Application of smartphone in recognition of human activities with machine learning

Sabah Mohammed Fayadh[1], Elham Mohammed Thabit A. Alsaadi[2], Huda Hallawi[2]
[1]Department of computer Systems, Al Nassriyah Technical Institute, Southern Technical University, Thi-Qar, Iraq
[2]Department of Information Technology, College of Computer Science and Information Technology, University of Karbala, Karbala, Iraq

Article Info

Article history:

Received Jul 21, 2022
Revised Dec 21, 2022
Accepted Dec 26, 2022

Keywords:

Euclidean distance
Human activity recognition
K-nearest neighbour algorithm
Machine learning
Quantitative

ABSTRACT

The aim of activity recognition is to determine the physical action being performed by one or more users based on a series of observations made during the user's actions in the relevant environment. Significant advancements in the field of human activity have resulted in the creation of novel ways for supporting elderly persons in doing their tasks independently. Using ambient computing, this type of service will be manageable. Many of services are provided by ambient technology, involving home automation tools, monitoring the behaviour of diseased individuals, and utility management. Numerous academics are focusing their efforts on computer software architectures, system infrastructure, and distributed applications utilising sensor devices. Aim of this project is to develop an algorithm that can perform human activity recognition (HAR) better than the existing state-of-the-art approach. Several tasks must be done to achieve this goal. To compete with an existing HAR system, this study will rely on secondary data from the cutting-edge experiment; no new data will be collected. The central experiment will be used to quantitatively identify the best classifier based on prediction accuracy. The current study entails monitoring and assessing existing literature in order to generate hypotheses that may be tested via experiment.

Corresponding Author:

Elham Mohammed Thabit A. Alsaadi
Department of Information Technology, College of Computer Science and Information Technology
University of Karbala
Karbala, Iraq
Email: elham.thabit@uokerbala.edu.iq

1. INTRODUCTION

Human activity recognition (HAR) has emerged as a key topic of research in recent years [1]. It is a crucial and demanding field that has the potential to yield numerous unique universal applications [2]. Surveillance, interactive gaming interfaces, smart homes, and just-in-time information systems for office workers are all built on a system that recognises human behaviour [3], [4]. This is a multidisciplinary study topic that is inextricably linked to artificial intelligence (AI), ubiquitous learning, human-computer interaction, physiology, sociology, and machine learning and perception. This is why several researchers from a variety of professions are interested in it [5], [6]. The human activity recognition system recognises and analyses the users' activities or actions by passively studying their behavioural patterns and their environment in order to take appropriate responses [7], [8]. By recognising activities in real time, the activity recognition system contributes to the development of just-in-time learning environments that teach individuals by giving information at the appropriate time for their surroundings [9]. By analysing the user's activity, these systems can determine the optimal time to interrupt the user and offer valuable information or messages.

While HAR looks to be a simple process, selecting the relevant tools and techniques for data collection, storage, and manipulation presents various difficulties and challenges [10], [11]. A prediction method is essential because it must be able to capture both inter-class variation and intra-class similarity [12], [13]. The difficulty of handling normal resource restrictions such as time availability, processing power, and adequate storage capacity is a significant impediment. The latency, precision, and computing power of the system must all be balanced.

The early stages of HAR focus on a small range of concerns. Choosing the suitable sensors, determining the quality and metrics to be monitored, and correctly positioning the sensor are all critical. Everything has to be done while considering the privacy and usefulness of the end user [14], [15]. User participation and interaction could have a significant impact on how the entire process works. Thus, the fundamental goal of this research is to develop an algorithm capable of performing HAR with a classification accuracy greater than that of the existing state-of-the-art technique [16], [17]. A number of tasks must be completed in order to reach this goal. As this study attempts to compete with an existing HAR system, it will primarily conduct secondary research using data from the state-of-the-art experiment; no more data will be generated for the scope of this project. The central experiment will be conducted with the quantitative purpose of selecting the best classifier as assessed by predicting classification accuracy. The current study entails monitoring and assessing existing literature in order to generate hypotheses that may be tested via experiment.

2. LITERATURE REVIEW

Over the recent decade, HAR has grown in popularity due to its multiple applications in human-centric sectors such as security, military, and medical. Because of the proliferation of wearable technology gadgets, the task of recognizing human activity has taken on increased significance in recent years. Piergiovanni *et al.* [18] provided another motion representation by proposing a convolutional layer technique inspired by optical flow to describe motion present in the video sequence for activity recognition.

To enhance the performance of two-stream networks, Tu *et al*. [19] proposed the human-related multi-stream CNN (HR-MSCNN) architecture that integrates appearance, human motion, and human-related regions. Despite awareness of human-to-human activity Xiong *et al.* [20] proposed activity recognition between human and robot in human robotic collaboration (HRC) environment due to two major challenges: i) lack of a large volume of human-robot activity data for deep network training and optimization; and ii) work operational constraints in the manufacturing industry. Consequently, they merged both problems and presented optical flow and CNN-based transfer learning.

Abdelbaky *et al*. [21] introduced a basic principal component analysis network based on unsupervised learning (PCNet). A short-time motion energy image (ST-MEI) template has been acquired for activity recognition in videos utilizing PCNet. For robust activity recognition, the suggested PCNet architecture learns hierarchical motion patterns using ST-MEI.

Similarly, Leong *et al.* [22] introduced a semi CNN architecture for activity recognition that combines 1D, 2D, and 3D convolutions to lower the trainable parameter of 3D convolution. The suggested architecture employs transfers learning for the acquisition of spatial feature knowledge, followed by temporal encoding and 3D convolutions. Khan *et al*. [23] also suggested a fusion technique for activity recognition based on deep neural network (DNN) and multi-view characteristics. The DNN features are retrieved using a pretrained network, specifically VGG-16, and the multi-view features are extracted using horizontal and vertical gradients in addition to vertical direction characteristics.

To eliminate the interclass parallelism between human activities Martin *et al.* [24] introduced an original twin spatio-temporal convolutional neural network (TSTCNN). TSTCNN is a two-stream network with three Spatio-temporal CNN layers in each stream. The proposed architecture's inputs consist of an red, green, and blue (RGB) frame sequence stream and an optical flow stream. Meng *et al.* [25] presented an adaptive temporal fusion network dubbed AdaFuse for robust temporal modeling. The adaFuse network combines channels from previous feature maps with current trimmed feature maps to generate a robust collection of temporal characteristics and enhance the accuracy of activity recognition.

3. THE PROPOSED METHOD

Predictive models are generated using machine learning (ML) algorithms, and insights can be gleaned from the prepared data. In order to acquire the best fit between the model and the data, each ML model needs to be fine-tuned for a number of parameters and variables. In the literature review, a number of classification algorithms that are often used in ML and the HAR area were identified. The information-based ML family will be used to model decision tree (DT). Each parameter can be changed to develop a classification model that is more accurate when modelling decision trees. Figure 1 depicts flow chart of the work.

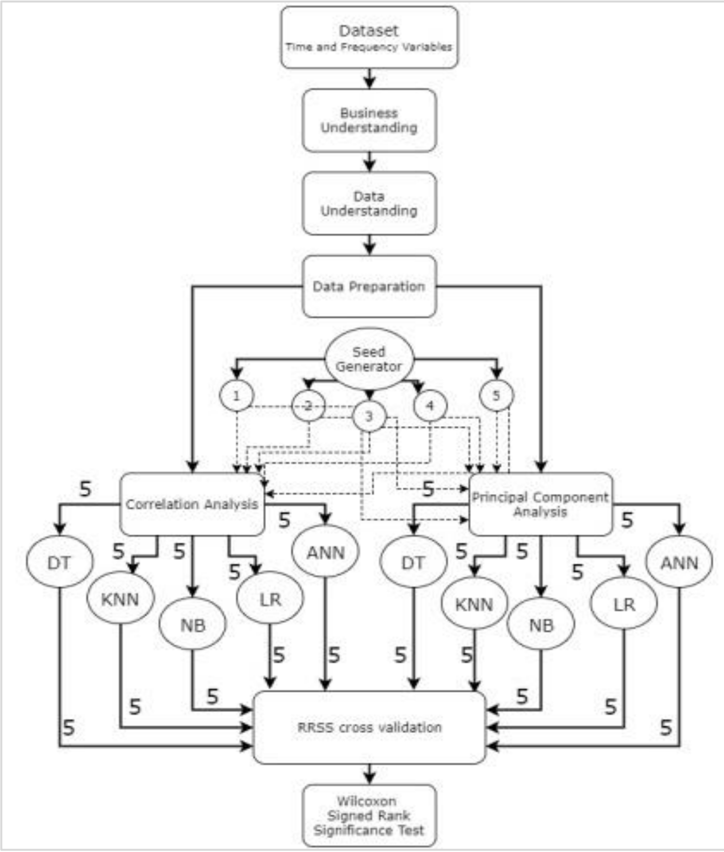


Figure 1. Flow chart of the work

The target and the independent variables selected will serve as the first parameters for the model. Splitting criteria, such as the information gain value, are also important. To help the K-nearest neighbour (KNN) method distinguish, target feature and the independent variables must be provided. The 'K' parameter indicates the number of neighbors to evaluate when using the KNN algorithm. A 'tune length' must be set to several values of 'K' in order to select the best one for the experiment. In KNN, the euclidean distance (ED) is used as the default distance. KNN in terms of ED are used to determine the target's final value.

Five alternative approaches are used for this method. To get started, gather five different sets of training and testing datasets. Each of the mentioned ML techniques must be used to model one of the five training sets. Confusion matrices are a useful tool for model evaluation since they summarise the model's recall, precision, and overall classification accuracy in an easy-to-understand manner. A distribution is generated by combining the “five cross-validation results” and the “classification accuracy of each induced ML model”.

4. RESULTS AND DISCUSSION

The dataset for this study was derived from the 2012 International Workshop on Ambient Assisted Living (IWAAL) conducted in Spain. The experiment has been done throgh a group of 30 participants performing a series of six physical exercises. The tri-axial linear acceleration and tri-axial angular velocity of a smart phone device are collected using the built-in accelerometer and gyroscope. To decrease noise from these sensor inputs, they were processed and sampled every 2.66 seconds with a fixed width sliding window that overlapped 50.21% of the time. Figure 2 displays a sample collection of feature names.

```
(names(fulldata))
 [1] "tBodyAcc-mean()-X"            "tBodyAcc-mean()-Y"
 [3] "tBodyAcc-mean()-Z"            "tBodyAcc-std()-X"
 [5] "tBodyAcc-std()-Y"             "tBodyAcc-std()-Z"
 [7] "tBodyAcc-mad()-X"             "tBodyAcc-mad()-Y"
 [9] "tBodyAcc-mad()-Z"             "tBodyAcc-max()-X"
[11] "tBodyAcc-max()-Y"             "tBodyAcc-max()-Z"
[13] "tBodyAcc-min()-X"             "tBodyAcc-min()-Y"
[15] "tBodyAcc-min()-Z"             "tBodyAcc-sma()"
[17] "tBodyAcc-energy()-X"          "tBodyAcc-energy()-Y"
[19] "tBodyAcc-energy()-Z"          "tBodyAcc-iqr()-X"
[21] "tBodyAcc-iqr()-Y"             "tBodyAcc-iqr()-Z"
[23] "tBodyAcc-entropy()-X"         "tBodyAcc-entropy()-Y"
[25] "tBodyAcc-entropy()-Z"         "tBodyAcc-arCoeff()-X,1"
[27] "tBodyAcc-arCoeff()-X,2"       "tBodyAcc-arCoeff()-X,3"
```

Figure 2. A sample collection of feature names

Typically, data is split 70:30 across training and test datasets. There were a total of 563 features, such as the subject and activity variable, and 10,299 rows, which represented the combined test and train sets. Produces a matrix of size 561 by 561 that contains the correlation coefficients for all features. The dataset's class, rows, and names of the columns are examined.

Figure 3 displays the target variable's histogram “Activity”, and the target variable's histogram “Subject” that serves as an identifier respectively, which were used to analyse the frequency distributions. Figure 3(a) the target variable's histogram “Activity”, Figure 3(b) the target variable's histogram “Subject”. Class noise and data inconsistencies were not found in either field.

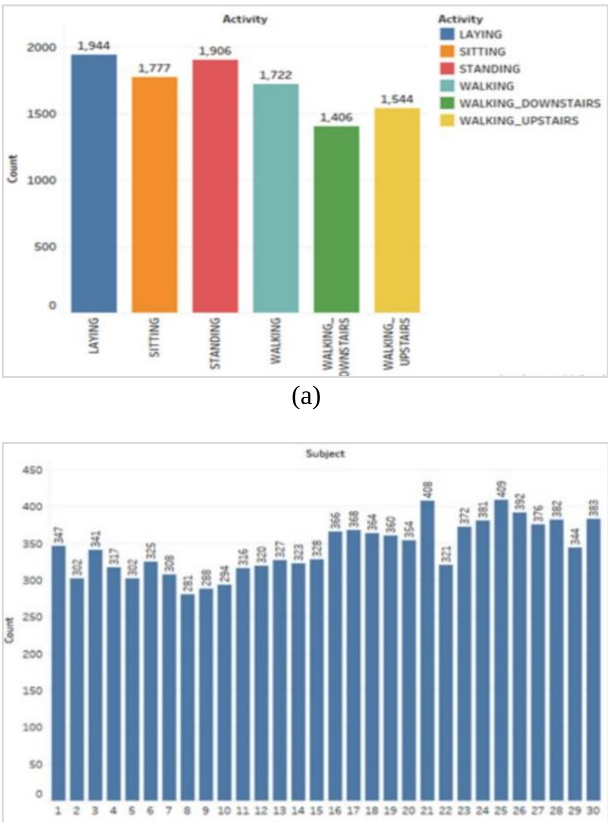


Figure 3. The target variable's histogram; (a) activity and (b) subject

Figure 4 displays a correlation coefficient with a low value is an illustration of a characteristic with a low correlation coefficient. While Figure 5 offers a characteristic with a high positive correlation coefficient value is an example. And Figure 6 offers a characteristic with a high negative correlation coefficient value is an example.

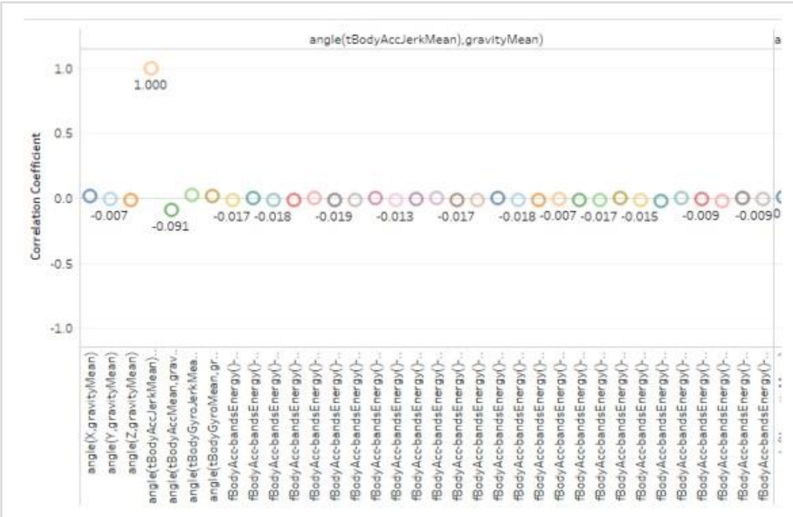


Figure 4. A correlation coefficient with a low value

Figure 5. A characteristic with a high positive correlation coefficient value

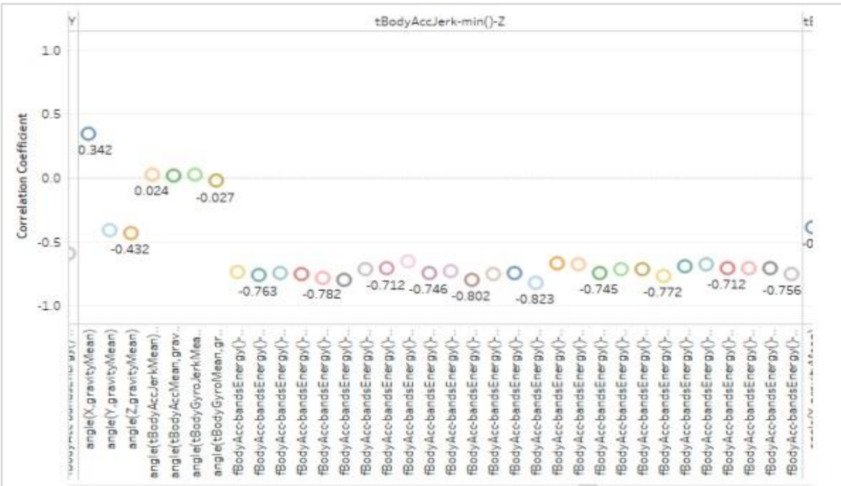
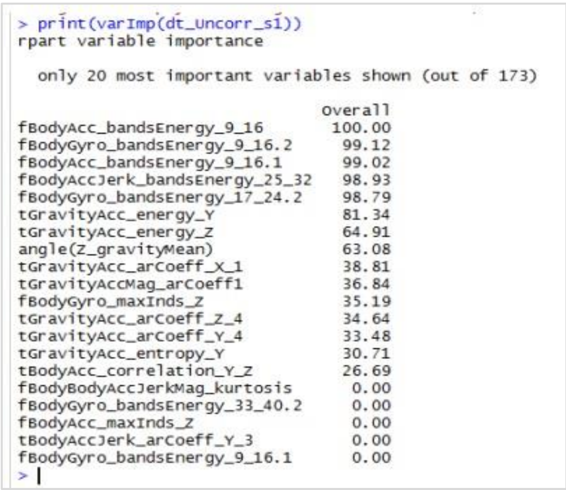


Figure 6. A characteristic with a high negative correlation coefficient value is an example

When using the caret package, the 'findCorrelation' function accepts a pairwise absolute correlation value of 0.8 as an input, denoting a strong relationship between the features (Taylor, 1990). 389 feature names had generated by utilizing this way, all of which are highly associated with one another and have an overall greater average value of mean absolute correlation than, their individual counterparts. In the last step, an uncorrelated dataset is formed through excluding all features from a previously prepared highly correlated list. As a result, we had a dataset with 174 features that weren't linked together. This data has been stripped of the Subject feature, which serves as an identification. 173 columns in the final uncorrelated dataset are examined for dimensions. Figure 7 shows a summary of the most noteworthy findings.

An optimal tuning length for K-NN algorithms has been set at '7', which appears to be a good balance between accuracy and model execution time. Cross validation was employed to iterate through K iterations in the training phase. It takes a long time to cross-validate a large number of K values. Summary of the KNN model in Figure 8 displays that a K value of 5 is the most suitable for the data. The default settings of the Naive Bayes algorithm are employed, and no kernel functions are applied. In terms of implementation, it was the quickest of the models. Each feature engineering technique generates a set of five accuracy values for each model. The classification accuracy distribution results in the selection of the optimal algorithm. For each seed value, the algorithm's results using both techniques are shown. Tables 1 to 4 show the accuracy for four different algorithms.

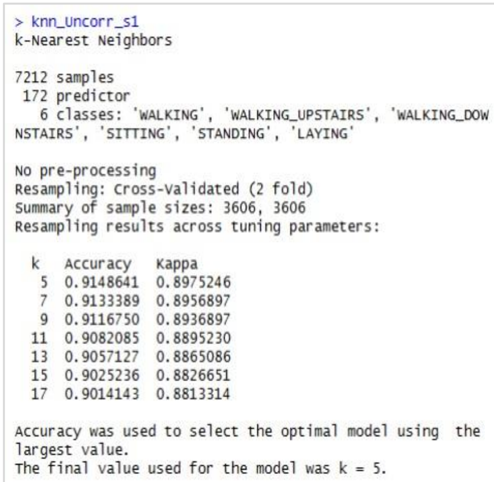


```
> print(varImp(dt_Uncorr_s1))
rpart variable importance

  only 20 most important variables shown (out of 173)

                              Overall
fBodyAcc_bandsEnergy_9_16      100.00
fBodyGyro_bandsEnergy_9_16.2    99.12
fBodyAcc_bandsEnergy_9_16.1     99.02
fBodyAccJerk_bandsEnergy_25_32  98.93
fBodyGyro_bandsEnergy_17_24.2   98.79
tGravityAcc_energy_Y            81.34
tGravityAcc_energy_Z            64.91
angle(Z_gravityMean)            63.08
tGravityAcc_arCoeff_X_1         38.81
tGravityAccMag_arCoeff1         36.84
fBodyGyro_maxInds_Z             35.19
tGravityAcc_arCoeff_Z_4         34.64
tGravityAcc_arCoeff_Y_4         33.48
tGravityAcc_entropy_Y           30.71
tBodyAcc_correlation_Y_Z        26.69
fBodyBodyAccJerkMag_kurtosis     0.00
fBodyGyro_bandsEnergy_33_40.2    0.00
fBodyAcc_maxInds_Z               0.00
tBodyAccJerk_arCoeff_Y_3         0.00
fBodyGyro_bandsEnergy_9_16.1     0.00
> |
```

Figure 7. The significance of a variable as determined by a decision tree



```
> knn_Uncorr_s1
k-Nearest Neighbors

7212 samples
 172 predictor
   6 classes: 'WALKING', 'WALKING_UPSTAIRS', 'WALKING_DOW
NSTAIRS', 'SITTING', 'STANDING', 'LAYING'

No pre-processing
Resampling: Cross-Validated (2 fold)
Summary of sample sizes: 3606, 3606
Resampling results across tuning parameters:

  k   Accuracy   Kappa
   5  0.9148641  0.8975246
   7  0.9133389  0.8956897
   9  0.9116750  0.8936897
  11  0.9082085  0.8895230
  13  0.9057127  0.8865086
  15  0.9025236  0.8826651
  17  0.9014143  0.8813314

Accuracy was used to select the optimal model using  the
largest value.
The final value used for the model was k = 5.
```

Figure 8. KNN model summary

Table 1. Accuracy for DT algorithm

Seed	Decision Tree (DT) Dimensionality Reduction (DR) Correlation Analysis (CA)	Principal component analysis (PCA)
1	0.62	0.52
2	0.61	0.51
3	0. 61	0.51
4	0.51	0.60
5	0.61	0.61

Table 2. Accuracy of Naive Bayes algorithm

	Naive Bayes (NB) DR	
Seed	CA	PCA
1	0.85	0.80
2	0.82	0.82
3	0. 84	0.81
4	0.85	0.82
5	0.83	0.82

Table 3. Accuracy for KNN algorithm

Seed	K Nearest Neighbor (KNN) Dimensionality Reduction (DR) Correlation Analysis (DR)	Principal Component Analysis (PCA)
1	0.93	0.90
2	0.94	0.90
3	0. 93	0.90
4	0.928	0.91
5	0.93	0.90

Table 4. Accuracy for ANN

	Artificial Neural Network (ANN) DR	
Seed	CA	PCA
1	0.97	0.96
2	0.97	0.96
3	0. 97	0.97
4	0.97	0.971
5	0.97	0.97

As shown in Figure 9, the target features are evenly distributed throughout the dataset, as well as in each of the test and train data samples that were generated. As you can see in the Table 5. the DT with the best accuracy value is shown. In terms of distinguishing between activities, the decision tree has been unable to do its job. Standing is frequently mislabeled as sitting, and this needs to stop. It's possible that the decision tree's preference for categorical variables had an impact on the algorithm's performance with this set of data. The K nearest neighbours technique is used to construct the next model. When it comes to accuracy, it's one of the models that has consistently outperformed the benchmark. The best K closest neighbour algorithm's confusion matrix is shown in table 6. Overall, the method has a relatively low number of misclassifications. The KNN method ensured good inter-cluster classification accuracy because of the two clusters of activities, mobile, and stationary.

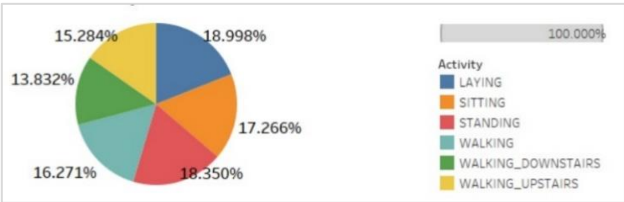


Figure 9. Target feature distribution

Table 5. Decision tree' confusion matrix

	Reference					
Prediction	W	WU	WD	S	ST	L
Walking(W)	441	68	111	0	1	0
Walking_Upstairs (WU)	75	395	310	0	0	4
Walking_Downstairs (WD)	0	0	0	0	0	0
Sitting (S)	0	0	0	0	0	0
Standing (ST)	0	0	0	475	556	44
Laying (L)	0	0	0	58	14	535

Table 6. KNN' confusion matrix

	Reference					
Prediction	W	WU	WD	S	ST	L
W	513	3	12	0	0	0
WU	1	456	6	3	0	0
WD	2	4	403	0	0	0
S	0	0	0	473	52	10
ST	0	0	0	54	519	5
L	0	0	0	3	0	568

However, the accuracy of intra-cluster classification, particularly for immobile activities, was not very satisfactory. Many activities, such as sitting and standing, were incorrectly categorised as sitting or standing. Observing user activity over the course of 2.5 seconds, it is clear that previous and subsequent activities have an impact on the current activity. As a result, the KNN algorithm's good performance can be justified by the fact that it considers the stated number of neighbours when doing classification. Nevertheless, the algorithm's modelling and assessment were computationally taxing and took the most time.

As the name suggests, the Naive Bayes model uses the Bayesian function to compute the needed probability. Table 7 displays the Naive Bayes algorithm's best-performing confusion matrix. When compared to other models' findings and benchmark results, the classification accuracy isn't all that impressive.

Table 7. Naive Bayes' confusion matrix

	Reference					
Prediction	W	WU	WD	S	ST	L
W	443	7	16	0	0	0
WU	47	432	67	10	11	11
WD	26	24	338	0	0	1
S	0	0	0	399	87	4
ST	0	0	0	112	466	0
L	0	0	0	12	7	567

The inherent assumption of parameter independence may be to blame for the Naive Bayes' low performance. Because these metrics were originally collected, ignoring interactions between features can result in the loss of useful information and be detrimental to the project. It's also worth mentioning that this strategy converged the quickest of all. The best artificial neural network (ANN) model's confusion matrix is shown in Table 8. As can be observed, mobile activities and lying down activities were both categorized extraordinarily well, with perfect classification accuracy at 100%. ANN outperformed all other ML algorithms in this experiment is clear from the table 9 results, which shows that it outperformed the benchmark test.

Accuracy estimates from several categorization algorithms are shown in Figure 10. However, using data from the correlation analysis technique, the accuracy distributions for KNN and NB appear positively skewed, whereas those for ANN and logistic regression (LR) are rather normal, with a small variation from the mean. Prediction intervals and confidence intervals for the decision tree are likewise clearly not normal. For data that does not fit the normal distribution, a non-parametric test is the best option for performing tests of significance for the model that had higher projected accuracies.

Table 8. ANN’s confusion matrix

	Reference					
Prediction	W	WU	WD	S	ST	L
W	515	0	8	0	0	0
WU	1	460	4	0	0	0
WD	0	3	409	0	0	0
S	0	0	0	478	47	0
ST	0	0	0	54	524	0
L	0	0	0	1	0	583

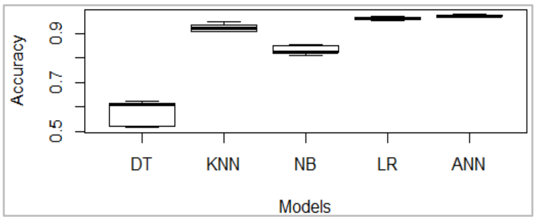


Figure 10. Accuracy distribution categorized by modeling technique

Findings show that the artificial neural network outperforms logistic regression in terms of classification accuracy by a little margin. Figure 11 depicts the outcomes of the LR and ANN models. The outcomes of the two tests are similar, as can be seen. The statistical significance of the ANN algorithm's results was compared to those of the LR method. ANN outperformed all other ML algorithms in this experiment is clear from the Table 9 results, which shows that it outperformed the benchmark test. According to the data presented in Table 9 and Figure 10, the artificial neural network outperformed all of the other machine learning algorithms developed for this experiment by a wide margin.

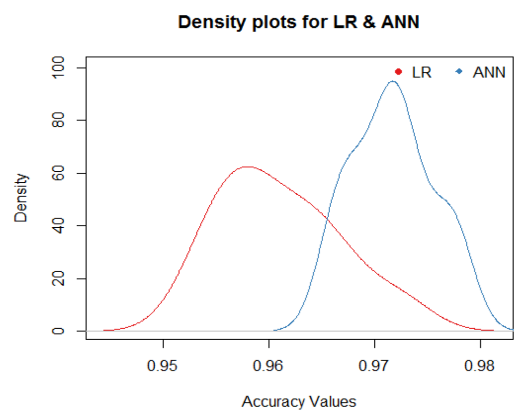


Figure 11. Density plot for LR and ANN

Table 9. Average accuracy (AA) list

Model	Average accuracy
DT	.59
KNN	.93
NB	.84
LR	.96
ANN	**.97**

5. CONCLUSION

Using data from a wearable sensor gadget, the researchers primarily sought to identify a user's physical activity. The first emphasis was on learning the most modern ways for recognizing human activity. Following the collection of relevant data for the study, the research team focused on feature engineering to convert the data so that it could be used by ML algorithms under time and memory constraints. The first method for calculating the correlation coefficients between several features was correlation analysis. Reducing features that have a high correlation with each other is the primary goal of this strategy. The study's purpose was to use machine learning techniques to model the condensed data in order to achieve high predicted accuracy. For both feature engineering strategies, the ANN had the highest average accuracy of 97%. With an average of 96%, the logistic regression algorithm comes in second and the KNN algorithm comes in third. The benchmark multiclass hardware-friendly SVM has been outperformed by these three algorithms. The 83% accuracy of the Naive Bayes is comparable to the benchmark. Only 57% of people were satisfied with their decision tree's performance. For this purpose, we employed a recursive partitioning strategy to describe the decision tree. Gini impureness is utilized as a splitting criterion in this investigation, however information gain can also be used. After seeing that increasing the K value had no positive effect on performance, the tune length of the KNN algorithm was set at 7. In this work, logistic regression was performed using a penalized multinomial regression.

REFERENCES

[1] L. M. Dang, K. Min, H. Wang, M. J. Piran, C. H. Lee, and H. Moon, "Sensor-based and vision-based human activity recognition: A comprehensive survey," *Pattern Recognition*, vol. 108, p. 107561, 2020, doi: 10.1016/j.patcog.2020.107561.

[2] A. Gupta, K. Gupta, K. Gupta, and K. Gupta, "A survey on human activity recognition and classification," in *Proceedings of the 2020 IEEE International Conference on Communication and Signal Processing, ICCSP 2020*, 2020, pp. 915–919, doi: 10.1109/ICCSP48568.2020.9182416.

[3] S. Milanko, A. Launi, and S. Jain, "Designing just-in-time detection for gamified fitness frameworks," 2020, *arXiv:* 2005.08834.

[4] M. E. N. G. Xianjia *et al.*, "Just-in-time human gesture recognition using WiFi signals," *Chinese Journal of Electronics*, vol. 30, no. 6, pp. 1111–1119, 2021, doi: 10.1049/cje.2021.07.022.

[5] M. Langer *et al.*, "What do we want from explainable artificial intelligence (XAI)? – A stakeholder perspective on XAI and a conceptual model guiding interdisciplinary XAI research," *Artificial Intelligence*, vol. 296, p. 103473, 2021, doi: 10.1016/j.artint.2021.103473.

[6] S. Sun, Z. Kuang, L. Sheng, W. Ouyang, and W. Zhang, "Optical flow guided feature: a fast and robust motion representation for video action recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1390–1399, doi: 10.1109/CVPR.2018.00151.

[7] B. Fu, N. Damer, F. Kirchbuchner, and A. Kuijper, "Sensing technology for human activity recognition: A comprehensive survey," *IEEE Access*, vol. 8, pp. 83791–83820, 2020, doi: 10.1109/ACCESS.2020.2991891.

[8] M. A. Khan, Y. D. Zhang, S. A. Khan, M. Attique, A. Rehman, and S. Seo, "A resource conscious human action recognition framework using 26-layered deep convolutional neural network," *Multimedia Tools and Applications*, vol. 80, no. 28–29, pp. 35827–35849, 2021, doi: 10.1007/s11042-020-09408-1.

[9] M. M. Islam, S. Nooruddin, F. Karray, and G. Muhammad, "Human activity recognition using tools of convolutional neural networks: A state of the art review, data sets, challenges, and future prospects," *Computers in Biology and Medicine*, vol. 149, 2022, doi: 10.1016/j.compbiomed.2022.106060.

[10] C. Jobanputra, J. Bavishi, and N. Doshi, "Human activity recognition: A survey," *Procedia Computer Science*, vol. 155, pp. 698–703, 2019, doi: 10.1016/j.procs.2019.08.100.

[11] J. Li, X. Liu, W. Zhang, M. Zhang, J. Song, and N. Sebe, "Spatio-temporal attention networks for action recognition and detection," *IEEE Transactions on Multimedia*, vol. 22, no. 11, pp. 2990–3001, 2020, doi: 10.1109/TMM.2020.2965434.

[12] F. Serpush, M. B. Menhaj, B. Masoumi, and B. Karasfi, "Wearable sensor-based human activity recognition in the smart healthcare system," *Computational Intelligence and Neuroscience*, vol. 2022, 2022, doi: 10.1155/2022/1391906.

[13] P. Paul and T. George, "An effective approach for human activity recognition on smartphone," in *ICETECH 2015 - 2015 IEEE International Conference on Engineering and Technology*, 2015, pp. 1–3, doi: 10.1109/ICETECH.2015.7275024.

[14] D. R. Beddiar, B. Nini, M. Sabokrou, and A. Hadid, "Vision-based human activity recognition: a survey," *Multimedia Tools and Applications*, vol. 79, no. 41–42, pp. 30509–30555, 2020, doi: 10.1007/s11042-020-09004-3.

[15] S. Asghari-Esfeden, M. Sznaier, and O. Camps, "Dynamic motion representation for human action recognition," in *Proceedings - 2020 IEEE Winter Conference on Applications of Computer Vision, WACV 2020*, 2020, pp. 546–555, doi: 10.1109/WACV45572.2020.9093500.

[16] S. Qiu *et al.*, "Multi-sensor information fusion based on machine learning for real applications in human activity recognition: State-of-the-art and research challenges," *Information Fusion*, vol. 80, pp. 241–265, 2022, doi: 10.1016/j.inffus.2021.11.006.

[17] K. Altun and B. Barshan, "Human activity recognition using inertial/magnetic sensor units," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2010, vol. 6219 LNCS, pp. 38–51, doi: 10.1007/978-3-642-14715-9_5.
[18] A. J. Piergiovanni and M. S. Ryoo, "Representation flow for action recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019, vol. 2019-June, pp. 9937–9945, doi: 10.1109/CVPR.2019.01018.
[19] Z. Tu *et al.*, "Multi-stream CNN: Learning representations based on human-related regions for action recognition," *Pattern Recognition*, vol. 79, pp. 32–43, 2018, doi: 10.1016/j.patcog.2018.01.020.
[20] Q. Xiong, J. Zhang, P. Wang, D. Liu, and R. X. Gao, "Transferable two-stream convolutional neural network for human action recognition," *Journal of Manufacturing Systems*, vol. 56, pp. 605–614, 2020, doi: 10.1016/j.jmsy.2020.04.007.
[21] A. Abdelbaky and S. Aly, "Human action recognition using short-time motion energy template images and PCANet features," *Neural Computing and Applications*, vol. 32, no. 16, pp. 12561–12574, 2020, doi: 10.1007/s00521-020-04712-1.
[22] M. C. Leong, D. K. Prasad, Y. T. Lee, and F. Lin, "Semi-CNN architecture for effective spatio-temporal learning in action recognition," *Applied Sciences (Switzerland)*, vol. 10, no. 2, p. 557, 2020, doi: 10.3390/app10020557.
[23] M. A. Khan *et al.*, "Human action recognition using fusion of multiview and deep features: an application to video surveillance," *Multimedia Tools and Applications*, pp. 1–27, 2020, doi: 10.1007/s11042-020-08806-9.
[24] P. E. Martin, J. Benois-Pineau, R. Péteri, and J. Morlier, "Fine grained sport action recognition with Twin spatio-temporal convolutional neural networks: Application to table tennis," *Multimedia Tools and Applications*, vol. 79, no. 27–28, pp. 20429–20447, 2020, doi: 10.1007/s11042-020-08917-3.
[25] Y. Meng *et al.*, "AdaFuse: Adaptive temporal fusion network for efficient action recognition," 2021, *arXiv:2102.05775*.

BIOGRAPHIES OF AUTHORS

Dr. Sabah Mohammed Fayadh received the BSc. in IT from Iraq, MSc. from India in IT, PhD. from College of IT at Babylon University-Iraq. He is a lecturer at Department of computer Systems, Southern Technical University, Thi-Qar, Iraq. His research interests include is artificial Intelligence, and modeling and simulation. He can be contacted at email: sabah.fayadh@stu.edu.iq.

Dr. Elham Mohammed Thabit A. Alsaadi received the BSc. in Computer Science from Iraq, MSc. from UK. Ph.D. from College of IT at Babylon University-Iraq. She is a lecturer in College of CO & IT, Karbala University. Her research interests include artificial intelligence, computer vision, and image processing. She can be contacted at email: elham.thabit@uokerbala.edu.iq.

Huda Hallawi received the Ph.D. from from UK. She got BSc., and MSc. in Computer Engineering and Information Technologyfrom Iraq. Her research interests is evolutionary algorithms, cloud computing, and artificial intelligent. She is a lecturer at Karbala University. She can be contacted at email: huda.f@uokerbala.edu.iq.