Phone, Cook, and Eat) from the dataset of 12 pre-defined activities including unlabeled activity namely "other activity", on the basis of the UCI Machine Learning Repository dataset "*Human Activity Recognition from Continuous Ambient Sensor Data Dataset*" from Washington State University.. The motivation is to precisely classify the activities while reducing the computational requirements through exhaustive data preparation. This originates from the idea to allow human activity recognition with less costs involved in computation so that we can incorporate the concept in the perspective of Bangladesh. The dataset is preprocessed, features with statistically significant values have been selected and finally we have applied three different classifier models to present a comparison output of the accuracy level.

The major contributions of the present paper include:

- Data preprocessing of the large CASAS dataset through Principal Component Analysis and Linear Discriminant Analysis
- Feature Selection based on statistical significance
- Classifier models comparison on the pre-processed dataset

The paper follows the following structure: Section II presents the related works on the research objectives. Section III presents Methodology, where data preprocessing and feature selection approaches are discussed and classifier model approach follows the discussion. Section IV consists of the results from three consecutive steps of the research. In the following Section V, Discussion presents the observation and areas for further improvement.

II. Related Work

The research field of activity recognition is quite large considering the combination of embedded sensors, different environmental setups and algorithms to detect activity points. Hence, there are number of approaches explored in this field.

Naïve Bayes classifiers have produced satisfactory output for offline detection of activities […..]. Decision trees are used to learn logical transition of the activity […] while Gu et al […] utilizes KNN to detect mode sensor values associated with activities which helps in recognition.

Probabilistic graph based Markov models […..], conditional random fields[….], Bayesian network have been used successfully to recognize activities even in complex environments. Studies have found that probabilistic graphs along with neural network approaches […] are significant at mapping pre-segmented sensor sequence to activity labels.

Different types of sensor data are proven to be effective for classifying different types of activities. Ambulatory movements (e.g. Walking, Running, Standing, Sitting, Climbing Stairs and Falling) are classified in […….] using accelerometer placed on the body. Recently smartphones with accelerometer and gyroscope sensors are used as wearable device to recognize gesture and motion patterns […..].

More complex activities that requires more information than body movement, in that case the user's interaction with key objects in the environment is recorded […]. Shake sensors or RFID tags are tagged with the object and are selected based on the targeted activities. Environment sensors such as motion detector, light sensor, door contact sensors are used to recognize daily activities in other researches […].

At realistic activity recognition tasks, the recognizing activities are performed with interleaved activities […], embedded errors […], and concurrent activities performed by multiple individuals in the setup […]. Detecting activities in free movement setup, where the residents perform usual daily routines in a smart home environment was the next step of advancement […]. These recorded datasets have required on manual labelling to segment and analyze the data. Recent further advancements of activity recognition has brought automated segmentation […], spontaneous selection of objects to tag and monitor […], and for transfer of pre-learned activities to new environment setup

An on-body approach is proposed by Kunze et al. […] that perceives if the target is walking and then apply pre-selected sensor reading pattern to predict the actual target's position. This approach involves attachment of sensor onto the target and hence, the consequent dataset is small. On-body approach with device localization approach presented by Sztyler et al. […] predicts the target on-body position with F-measure calculation and cross-subject activity recognition.

Dedicated HAR architectures use various methods to perceive the complex concerns from recognizing sequential and concurrent human activities. Two key approaches are followed in HAR: data-driven and knowledge-driven technique. Naïve Bayes (NB) classifiers, Decision Trees, Hidden Markov Models, Bayesian Networks and Support Vector Machine (SVM) classifier are the machine learning techniques and probabilistic approaches in Data-driven method. The algorithms work on inductive reasoning to detect human activities in data-driven approach. Existing works including data-driven technique utilizes supervised approach using manually labeled data for training. The approach is restrained by complex method and additional computational cost.

The unsupervised approaches are often restricted by low performance in comparison with the supervised approach in indoor home environment. In the knowledge-based HAR, activities are modeled with their contextual information in the common ground as new activity record is detected via deductive reasoning. The construction of a common ground to present the set of concepts along with their relationships in a machine-interpretable approach is a restraint of knowledge-based HAR. Data-driven techniques are useful for detecting basic distinctive activities, on the other hand unsupervised approach is suitable for creating probabilistic models with expected accuracy score.

A knowledge-based approach utilizing the inter-frame algorithm convolutional neural network is applied in Chen et al. […], where distinguishing features are collected through cameras and learnt, filters non-target objects and estimate skeleton sequence from RGB images.

III. Data Source

The dataset of the project is collected from UCI Machine Learning Repository, *Human Activity Recognition from Continuous Ambient Sensor Data Dataset*. The dataset is fairly new, published on 20th September, 2019.

This dataset represents ambient data collected in homes with volunteer residents with their usual daily activities at home. **Ambient PIR motion sensors, door/temperature sensors, and Light Switch sensors** are placed throughout the home of the volunteer. The sensors are placed in locations throughout the home that are related to specific target activity of daily living.

The classification task is to predict the activity that is occurring in the smart home and being observed by the ambient sensors. The sensors communicate using the **ZigBee Pro protocol**.

The original format captured from the sensors is provided, as well as the feature vector we generate using a sliding window of 30 sensor events. Each annotated data file (ex: csh101/csh101.ann.txt) has a corresponding feature vector CSV file (ex: csh101/csh101.ann.features.csv). Most of the sensor data files contain labels for **two months of the collection period**, though some contain labels for extended time periods. The motion sensors determines the time of motion occurrence in the range of the sensor. The motion sensor reports 1/0 depending on the record of motion activity. The transition period between on and off status is roughly 1.25 seconds. For continuous activity record beyond the threshold time, the sensor won't record 0 until 1.25 seconds after the activity has ceased.

The smart home layout and sensor placement from the original formats is found in the included sensor map for each smart home. One example layout is attached below:

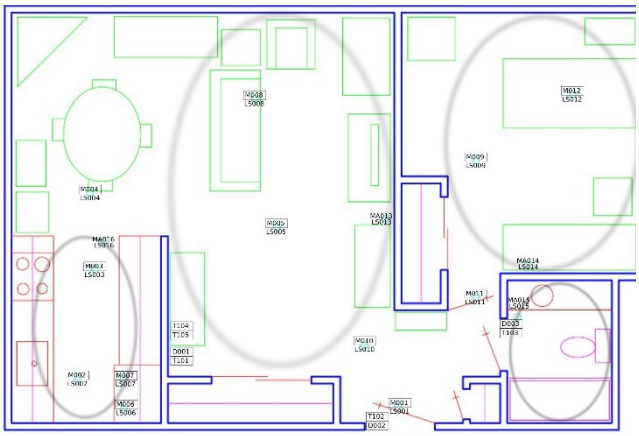

Figure 1: Sensor Layout of One of the Volunteer Resident House

The dataset is collected under the lead of Diane J. Cook from School of Electrical Engineering and Computer Science at Washington State University, and the other creators are Aaron S. Crandall, and Brian L. Thomas.

The key features of the scraped dataset is presented in the below table:

Data Set Characteristics	Multivariate, Sequential, Time Series	Number of Instances	4475631
Attribute Characteristics	Integer, Real	Number of Attributes	37
Associated Tasks	Classification	Missing Values	Yes
Area	N/A		

Table 1: Key features of the Scraped Dataset

IV. METHODOLOGY

We have scraped the dataset for this research from the UCI dataset, for the five selected activities (Watch TV, Read, Phone, Cook, and Eat). The dataset attributes is presented in table 1. The whole work is divided into three major segments- Data Preprocessing, Feature Selection and Classifier Model execution. In this section, we discuss about the working principle of each of the segment.

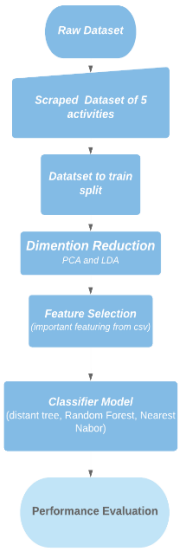

Figure 2: Flowchart of Methodology

A. Data Preprocessing

The scraped dataset is standardized and divided into test and train set (split = 0.3). The scatter plot is presented in Figure 2.

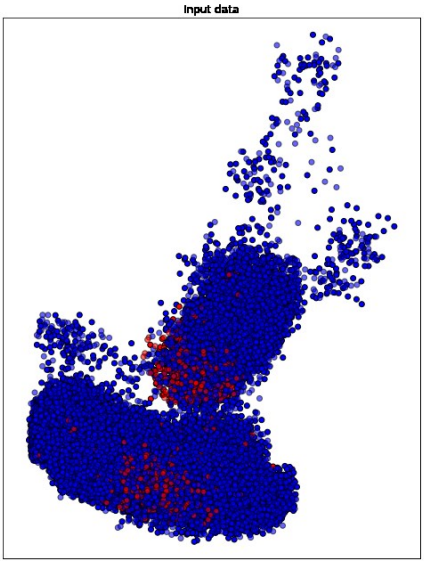

Figure 3: Input Data Distribution of Test and Train Split

B. *Dimensionality Reduction*

We have applied dimensionality reduction through Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) methods to make the data more visually understandable.

1) Principal Component Analysis

Principal Component Analysis (PCA) applied to this data identifies the combination of attributes (principal components, or directions in the feature space) that account for the most variance in the data. Here we plot the different samples on the 2 first principal components.
The feature of the dataset is standardized first through *StandardScaler( )* and reduced to dimension of 2. Here is the visual graph output of the dataset:

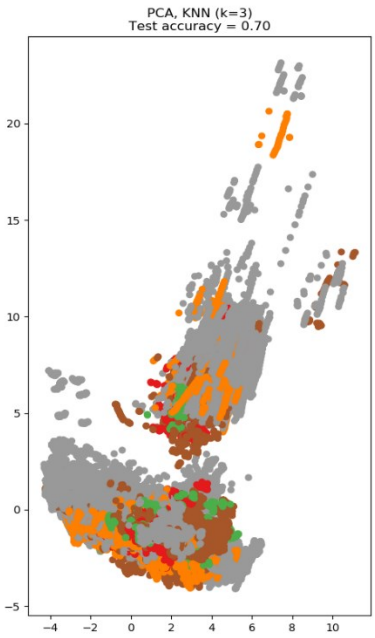

Figure 4: PCA Presentation with 70% Accuracy

The PCA variance presents 70% accuracy on the test dataset, which is significantly low since the dimensions are reduced from 37 to principal 2 dimensions.

2) Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) tries to identify attributes that account for the most variance between classes. In particular, LDA, in contrast to PCA, is a supervised method, using known class labels.

The LDA accuracy score outperforms PCA score, with a 77% accuracy score. LDA finds centroid of each data point and projects the cluster of data points.

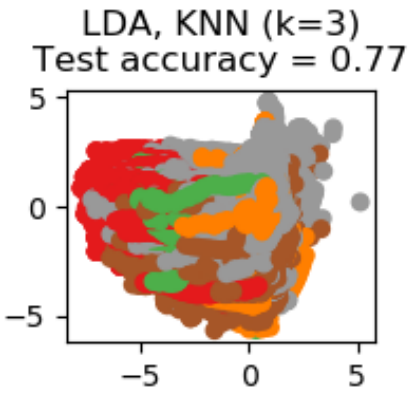

Figure 5: LDA Presentation with 77% Accuracy

C. *Feature Selection*

The primary goal of this research work is to activity detection through recognizing the pattern of data mined in DATSET NAME [dataset paper citation]. Primarily 5 distinct activities have been selected to train for pattern recognition purpose. The end goal of the research work is to inject adversarial attack on the model to confuse the network and identify the actual activity after the injection. To suggest more amicable work based on such data, the research team is exploring variety of fields in health, administration and security issues where such dataset generation and model implementation will be useful for activity recognition. Before fitting the dataset into the explored classifiers for activity recognition several preprocessing techniques have been applied for statistical analysis of the attributes the dataset to reduce those number of features that do not contribute to training. The research team believes the feature selection approach not only reduce the number of training time and computational cost but also will reduce the variance of the model, thus avoiding overfitting. The following section describes the feature selection techniques that the research team has applied for feature selection and the theoretical background of the techniques.

A. Low Variance Feature Removal:

The low variance feature selection technique removes the features which is found to be constant mostly. The constant value of a feature is not very interesting to find pattern and can be removed from the dataset. For dataset with large attributes the scikit-learn library automatically identifies the features which have the lowest variance. The heuristic approach before running the feature selection techniques is to use a threshold value to use as cut-off. The feature elimination is run when any features comes beneath this threshold value. On the given threshold the library computes the covariance against each tuple of the dataset and generates the result. The research team has kept a threshold of 80% as the threshold.

B. L1 Based Feature Selection:

In SVM the parameter C controls the sparsity of the vectors. The smaller C is the fewer features elected. In large number of samples, the L1 model perform at random where it depends on the number of non-zero coefficients, the logarithm number

of features, the amount of noise, the smallest absolute value of non-zero coefficients and the structure of the design matrix. The design matrix must contain the property of not being too correlated.

C. Tree-based Feature Selection:
The tree-based estimators are used to compute the importance of features and to discard the irrelevant features.

D. Feature Selection with Random Forest:
Random forest classifier uses the tree-based strategies to rank the features for improving purity of the node.

Original Set of Features	Selected Features with Low Variance Feature Removal	Selected Featured with L1 Based Feature Selection	Selected Features with Tree-based Feature Selection	Feature Selection with Random Forest
lastSensorEventHours	lastSensorEventHours	lastSensorEventHours	lastSensorEventHours	lastSensorEventHours
lastSensorEventSeconds	lastSensorEventSeconds	lastSensorEventSeconds	lastSensorEventSeconds	lastSensorEventSeconds
lastSensorDayOfWeek	lastSensorDayOfWeek	lastSensorDayOfWeek	lastSensorDayOfWeek	
windowDuration	windowDuration	windowDuration		windowDuration
timeSinceLastSensorEvent	timeSinceLastSensorEvent	timeSinceLastSensorEvent		
prevDominantSensor1	prevDominantSensor1	prevDominantSensor1	prevDominantSensor1	
prevDominantSensor2	prevDominantSensor2	prevDominantSensor2		
lastSensorID	lastSensorID	lastSensorID		
lastSensorLocation	lastSensorLocation	lastSensorLocation	lastSensorLocation	
lastMotionLocation	lastMotionLocation	lastMotionLocation	lastMotionLocation	lastMotionLocation
complexity	complexity	complexity		
activityChange	activityChange	activityChange		
areaTransitions	area transitions	area transitions		
numDistinctSensors				
sensorCount-Bathroom	sensorCount-Bathroom	sensorCount-Bathroom		
sensorCount-Bedroom	sensorCount-Bedroom	sensorCount-Bedroom		
sensorCount-Chair	sensorCount-Chair	sensorCount-Chair		
sensorCount-DiningRoom	sensorCount-DiningRoom	sensorCount-DiningRoom		
sensorCount-Hall	sensorCount-Hall	sensorCount-Hall		
sensorCount-Ignore	sensorCount-Ignore	sensorCount-Ignore		
sensorCount-Kitchen	sensorCount-Kitchen	sensorCount-Kitchen	sensorCount-Kitchen	sensorCount-Kitchen
sensorCount-LivingRoom	sensorCount-LivingRoom	sensorCount-LivingRoom	sensorCount-LivingRoom	
sensorCount-Office	sensorCount-Office	sensorCount-Office		sensorElTime-Bedroom
sensorCount-OutsideDoor	sensorCount-OutsideDoor	sensorCount-OutsideDoor		
sensorCount-WorkArea	sensorCount-WorkArea	sensorCount-WorkArea		

Table 2: Selected Features through Feature Selection Approach

D. Feature Importance

Feature importance calculates the score for each feature in a dataset through use of forests of trees. The red bars present the feature importance of the forest, along with inter-trees variability. Here we have applied the score calculation on 37 column attributes through Extra Tree Classifier and Random Forest Classifier.

1) Extra Tree Classifier

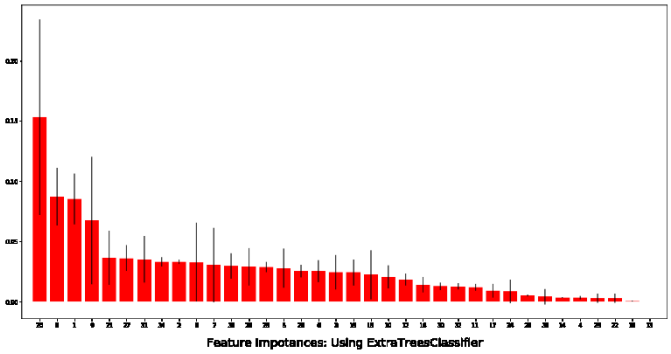

Figure 6: Extra Tree Classifier Feature Score

Feature	Score
1. feature 20	(0.153283)
2. feature 0	(0.087287)
3. feature 1	(0.085212)
4. feature 9	(0.067567)
5. feature 21	(0.036598)
6. feature 27	(0.036238)
7. feature 31	(0.035192)
8. feature 34	(0.033122)
9. feature 2	(0.033049)

2) Random Forest Classifier

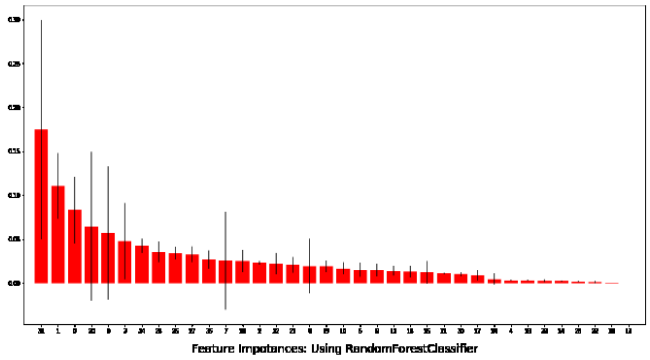

Figure 7: Random Forest Classifier Feature Score

Top 10 features through this approach is

Feature	Score
1. feature 31	(0.149931)
2. feature 1	(0.116110)
3. feature 20	(0.091917)
4. feature 0	(0.080503)
5. feature 3	(0.055434)
6. feature 9	(0.045336)
7. feature 34	(0.041218)
8. feature 25	(0.035567)
9. feature 26	(0.035500)
10. feature 27	(0.031554)

E. Backward Elimination Output

Variance Threshold is a simple baseline approach to feature selection. It removes all features whose variance doesn't meet some threshold. By default, it removes all zero-variance features, i.e. features that have the same value in all samples. With this technique, 5 features have been. found which can be reduced. Univariate feature selection works by selecting the best features based on univariate statistical tests. It can be seen as a preprocessing step to an estimator. The two columns that are found most significant through this technique are "lastSensorEventSeconds, sensorElTime-Bedroom."

	coef	std err	t	P>|t|	[0.025	0.975]
const	6.402e-11	3.55e-12	18.035	0.000	5.71e-11	7.1e-11
x1	0.0515	0.017	3.028	0.002	0.018	0.085
x2	-2.103e-05	4.74e-06	-4.432	0.000	-3.03e-05	-1.17e-05
x3	-0.0165	0.003	-5.931	0.000	-0.022	-0.011
x4	-3.727e-06	1.46e-05	-0.255	0.799	-3.24e-05	2.49e-05
x5	-0.0002	7.95e-05	-2.937	0.003	-0.000	-7.76e-05
x6	0.0177	0.003	7.012	0.000	0.013	0.023
x7	0.0103	0.002	4.486	0.000	0.006	0.015
x8	-0.0238	0.002	-13.595	0.000	-0.027	-0.020
x9	-0.0238	0.002	-13.595	0.000	-0.027	-0.020
x10	-0.0261	0.004	-6.118	0.000	-0.034	-0.018
x11	0.4485	0.016	27.341	0.000	0.416	0.481
x12	0.0450	0.021	2.145	0.032	0.004	0.086
x13	0.0022	0.002	0.899	0.369	-0.003	0.007
x14	6.307e-14	8.49e-15	7.427	0.000	4.64e-14	7.97e-14
x15	-0.0228	0.004	-5.968	0.000	-0.030	-0.015
x16	-0.0272	0.003	-8.212	0.000	-0.034	-0.021
x17	-0.0905	0.003	-34.644	0.000	-0.096	-0.085
x18	0.0667	0.001	66.164	0.000	0.065	0.069
x19	3.908e-17	2.52e-18	15.507	0.000	3.41e-17	4.4e-17
x20	0.0066	0.001	5.101	0.000	0.004	0.009
x21	0.0442	0.001	40.071	0.000	0.042	0.046
x22	0.0027	0.001	2.399	0.016	0.000	0.005
x23	4.959e-17	5.55e-18	8.931	0.000	3.87e-17	6.05e-17
x24	-0.0009	0.005	-0.191	0.849	-0.011	0.009
x25	0.0214	0.001	17.513	0.000	0.019	0.024
x26	-3.671e-05	5.43e-06	-6.754	0.000	-4.74e-05	-2.61e-05
x27	6.864e-06	5.43e-06	1.265	0.206	-3.77e-06	1.75e-05
x28	4.84e-06	1.84e-07	26.352	0.000	4.48e-06	5.2e-06
x29	9.05e-06	1.23e-06	7.345	0.000	6.63e-06	1.15e-05
x30	5.531e-06	3.07e-07	18.035	0.000	4.93e-06	6.13e-06
x31	5.495e-06	1.96e-07	28.003	0.000	5.11e-06	5.88e-06
x32	-9.561e-05	9.06e-06	-10.552	0.000	-0.000	-7.79e-05
x33	7.301e-05	1.06e-05	6.898	0.000	5.23e-05	9.38e-05
x34	5.531e-06	3.07e-07	18.035	0.000	4.93e-06	6.13e-06
x35	-3.97e-06	1.19e-06	-3.322	0.001	-6.31e-06	-1.63e-06
x36	-1.495e-05	8.93e-07	-16.750	0.000	-1.67e-05	-1.32e-05

Omnibus:	1972.542	Durbin-Watson:	0.063
Prob(Omnibus):	0.000	Jarque-Bera (JB):	8908.507
Skew:	0.167	Prob(JB):	0.00
Kurtosis:	5.698	Cond. No.	2.79e+19

Figure 8: Backward Elimination Output

F. Classifier Comparison

The classifier comparison presents a set of classifying methods in scikit-learn on our dataset. The point of this comparision is to illustrate the nature of decision boundaries of different classifiers.

Particularly in high-dimensional spaces, data can more easily be separated linearly and the simplicity of classifiers such as naive Bayes and linear SVMs might lead to better generalization than is achieved by other classifiers.

The plots show training points in solid colors and testing points semi-transparent. The lower right shows the classification accuracy on the test set. After feature selection is done, two datasets are generated based on the Tree-based and Random-forest based feature selection. The L1-based and Low-variance approach don't reduce the dimension significantly and hence we have discarded those results.

We have tested on **Nearest Neighbor, Decision Tree and Random Forest Classifiers** to run on the two datasets.

V. PERFORMANCE EVALUATION

In this part of the report, we present the outputs of preprocessing and other parts of the project.

A. Dataset without Feature Selection

First, we have run the classifier models in the dataset without feature selection approach. The below figures represent the model accuracy of each of the classifier on the dataset.

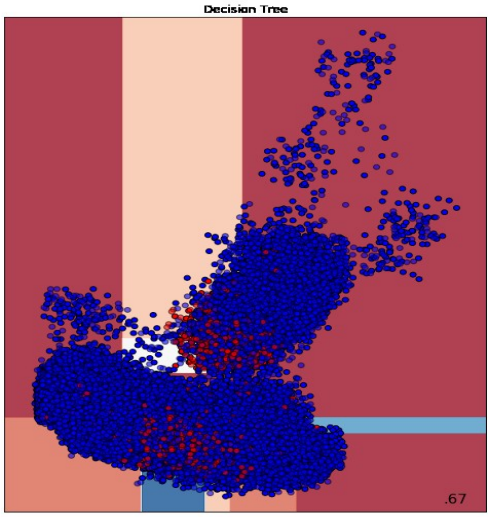

Figure 9: Decision Tree Accuracy 67%

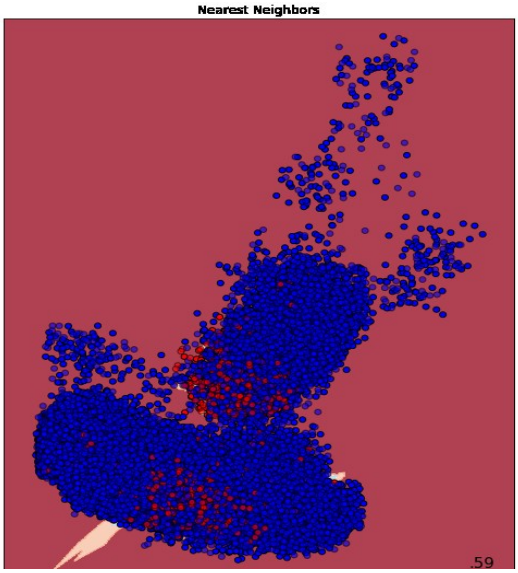

Figure 10: Nearest Neighbor Accuracy 59%

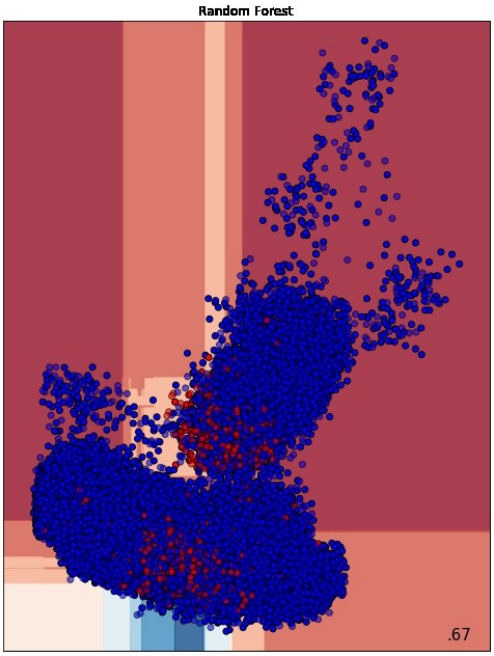

Figure 11: Random Forest Accuracy 67%

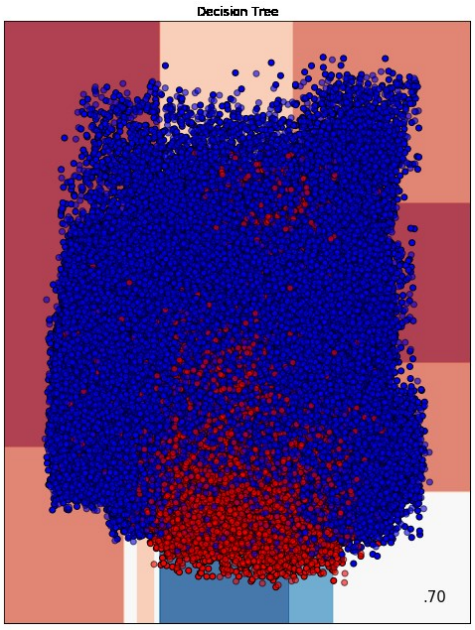

Figure 13: Decision Tree Accuracy 70%

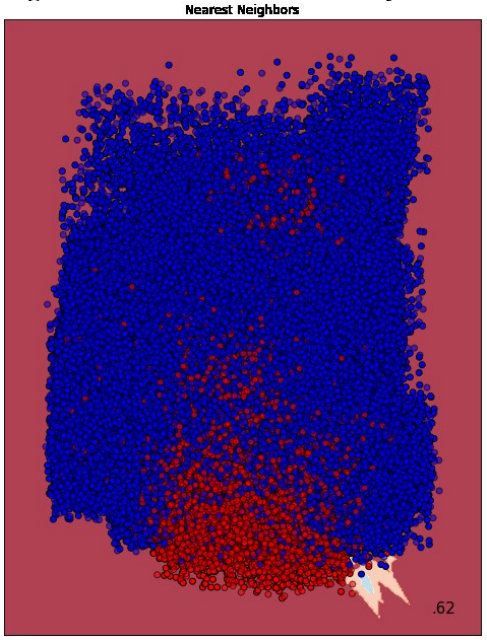

Figure 14: Nearest Neighbor Accuracy 62%

B. Dataset from Tree-based Classifier Feature Selection

The tree-based classifier feature selection selects top few attributes and produces a new dataset based on the selection. Below figures represent the new dataset distribution based on the selection and the following classifier model accuracy scores.

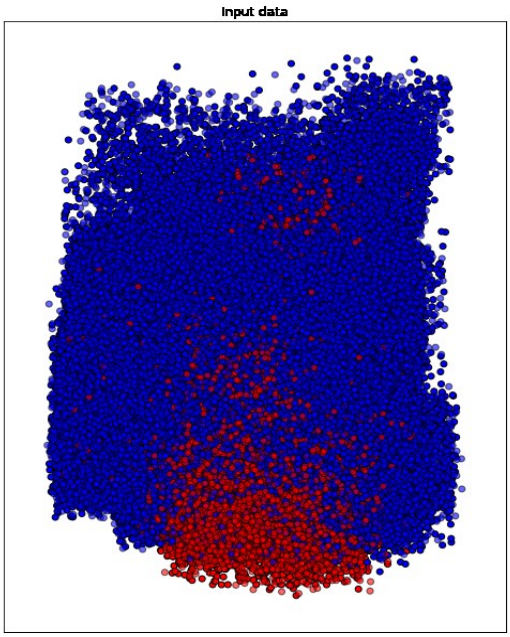

Figure 12: Input Distribution of Tree-Based Feature Selection

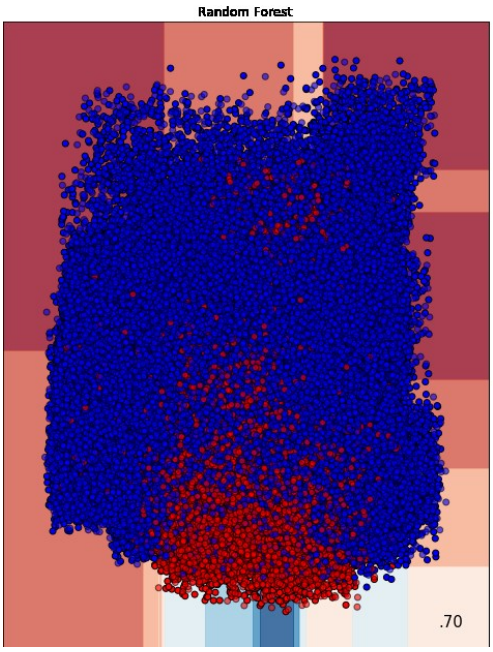

Figure 15: Random Forest Accuracy 70%

C. Dataset From Random Selection Classifier Feature Selection

The Random-forest classifier feature selection selects top few attributes and produces a new dataset based on the selection. Below figures represent the new dataset distribution based on the selection and the following classifier model accuracy scores.

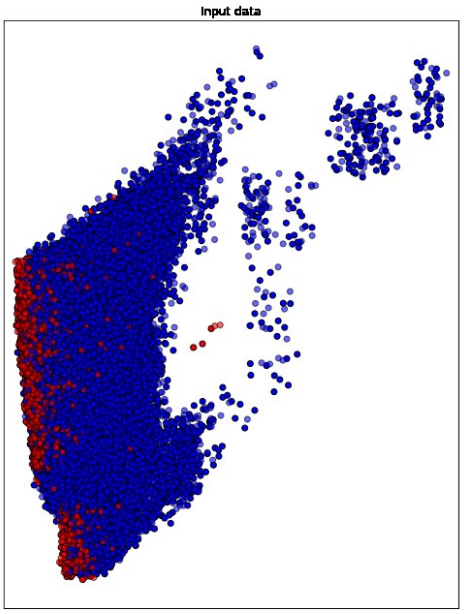

Figure 16: Input Distribution of Random-Forest Based Feature Selection

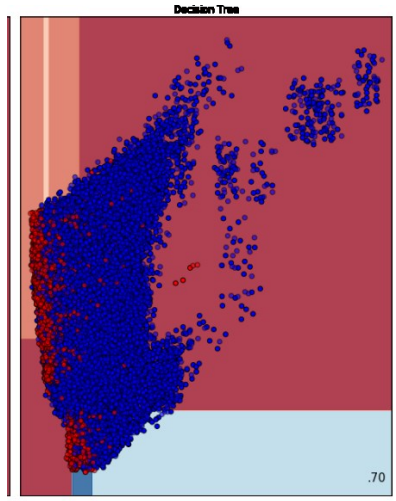

Figure 17: Decision Tree Accuracy 70%

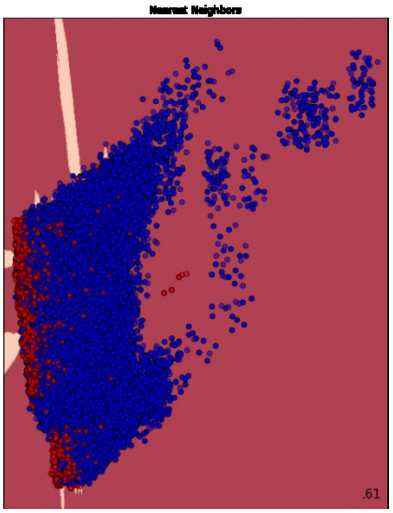

Figure 18: Nearest Neighbor Accuracy 61%

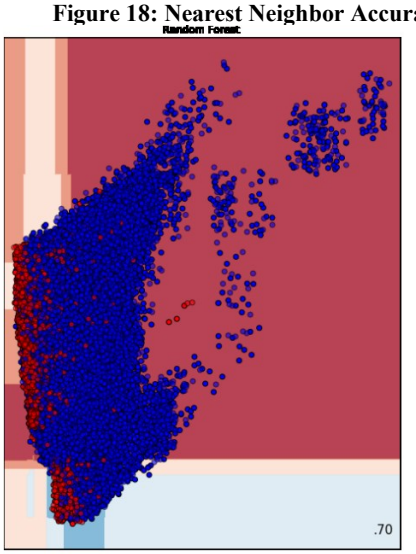

Figure 19: Random Forest Accuracy 70%

VI. Conclusion

In this paper, we present a comparative model approach to classify five selected activities from the dataset. The classifier models show significant changes after application of precise data preprocessing and feature selection approach. The accuracy score increased by 10% from the raw dataset when feature selection is applied. The future work includes preparing neural network approach to classify the activities and on the basis of the model, we aim to produce a robust time-series model to handle adversarial attack.