

Smartphone-based Recognition of Human Activities using Shallow Machine Learning

Maha Mohammed Alhumayyani¹
Information Systems Department
Faculty of Computer and
Information Sciences
Ain Shams University, Egypt

Dr. Mahmoud Mounir²
Faculty of Computer and
Information Sciences
Ain Shams University, Egypt

Prof. Dr. Rasha Ismael³
Vice-dean fea graduate studies and
research, Faculty of Computer and
Information Sciences
Ain Shams University, Egypt

Abstract—The human action recognition (HAR) attempts to classify the activities of individuals and the environment through a collection of observations. HAR research is focused on many applications, such as video surveillance, healthcare and human computer interactions. Many problems can deteriorate the performance of human recognition systems. Firstly, the development of a light-weight and reliable smartphone system to classify human activities and reduce labelling and labelling time; secondly, the features derived must generalise multiple variations to address the challenges of action detection, including individual appearances, viewpoints and histories. In addition, the relevant classification should be guaranteed by those features. In this paper, a model was proposed to reliably detect the type of physical activity conducted by the user using the phone's sensors. This includes review of the existing research solutions, how they can be strengthened, and a new approach to solve the problem. The Stochastic Gradient Descent (SGD) decreases the computational strain to accelerate trade iterations at a lower rate. SGD leads to J48 performance enhancement. Furthermore, a human activity recognition dataset based on smartphone sensors are used to validate the proposed solution. The findings showed that the proposed model was superior.

Keywords—Data preprocessing; data mining; classification; genetic programming; Naïve Bayes; decision tree

I. INTRODUCTION

The aim of human action recognition (HAR) is to recognize activities extracted from a number of observations concerning the behavior and environmental conditions of subjects. A lot of applications for HAR research include video monitoring, healthcare and contact with human-computer. HAR uses sensors influenced by human movement for the classification of an operation of the individual. Both users and sensors of smartphones expand as users also bring their smartphones. HAR seeks to identify activities arising from a variety of observations concerning the behavior and environmental conditions of subjects.

Sensors can help patients always record and track and automatically report if abnormal behavior has been detected by a huge quantity of resources. The research benefits from other applications, including the human survey method and position predictor. Many experiments have successfully established wearable sensors with a low error rate, but most work is conducted in labs with very limited settings. Readings from

many body sensors achieve a low error rate, but in reality the complex environment cannot be achieved [1].

The efficiency of the human action mechanism can be deteriorated by several challenges. One is that the extracted features need to generalize many variations in order to address the challenges of action recognition, including individual appearances, viewpoints and histories. In addition, the relevant classification should be guaranteed by those features. The creation of a lightweight, precise device on Smartphones that can detect human activities and reduce labelling time and burden is another challenge.

The main purpose of this paper is to reliably detect the type of physical activity that the user conducts using the phone sensors. This involves an analysis of existing solutions, finding ways to strengthen them and finding a new approach to the issue. Furthermore, a human activity recognition dataset based on smartphone sensors are used to validate the proposed solution. Section 2 is associated work on recent study events in the field of methods and applications for human action detection. Section 3 describes the basic methodologies and principles. Section 4 addresses with shallow learning the proposed method of human behavior recognition. Section 5 symbolizes the findings of the experiment. The conclusion of Section 6 is the representation of the result of the proposed scheme.

II. RELATED WORK

Anguita et al. [2] introduced a system that uses inertial smartphone sensors to recognise human physical activity (AR). Since the energy and computer power of these mobile phones is small, they suggest a new hardware-friendly method for classification of multi-class problem. This approach adapts the regular Support Vector Machine (SVM) and uses fixed-point arithmetic for the reduction of computational costs.

Tran and Phan [3] have created and built a smartphone framework for human activities through the use of integrated sensors. For acknowledgement, six acts are selected: standing, upstairs, walking, sitting, downstairs, lying down. The Support Vector Machine (SVM) for classification and identification of the operation is used in this method. For the model classification model - the model file, data obtained from sensors is analyzed. The classification models are optimized to generate the best results for the human activity described.

In the sense of human recognition of human behavior, Gusain et al. [1] evaluated gradient boosted machines (GBM). The proposal solution uses an ensemble of SVM to incorporate incremental learning. After the first batch of data has been trained, the computer is stored in many machines. This machine is trained on the new batch for the second time, and correctly classified information is removed, but the misclassified machine is trained.

The generic feature engineering approach Zdravevski et al. [4] have proposed to pick robust characteristics from a variety of sensors that can be used to generate accurate classification models. A number of time and frequency domain features have been extracted in the initially registered time series and some newly created time series [i.e. fast Fourier transformation series, first derivatives, magnitudes and Delta series]. Also, the number of functions generated is substantially reduced with a two-phase function selection. Finally, various classification models are trained and tested in a separate test collection.

Hassan et al. [5] proposed an inertial smartphone sensor method for detection of human behavior. Second, raw data extract productive functionality. The characteristics include autoregressive coefficients, meaning, median, etc. A Linear Discriminant Analysis (LDA) and kernel principal component analysis (KPCA) further process the features to make them robust. The features are eventually educated in effective

identification of behavior with the Deep Belief Network (DBN).

Xu et al. [6] proposed an InnoHAR deep learning model based on a neural network and a recurring neural network. The model enters end-to-end multi-channel sensor waveform data. Multi-dimensional functions with different kernel-based convolution layers are extracted in initial modules. In conjunction with GRU, time series characteristics are modelled and data characteristics are used entirely to complete classification tasks.

Inertial smartphone accelerometer architecture design for HAR has been developed by Wan et al. [7]. In traditional everyday activities, the smartphone gathers the sensory data sequence and extracts from the original data the high efficiencies, and then uses several three-axis accelerometers to acquire the physical behavioral data of the consumer. The data are preprocessed to extract useful feature vectors by denoising, normalizing and segmenting. A real-time method of classification of human behavior based on the neural convolution network (CNN) using a CNN for the extraction of local functions is also suggested.

Next, Table I compares between recent literatures of different researchers works on human activity recognition. The table spots the major drawbacks and dataset(s) already used for benchmarking.

TABLE I. SUMMARY OF RECENT LITERATURE REVIEWS

Author	Year	Dataset	Technique	Results	Advantages	Drawbacks
Anguita et al. [2]	2012	Human Activity Recognition	Adapted support vector machine (MC-HF-SVM)	Accuracy (89%)	Improvement in computational cost	The same accuracy as the standard SVM
Tran and Phan [3]	2016	Human Activity Recognition	Support vector machine	Accuracy (89.59%)	A full human activity recognition system has been created. The system completed included systems for data collection, retrieval, analysis, training and identification of human activities. This system can be used in many fields of practice, in particular in the field of health care.	There are also a few restrictions in the system. Some indicators have a low percentage of acceptance.
Gusain et al. [1]	2017	Human Activity Recognition	extreme Gradient Boosted Decision Trees	Accuracy (90%)	The proposed solution is more accurate than the conventional approach and requires far less time. The findings are positive and show the truthfulness of the model.	More complexity in the system and need real-time processing
Zdravevski et al. [4]	2017	Human Activity Recognition and other four datasets	6 classifiers	accuracy (95.8%)	Identified that a high degree of activity recognition can be achieved with just a smartphone and smart watch. This result is	Data imbalance issue in applications for identification of human behaviors in real life.

					important and inspiring since the two devices are now widespread and not too invasive.	
Hassan et al. [5]	2018	Human Activity Recognition	(KPCA) and (LDA) +a Deep Belief Network (DBN)	Accuracy (95.85%)	The solution proposed was compared to the conventional SVM multi-class approach where its superiority was seen. It has also demonstrated its ability to differentiate between fundamental transitional and non-transitional practices.	More complexity in the system and need real-time processing
Xu et al. [6]	2019	Opportunity activity recognition	InnoHAR	Accuracy (94.6%)	The proposed approach has consistently superior performance in three most commonly used public datasets and strong generalization performance. During the experiment, they also showed that in real time, practices on the Minnow Board Turbot Dual Core Board have more potential with its creative structure.	Data imbalance issue in applications for identification of human activities in real life.
		PAMAP2		Accuracy (93.5%)		
		Human Activity Recognition		Accuracy (94.5%)		
Wan et al. [7]	2020	Human Activity Recognition	CNN	Accuracy (92.71%)	The advantages and drawbacks of human enforcement are contrasted with five algorithms, CNN, MLP, BLSTM, SVM and LSTM. Experiments show that CNN still has considerable value in recognition of human behavior and is an outstanding classification and recognition algorithm.	Further optimization is possible in the structure of the four neural network models used in the experiment and further comparative studies can be carried out.
		Pamap2		Accuracy (91%)		

III. PRELIMINARIES METHODS

A. Shallow Learning

Machine learning is seen as a form of artificial intelligence (AI) which deliver learning-free machines with no more processes and Shallow learning [8] is regarded as machine learning. They have evolved from theory of machine learning and pattern recognition. Two key categories of learning are typically un-supervised and supervised. The training set comprises samples of input vectors and matched objective vectors for supervised learning. No labels are required for the training set in unsupervised training. The supervised target of learning is to predict an adequate output vector for each vector. Classification tasks are functions where the objective label is a discrete finite number of the group. It is difficult to describe the unsupervised learning target. The related samples of sensitive clusters within input data, known as clustering, are a primary objective.

B. Genetic Programming (GP)

Genetic programming (GP) is a technique of evolutionary computing (EC) that solves problems automatically without asking the machine how to do this [9] directly. From the most abstract level, GP is a domain-independent, systematic way to get computers to solve problems automatically, from a high-level argument. GP is a special evolutionary algorithm (EA) where computer programs are present in the population. GP thus converts populations of programs, as shown in Fig. 1, from generation to generation [10].

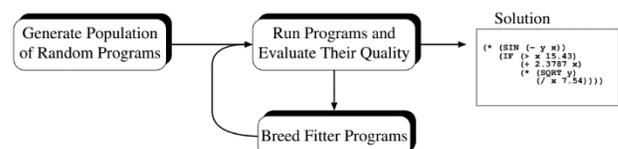


Fig. 1. Block Diagram of Genetic Programming.

Any computer application, with ordered branches, can be graphically displayed as a rooted label tree. Genetic programming is an enhancement to the conventional genetic algorithm, which is a computer programme for every person in the population. The genetic programming search area is the space for all possible computer applications, which consist of functions and terminals that are suitable for the problem area. The features may include standard arithmetic operations, logical functions, standard programming, standard math, or domain-specific functions.

Five preparatory steps have been taken [11]. The following five steps are:

- 1) The terminal package,
- 2) The elementary functions set,
- 3) The measure of fitness,
- 4) The run control parameters, and
- 5) The formula for the outcome and the end of the run criterion.

In preparation for genetic programming the first step is to classify the terminal sequence. The terminals can be seen as the entries into the computer program that has been uncovered. Terminals are the ingredients from which genetic programming tries to construct or approximately solve a computer program to solve the problem.

The second step in planning for the use of genetic programming is to recognize the set of functions to produce the mathematical expression to match the unique finite data sample. The functions of the F function set and the terminals of the T terminal set are used in any computer program. In each function set, any value and data type that may be returned to a function set and to any value and data type that may be assumed by the terminal in the set should be recognized as its arguments. That is, the selected function set and terminal set should be closed. These first two steps correspond to the step of specifying the representation scheme for the conventional genetic algorithm. These two first steps correspond to the step of defining the traditional genetic algorithm representation scheme.

The remaining genetic programming steps are the three last preparatory steps for typical genetic algorithms. Populations of hundreds, thousands and millions of computer systems are genetically derived from genetic programming. This breeding takes place using the Darwinian survival and reproduction concept of the most suitable and genetic crossover operation for computer-based programming. This combination of Darwinian natural selection and genetic operations frequently results in a computer program that solves a given problem. Genetic programming begins with an initial population of computer programs randomly generated (generation 0) consisting of features and terminals for problem domain applications.

The establishment of this initial random population is essentially a blind random search of the problem's search space as a computer program. Each computer program within the population is calculated by the fitness of the problem area. The fitness calculation is different from the problem [12]. A combination of the number of properly treated instances (i.e.,

true negatives and true positives) and the number of correct instances will calculate the fitness of a program (i.e., false positives and false negatives). Correlation is also used as a test of fitness. From the other hand, the fitness of a particular computer program may be calculated using entropy, the fulfilment of the gap test, the success test satisfaction or a combination of these. For several problems a combination of factors like correctness, parsimony (smallness in the program), or effectiveness (of execution) may be needed to use a multifunctional fitness measure [12].

In general, each computer program in the population has numerous different fitness instances, with the consequence that it is evaluated in a number of representative situations, either in total or in average. These fitness instances may be a sampling of different independent variable values or a sampling of different initial system conditions. The fitness cases can be picked alone or arranged (e.g., over a regular grid or at regular intervals). The initial conditions are also usual in cases of fitness (as in a control problem). In genetic programming generation 0, computer programs are almost always extremely suited.

Some people would however prove more suited than others in the population. These output disparities are then taken advantage of. A new descendant population of individual computer programs is being generated through the Darwinian theory of reproduction and survival of the fittest and genetic fusion process. The reproductive method involves choosing a computer program, which can be used by copying into new population [13], from the existing population of fitness-based programs. The crossover operation is used to construct new descending computer programs from two fitness-based parental programs.

The genetic programming parental systems are of various sizes and types. The offspring programs consist of their parents' sub-expressions (building block, sub-programs, subtrees, subroutines). These descent programs, as opposed to their parents, have different sizes and styles. In genetic programming the mutation operation can also be used. The population of offspring (i.e., the new generation) replaces the old population after the genetic operations on the present population (i.e., the old generation). Each participant in the new program population will then be assessed for fitness and over several generations the process is replicated. At every point, the state of the process will consist only of the present population of people in this highly parallel, locally regulated, decentralized process. The driving force behind this mechanism is just the human health in the existing population that has been observed. As can be seen from this algorithm, populations of programs are generated that appear to display an increasing average fitness in their environment over several generations. Furthermore, these machine populations are able to adapt quickly and efficiently to environmental changes. The best person in any run is usually referred to as the outcome of the course of genetic programming [13]. Inherently hierarchical are the products of genetic programming.

Sometimes, genetic programming effects are default hierarchies, priority hierarchies of tasks, or hierarchies where one action subordinates or suppresses another. Another

fundamental characteristic of genetic programming is the dynamic variability of the computer programs that are built along the way to a solution [10]. The effort to describe or minimize in advance the dimensions and form of the potential solution is always hard and unnatural. Furthermore, advancing or restricting the solution's dimensions and type narrows the window through which the machine sees the world and may prevent the solution of the problem from ever being found. The absence or a relatively minor function of inputs and post-processing inputs is another important aspect of genetic programming. Usually, the inputs, intermediate effects and outputs are directed in the natural terminology of the problem region. Genetic programming systems consist of functions that are natural to the problem area. If appropriate, a wrapper will conduct the post processing of the output of a program (output interface). Eventually, another key element of genetic programming is the active genetic programming structures [11]. They really aren't inactive encodings of the problem (i.e. chromosomes). Instead of running a machine, the genetic program structures are active structures that can be run as they are.

C. Decision Tree

The classification in decision trees [14] is based on a sequence of decision classification of the sample. The present decision helps to make a subsequent decision in a decision tree to create a sequence that is indicative of the structure of the tree. The structure includes two key types of attributes and allows to use attributes during the prediction process. The predicted attribute is described as a dependent variable because the value of the other attributes depends on or depends upon the values. The other attributes that help to forecast the dependent variable value are known as the independent dataset variables. In the case of classification, each end leaf node represents one decision or category, the root node becomes an eligible end leaf node, for instance. Each node has the attributes of the instances and the value of each division is the same as the attributes of each division. The decision tree is a model that determines the value of the dependent variable(s) based on the values of various attributes of the data available in a new case. In decision tree, The inner nodes indicate the various attributes The divisions between the nodes reflect potential values in the observed samples for these attributes, whereas the classification of the dependent variable or the final value are represented by the terminal nodes.

After the related basic calculation, the J48 [15] decision tree classifier shall be used. In order to order anything else, the ultimate objective is to create a selection tree first because of the quality estimates of the accessible preparation information. Therefore, whatever the stage in which things are organized (training), it identifies the characteristic that usually clearly distinguishes the various occurrences. This distinctness, which has the capacity to get the best out of the instances, may be structured to collect the necessary data. At present, if the standard for which the information events falling into its class have the same meaning for the target variable is of some fair value that there is no vagueness, this expansion would be terminated and designated as the objective value that could be achieved. For alternative situations, the search for alternate quality begins which results in the most astonishing data

collected—and goes on until either a fair choice has been made about which combination of the unique characteristics of a particular target quality or the use of properties. If qualities are used or if an exact result from the accessible information cannot be obtained, the degradation of this extension goal was priced for most of the items in this branch. [5]. See Table II which compares between different algorithms can be used for building decision tree.

D. Naïve Bayes Classifier

One of the most popular simple machine learning classifiers is a probabilistic classifier. Due to the use of probability distribution over a set of classes, the classifier can predict a sample instance instead of predicting only one class for the sample. Probabilistic classifiers have a certain description that can be useful when classifiers are combined into ensembles. Naïve Bayes is known by its probabilistic designation as the straightforward street algorithm. Naïve Bayes is a statistical classification that calculates the likelihood of a certain class of tuple based on the Bayes theorem [16]. The class-conditional Independence characterizes Naïve Bays, implying that the influence of an attribute-value on a certain class is irrespective of the other attributes. High accuracy, pace and many advantages are of Naïve Bayes. In principle, in contrast with all other classifiers, minimum error rate is the main characteristic of Bayesian classifiers [17].

Naïve Bayes is working on a basic definition, but a very intuitive one. In certain instances, Naïve Bayes beats several comparatively complex algorithms by using the variables in the data sample and by observing each other separately and independently. The classification of Naïve Bayes is based on the conditional probability rule of Bayes. It begins with all of the attributes in the data that are equally relevant, independent from one another and is evaluated individually. It works with the hypothesis that one feature works without the others in the study. The model offers a response to questions such as "What is the likelihood of a certain type of attack, provided certain device events, when it comes to using Naïve Bayes, NB in model intrusion attacks? The query in turn is reworded in the context of conditional probability. A directed acyclic (DAG) charts the structure of an NB. Each node represents one of the system variables and each relation codes the effect of one node on the next. So A directly influences B when it has a relation from node A to node B [6].

TABLE II. COMPARISON BETWEEN DIFFERENT ALGORITHMS FOR BUILDING DECISION TREES

Characteristics	CART	C4.5	ID3
Pruning	Post pruning	Pre-pruning	Not applicable
Improvement	Accepted	Not accepted	Not accepted
The lack of values	Can handle the issue	Cannot handle the issue	Cannot handle the issue
Data type	Nominal & Continuous	Categorical & Continuous	Categorical
Formulation	Gini index is used	Gain ratio is used	Information gain & entropy are used
Rapidity	On-average	More quickly than ID3	Low speed

E. Proposed Human Action Recognition System (HARS)

1) *Problem formulation:* Recognition of human activity (HAR) is a method of pattern recognition. Since preprocessing, feature engineering and assignment to labels are the key recognition processes, HAR has applied the method to all sub processes mentioned. In order to identify an action as a label, it is important to preprocess the input data and evaluate it in order to detect whether there is an abnormal value...etc. Due to the acquiesced data from sensors of the smartphone, which represents the characteristics and features of human activity. The final step before classification, however, includes the study and engineering of features. We addressed each method for more details in depth in the following sections.

2) *Shallow human action recognition system:* The framework proposed is planned and built in two phases: the first phase consists of the pre-processing of acquired data values, feature engineering and analysis. In contrast, the second stage is the process of using shallow learning algorithms and applies classification based on a Decision tree, Naïve Bayes, and Genetic Programming. This paper proposes

a comparatively shallow learning algorithm for human action recognition based on smartphones. We have taken full advantage of their strengths. Fig. 2 demonstrates our proposed recognition model's abstract architecture. The stream data is pre-processed in instances and is inserted into the classifier in each instance. The shallow learner infers the corresponding action of the input tuple. We have adopted a decision tree architecture to automatically learn efficient and robust action features from the training instances.

Stage (1): Pre-processing

Data is first prepared in the attributes at the last index of the data set class name. In order to minimize the difficulty of the performance assessment of the proposed model, the module data of the class is divided into two key categories. The dataset had already been prepared and made available online. For easy study, activity labels are then changed by having the class label 1, 2, 3, 4, 5 or 6 for WALKING, WALKING_UPSTAIRS, WALKING_DOWNSTAIRS, SITTING, STANDING or LAYING, respectively. The datasets were transferred with the best search evaluator to the feature analysis with the selection of correlation features to analyze the correlated and uncorrelated features during the processing.

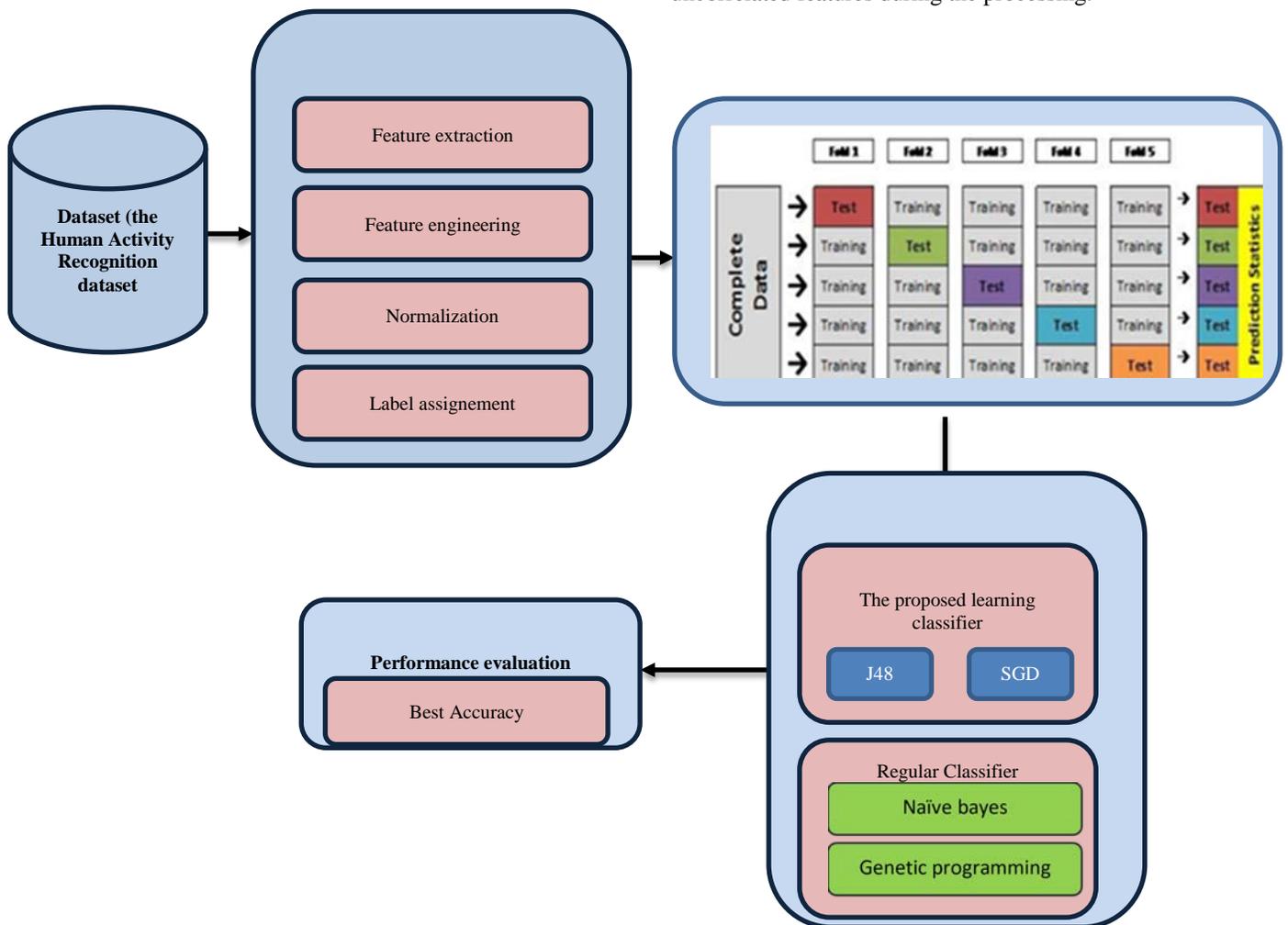


Fig. 2. The Block Diagram for the Proposed Model.

Stage (2): Modeling using Shallow Learning Algorithm

1) *Dataset splitter*: The process of loading data set into two components obtained from the preprocessing module is the process which divides the dataset. The technique of cross-fold validation [18] divides the date set into 2 parts when data is partitioned randomly into k-fold independent set, group is a training set and fold is a test set. The training set is used to train the proposed system while checking and validating the accuracy of the trained model is carried out.

2) *Learning phase*: In the proposed system, the learning phase starts the learning process of the classifier using the current instance from the training dataset that was realized. The results of this base classifier are considered to be the input data for the second stage. The update to the classifier has its own rule set for every detail to be independently conscious of the behavior of human activity.

Stochastic descent [19] (often SGD) is an iterative way to refine an objective function with sufficient smoothness properties (e.g. sub differentiable or differentiable). The gradient optimization can be seen as a stochastic approx. because the real gradient (calculated from the whole dataset) is replaced by an approximation (calculated from a randomly selected subset of the data). This reduces the machine burden, achieving faster iterations in trade at a lower convergence rate, particularly in high-dimensional optimization problems. This is the stochastic gradient descent used to optimize the parameters of j48.

Stage (3): Action recognition stage

The proposed framework uses the test dataset developed by the Data Splitting module during this stage. The test data set is used to assess the model's output. The results of this module are redirected to create the complete classifier performance assessment process that is discussed in detail in the experimental section using the classification rules created during the learning phase for the assessment of the proposed model.

IV. EXPERIMENTAL RESULTS

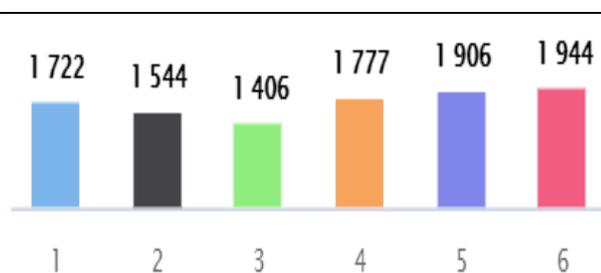
A. Runtime Environment

On a computer system with features the proposed recognition system has been implemented and designed using JAVA 8. Such features have been listed as hardware, comprising 64-bit machine and Intel of Core i7 and 2.2 GHz rather than as a software framework for Windows 10 Professional.

B. Data Set(s)

To evaluate the proposed system, we have used a dataset of Human Activity Recognition provided by Jorge L et al.[2]. It is available online for public use on the UCI repository since 2013. It contains six types of actions, including, LAYING, SITTING, WALKING, STANDING, WALKING_DOWNSTAIRS and WALKING_UPSTAIRS. All has a smartphone on the waist (Samsung Galaxy S II). This dataset version contains all the training and testing examples provided in the original data repository. The data collection was randomly divided into two groups, with 70% of the volunteers chosen to produce the training data and 30% of the test data. The database is composed of the logs of 30 people who conduct everyday life activities (ADL) with an embedded inertial sensor waist-mounted smartphone. In an age group of 19-48 years, the tests were performed with 30 volunteers. We captured three axial linear acceleration and three axial angular speeds at constant speeds of 50Hz using its embedded accelerometer and gyroscope. The tests were captured by video to manually mark the data. The sensor signals (gyroscopes and accelerometers) were pre-processed by using noise filters and sampled in 50 percent overlap and 2.56 sec and (128 readings/windows) in fixed-wide sliding windows. A Butterworth low-pass filter into body acceleration and gravity separated the sensor acceleration signal with gravitational and body movement elements. It is believed that the gravitational table force is only low frequency; a filter with a cutoff frequency of 0,3 Hz has thus been utilized. The measurement of time and frequency domain variables obtained a vector of features from each window. Table III represents a summary of these dataset.

TABLE III. SUMMARY OF THE DATASET

Item	Description
Number of Instances	10299 
Number of features	562
Number of Labels	6
Has missing values	No

V. PERFORMANCE MEASURE

In order to test the accuracy of the classification, the Classifier Performance Evaluator [20] applies various classification performance tests. These measurements are based on certain concepts such as TN (True Negative), FN (False Negative), TP (True Positive) and FP (False Positive).

In Table IV, a table visualization of algorithmic output is represented by the confusion matrix.

TABLE IV. PERFORMANCE MEASURE

Metric	Formula	Metric	Formula
accuracy	$\frac{TP + TN}{P + N}$	Recall, sensitivity	$\frac{TP}{P}$
Precision	$\frac{TP}{TP + FP}$	error rate	$\frac{FP + FN}{P + N}$
Specificity	$\frac{TN}{N}$		

Precision is the most sensitive and interesting measure to be used to compare the fundamental classifiers and the proposed system in a detailed range.

VI. EXPERIMENTAL RESULTS AND DISCUSSION

The results of the proposed model are provided in this section. The results provided a comparison between the proposed as an application of three main classifiers. According to Table V, the J48 achieved better accuracy than the naïve Bayes and the genetic programming. These results displayed graphically in Fig. 3. The J48 take the advantages of no domain knowledge required, no parameter setting, can handle multidimensional data, simple and fast.

TABLE V. ACCURACY EVALUATION OF THE PROPOSED CLASSIFIERS

Classifier Algorithm	Accuracy
Decision Tree (J48)	96.6387
Naïve Bayes	94.958
Genetic Programming	94.958

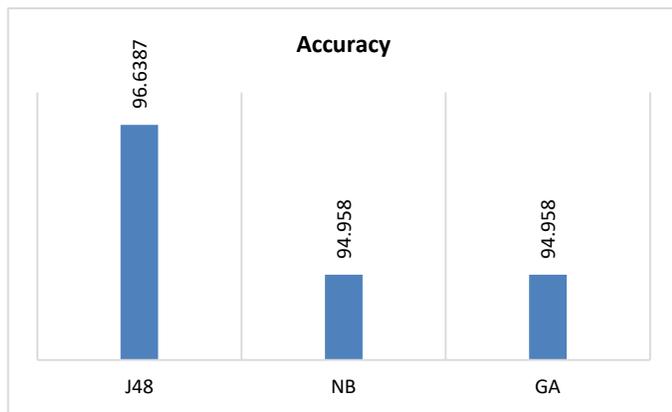


Fig. 3. The Proposed Model Results.

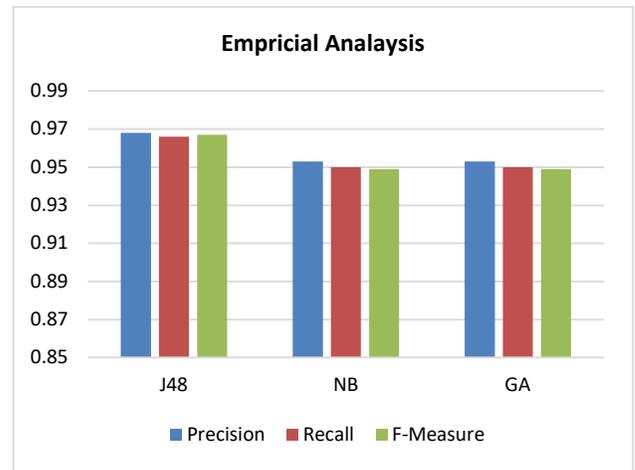


Fig. 4. Empirical Analysis of the Proposed Optimized Algorithm J48 with SGD versus Naïve Bayes and Genetic Classifiers.

Fig. 4 compares the major enhancements of the proposed optimized classifier J48 with SGD versus other regular classifiers including Naïve Bayes and genetic classifiers. The proposed classifier achieves the highest values in precision, recall and F-score measure. While other classifiers almost have the same level of achievements regards the different metrics.

TABLE VI. A COMPARISON TO THE LITERATURE

Authors	Algorithm	Accuracy
Anguita et al. [2]	Adaptive SVM	89
Tran and Phan [3]	SVM	89.5
Gusain et al. [1]	Extreme Gradient DT	90
Hassan et al. [5]	DBN	95.85
Xu et al. [6]	InnoHAR	94.5
Wan et al. [7]	CNN	92.7
The Optimized Proposed Classifier	Hybrid J48-SGD	96.6387

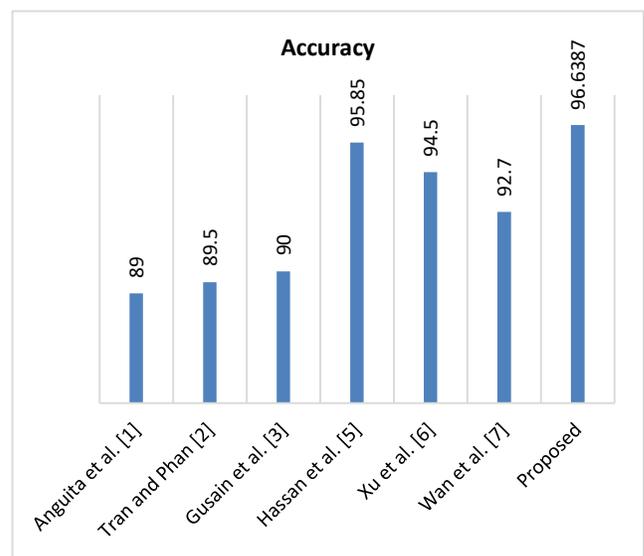


Fig. 5. A Comparison to the Literature Regards to Accuracy Metric.

Moreover, the proposed model compared to the literature according to the accuracy parameter. The results accepted the supremacy of the proposed model, as seen in Table VI and Fig. 5, in achieving the best recognition rate rather than the sixth compared models.

VII. CONCLUSION

If sensors help patients register and monitor them all time and automatically report because of any abnormal activity is found, a massive number of resources may be saved. This paper presented a model to detect how effectively the user conducts physical activity using the sensors of the telephone. The model utilized three different classifiers to test the percentage of the recognition. The j48 with stochastic gradient descent approved its superiority rather than naïve Bayes and genetic programming. Moreover, the model compared to the literature and success to achieve the highest accuracy reached to 96.6%. In future work, the proposed model will be integrated to different optimization algorithm like the slap swarm optimization algorithm, the crow search algorithm or the grey wolf optimization algorithm to improve the recognition rate.

REFERENCES

- [1] K. Gusain, A. Gupta, and B. Popli, "Transition-aware human activity recognition using extreme gradient boosted decision trees," in *Advanced Computing and Communication Technologies*, Springer, 2018, pp. 41–49.
- [2] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine," in *International workshop on ambient assisted living*, 2012, pp. 216–223.
- [3] D. N. Tran and D. D. Phan, "Human activities recognition in android smartphone using support vector machine," in *2016 7th international conference on intelligent systems, modelling and simulation (isms)*, 2016, pp. 64–68.
- [4] E. Zdravevski et al., "Improving activity recognition accuracy in ambient-assisted living systems by automated feature engineering," *Ieee Access*, vol. 5, pp. 5262–5280, 2017.
- [5] M. M. Hassan, M. Z. Uddin, A. Mohamed, and A. Almogren, "A robust human activity recognition system using smartphone sensors and deep learning," *Futur. Gener. Comput. Syst.*, vol. 81, pp. 307–313, 2018.
- [6] C. Xu, D. Chai, J. He, X. Zhang, and S. Duan, "InnoHAR: a deep neural network for complex human activity recognition," *Ieee Access*, vol. 7, pp. 9893–9902, 2019.
- [7] S. Wan, L. Qi, X. Xu, C. Tong, and Z. Gu, "Deep learning models for real-time human activity recognition with smartphones," *Mob. Networks Appl.*, vol. 25, no. 2, pp. 743–755, 2020.
- [8] Q. Tian, J. Li, and H. Liu, "A method for guaranteeing wireless communication based on a combination of deep and shallow learning," *IEEE Access*, vol. 7, pp. 38688–38695, 2019.
- [9] S. Nguyen, Y. Mei, and M. Zhang, "Genetic programming for production scheduling: a survey with a unified framework," *Complex Intell. Syst.*, vol. 3, no. 1, pp. 41–66, Mar. 2017.
- [10] M. Suganuma, S. Shirakawa, and T. Nagao, "A genetic programming approach to designing convolutional neural network architectures," in *Proceedings of the Genetic and Evolutionary Computation Conference*, 2017, pp. 497–504.
- [11] A. H. Gandomi, A. H. Alavi, and C. Ryan, Eds., *Handbook of Genetic Programming Applications*. Cham: Springer International Publishing, 2015.
- [12] K. Nag, T. Pal, and N. R. Pal, "ASMIGA: An Archive-Based Steady-State Micro Genetic Algorithm," *IEEE Trans. Cybern.*, vol. 45, no. 1, pp. 40–52, Jan. 2015.
- [13] B. Tran, B. Xue, and M. Zhang, "Genetic programming for feature construction and selection in classification on high-dimensional data," *Memetic Comput.*, vol. 8, no. 1, pp. 3–15, Mar. 2016.
- [14] M. Li, H. Xu, and Y. Deng, "Evidential decision tree based on belief entropy," *Entropy*, vol. 21, no. 9, p. 897, 2019.
- [15] D. Bienvenido-Huertas, J. E. Nieto-Julián, J. J. Moyano, J. M. Macías-Bernal, and J. Castro, "Implementing artificial intelligence in H-BIM using the J48 algorithm to manage historic buildings," *Int. J. Archit. Herit.*, pp. 1–13, 2019.
- [16] I. M. El-hasnony and O. H. Al-tarawneh, "A Proposed Hybrid Effective Technique for Enhancing Classification Accuracy," no. December, 2017.
- [17] Y.-C. Zhang and L. Sakhanenko, "The naive Bayes classifier for functional data," *Stat. Probab. Lett.*, vol. 152, pp. 137–146, 2019.
- [18] I. M. El-Hasnony, S. Baracat, M. Elhoseny, and R. R. Mostafa, "Improved Feature Selection Model for Big Data Analytics," *IEEE Access*, 2020.
- [19] R. Sweke et al., "Stochastic gradient descent for hybrid quantum-classical optimization," *Quantum*, vol. 4, p. 314, 2020.
- [20] S. Alemam, S. Abuelsadat, S. Saber, and T. Elsewify, "Accuracy, sensitivity and specificity of three imaging modalities in detection of separated intracanal instruments," *G. Ital. Endod.*, vol. 34, no. 1, 2020.