Sigma J Eng Nat Sci, Vol. 41, No. 4, pp. 716–728, August, 2023

Sigma Journal of Engineering and Natural Sciences
Web page info: https://sigma.yildiz.edu.tr
DOI: 10.14744/sigma.2023.00082

Research Article

Battery-friendly tiny models for activity recognition on energy-constrained devices

Cihat AKKIRAZ[1], Hafıza İrem TÜRKMEN[1], Mehmet Amaç GÜVENSAN[1,*], Barış CAN[2]

[1]*Department of Computer Engineering, Istanbul Technical University, Istanbul, 34349 Türkiye*
[2]*The Scientific and Technological Research Council of Türkiye, Kocaeli, 41470, Türkiye*

ARTICLE INFO

Article history
Received: 29 March 2021
Accepted: 21 February 2022

Keywords:
Feature Extraction; Instance Selection; Activity Recognition; Resource Constrained Device; Faster and Energy Efficient Classification

ABSTRACT

The number of the applications that analyze and evaluate human activities of daily living such as transport mode detection and activity recognition is increasing rapidly due to the requirements in several fields such as transportation planning, elderly care and ambient assisted living. One of the drawbacks of these systems is their high battery consumption characteristics. In this study, we introduce a novel instance selection methodology that provides energy saving in testing process by reducing the amount of the training data while preserving the accuracy of the system. By their nature, daily living activities separate to several sub-classes within each class. The proposed method selects instances in an iterative cluster-based manner assorting with the characteristic structure of the daily activities. The success of the system is evaluated by applying Decision Tree (J48) and k-Nearest Neighborhood (k-NN) algorithms to two different publicly available daily activity datasets. Obtained results show that the proposed instance selection algorithm based on sub-activity characteristics could achieve up to 11% improvement of the classification results when 50% of the training instances are eliminated. With the help of this selection process, we built 4% to 57% smaller and 4% to 62% faster models for activity recognition on energy-constrained devices.

Cite this article as: Akkiraz C, Türkmen Hİ, Güvensan MA, Can B. Battery-friendly tiny models for activity recognition on energy-constrained devices. Sigma J Eng Nat Sci 2023;41(4):716–728.

INTRODUCTION

In the last decade, a huge amount of environmental and personal data started to be generated/flow on/into our electronic devices including our smartphones, smartwatches, laptops, cloud-computers and etc. With the help of powerful hardware, researchers exploit the obtained big data to build more skill full expert systems, stronger training models, and reliable forecasting architectures. However, with the era of big data the systems started to demand much more resources in terms of storage, processing capability, energy and communication. Especially wearable and portable devices such as smartwatches, smartphones, smart bracelets and so on could survive still only a very limited time with a full-charged battery. Therefore, smart applications should be designed regarding this issue to consume energy attentively.

***Corresponding author.**
*E-mail address: amac@yildiz.edu.tr

This paper was recommended for publication in revised form by Regional Editor Pravin Katare

Published by Yıldız Technical University Press, İstanbul, Turkey

Mobile applications running machine learning algorithms especially 7/24 in the background are the major threat for the battery and storage of mobile devices. Transport mode detection, activity recognition, fall detection, sleep quality analyzer, driving quality scoring and similar applications are very popular among the users to watch human behavior during daily life. Those applications suffer very much from depleting the battery very quickly. There are several solutions to save battery for such cases. The first one is to select the least energy consuming sensors and keep away from the energy-hungry sensors such as GPS though it slightly helps to make accurate decisions for recognition of daily outdoor activities especially for motorized activities. Moreover, it is important to note that GPS usage became widespread due to location-based applications. On the other hand, the system designer should also avoid computational complex operations. For example, simple machine learning algorithms always reduce the consumed energy on resource-constrained devices. As a last solution, researchers try to minimize the number of features and/or instances required for the model training.

In this study, we focus on both transport mode detection and activity recognition problems. There are several studies [1, 2], about recognition of motorized and non-motorized activities. Most of them deals with increasing the success rate of daily activities. Some of them are interested in the existing energy-problem and offer feature reduction [3] or the systems that work on low resolution sensor data [4] as solution. However, to the best of our knowledge, none of the studies exploit the instance selection approach for minimizing both the complexity and size of the model, i.e. the energy consumption.

In this study, we aim at selecting the discriminative instances for generating efficient training models to be run on energy-constrained devices. We propose an instance-selection algorithm exploiting the clustering approach. Since motorized and non-motorized activities consist of several sub-activities, our algorithm aims at finding the representative instances and eliminating the redundant and misleading examples. Thus, we obtain training models with smaller sizes and less computational complexity without sacrificing the success rate.

The main contributions of the study are given as follows;

- We obtain training models with smaller sizes and less computational complexity without sacrificing the success rate. This scheme solves many serious difficulties, such as lack of memory and long processing time.
- Our proposed instance selection approach enables mobile applications that uses energy consuming sensors such as GPS in an energy efficient way.
- Its structure that taking within-class differences into account may contribute developing personalized models for mobile device users.

In the following chapters, we first discuss the available solutions on minimizing energy about activity recognition and instance selection approaches for different domains. Then, we introduce the details of our novel instance selection algorithm in Section 3. In Section 4, we present the experimental results and talk over the results. Finally, we conclude our paper in Section 5.

Related Work

The requirement of analyzing the traffic flow of cities has increased the number of studies published in this area in recent years [5]. These studies exploit various sensors including GPS, accelerometer, barometer and magnetometer. Most of the studies extract time-domain features from gathered sensor data and employ traditional shallow models in order to make classification [6–10]. In [11], researchers used Discrete Hidden Markov Model (DHMM) to classify transportation modes into 8 different classes namely still, walk, run, bike, road, rail, plane and other. They achieved a success rate of up to 96% using GPS information and 94% relying only on an accelerometer and magnetometer. There are a few studies that reveal the effect of post processing techniques on increasing the performance of shallow classification algorithms [12, 13]. Although studies that use time-domain features dominate the literature on transport mode detection, a few studies have also examined the success of frequency-domain features [14].

Analysis of non-motorized activities of humans is another active research area of daily living activity recognition on mobile devices [15]. Most of the proposed approaches perform activity recognition by using on-board sensors of smartphones and employ shallow classification techniques as transport mode detection systems. It is hard to compare the success rates of the presented work due to different experimental setup. Ortiz et.al. achieved a success rate of 96% in classification of lying, sitting, standing, walking, walking upstairs and downstairs, sit-to-stand, sit-to-lie and lie-to-stand groups by combining SVM with a heuristic filtering approach [16]. An interesting work is published by Das et.al. which exploits ordinal classification in order to determine human activities [17]. The researchers obtained significant improvement in shallow techniques including k-Nearest Neighbour (k-NN), Support Vector Machine (SVM) and Random Forest (RF). Their proposed ordinal classification based approach achieved an accuracy value of 97.96% when it is combined with the AdaBoost. Rahman et.al. evaluated the performance of boosting classifiers in activity recognition [18]. A comprehensive survey that compares human activity recognition systems in terms of battery, memory and CPU usage is published by Shoaib et.al. [1]. There are also attempts to reduce battery consumption and annotation cost. In [19], researchers propose a multi-tier architecture that combines thresholding methods with machine learning algorithms in order to provide energy saving. Cruciani et.al. proposed an automatic labeling method that employs a heuristic function which combines step count and GPS information in order to facilitate the collection of labeled datasets. The authors claim that it is possible to obtain labels automatically with an 85% average precision rate [15].

The ability of deep learning architectures to model non-linear data, leads increasing number of studies exploiting deep learning techniques in transport mode detection and activity recognition problem. Fang. et.al. achieved a success rate of 95% in classification of transport modes into still, walk, run, bike and vehicle groups using accelerometer, magnetometer, and gyroscope sensors [20]. In [21], researchers exploit accelerometer sensor and employ Convolutional Neural Network (CNN) to perform transport mode detection. They reported a success rate of 94% in classification of still, walk, bike, rail, train, car and bus groups. Vu et.al [22] use Recurrent Neural Networks and obtain a classification accuracy of 94%. Hassan et.al. used Kernel Principal Component Analysis (KPCA) and Linear Discriminant Analysis (LDA) for extracting robust features of the data gathered from smartphone inertial sensors. Deep Belief Network (DBN) is then feeded with extracted features in order to perform activity recognition [23]. In [24], researchers perform real-time human activity recognition with a success rate of 82% by using accelerometer data and CNNs. Sharma et.al. presented a hybrid deep learning classifier which utilizes the capabilities of the CNN, recurrent neural network (RNN), and deep neural network (DNN) in order to detect the transportation mode at an early stage [25]. In a recent study, researchers proposed a method that combines Long Short Term Memory (LSTM) layer and convolutional layer for classifying eight different transportation modes. They also analyzed the contribution of each sensor to the classification performance which may provide insights into more energy efficient designs [26].

Thanks to big data and deep learning approaches, researchers are able to create models with high accuracy rates. A recent study which is published by Nawaz et.al. fused weather and GPS information to predict the transportation type. They used CNN to extract deep high-level features and then LSTM is employed to learn the sequential patterns [27]. In [28], researchers compared CNN, Ensemble of Autoencoders (EAE) and RF methods for determining transportation mode of the people using GPS data generated by smartphones and achieved best results with RF model. Nevertheless, one of the major constraints of developing mobile applications is the energy consumption problem. There are a lot of work which report that using GPS sensor inevitably increases the energy consumption of mobile devices [5, 29, 30]. Su et.al. used low sampling frequency in order to save battery of a smartphone [31]. In another study [32], researchers provide battery saving by using GSM cell tower information.

Reducing the amount of the data to be processed such as feature selection and dimensionality reduction is an effective way of energy saving which is also possible by instance selection from training set. There are many studies in the literature that make instance selection in different fields [33]. Li et.al. presented a sparse coding based approach to select samples for image classification [34]. Ramirez et.al. proposed an optimization-based method in order to select samples for face recognition [35]. In [36], researches employed fuzzy-means clustering algorithm to remove outliers from the original samples which are derived from network traffic data. Researchers also appealed instance selection for establishing a robust framework for imbalanced data sets, active learning and dealing with big data. In [37] and in [38] authors performed instance selection for active learning approaches. Kuncheva et.al. delineated the contribution of the instance selection to the classification success of imbalanced data by employing several instance selection approaches including random undersampling, AdaBoost-like ensemble of evolutionary undersampling, particle-swarm optimisation, one-sided selection and neighbourhood cleaning rule [39]. Bi et.al. proposed a dynamic active learning-based activity recognition method in order to address training data labelling issue and to overcome the difficulties caused by variety of the activities performed by individuals. Their approach selects the most representative samples by taking their uncertainty and diversity into account. In this way, the annotation cost is significantly decreased [40].

Besides, there are substantial studies that propose a classifier type dependent instance selection approach [41, 42]. Kavrin et.al. proposed a Bagging-based instance selection algorithm for instance-based classification [43]. Pérez et.al. used Bagging of 1-NN and 1-NN algorithms in order to classify 66 binary imbalanced data sets by using SVM algorithm [44]. In [45], researchers proposed a method that serves for instance-based learning approaches especially for k-NN. Researchers exploited CHC (Cross generational elitist selection, Heterogeneous recombination and Cataclysmic mutation) genetic algorithm framework. They combined CHC with two strategies: using local k values for k-NN rule and selecting each instance more than once. They evaluated the performance of the proposed approach on 150 databases with different class and feature numbers by using k-NN.

Max et.al. exploited metric learning in order to transform samples to a more organized form in which samples within same class close to each other while they are far from the samples of other classes [46]. They combined metric learning with four instance selection strategies namely CHC, Condensed Nearest Neighbor, Random Mutation Hill Climbing (RMHC) and Edited Nearest Neighbor (ENN). These instance selection strategies rely on the idea that selecting the most representative instances by removing noisy ones will directly affect the performance of the k-NN classifier. Researchers evaluated the performance of metric learning by applying k-NN classifier to Iris UCI dataset. They claimed that proposed method provide improvement in all of the tested instance selection methods.

Kim et.al. presented two cluster based approaches namely Unlabeled Data Prototypes (UDP) Selection and Labeled Data Counterparts (LDC) Selection [47]. UDP starts with clustering unlabeled instances. Then clusters with low purity were discarded and the most representative instance from each cluster is selected as additional training data. Average cosine similarity is used to determine the samples to be retained. LDC involves selection of the instances from the clusters containing both labeled and unlabeled instances. For each labeled instance the unlabeled instance most similar to it in the same cluster is selected. By that way, new training instances that share features with the original labeled data and maintain the same class distribution are acquired. Researchers employed SVM for classification of clinical texts. They noted that proposed approach brings advantage when large amount of unlabeled data is available but manual annotation is expensive. This study selects instances based on evaluation of the within class similarity.

While within class similarity is critical criteria for instance selection, a different approach should be considered for transport mode detection since it comprises multiple sub clusters within each class. Abdallah et.al. proposed an active learning approach which applies clustering algorithms to the activity classes by the purpose of revealing sub-clusters that represent different patterns within each particular activity [48]. Although that study involves evaluation of the sub-clusters as our proposed system does, the main goal of it is incremental learning from unlabeled data and minimizing labeling costs while we aim at increasing training efficiency in order to obtain less complex and energy-efficient training models and achieve better success rates.

Another study [49] that exploits characteristics of clusters is presented by Czarnowski et.al. They proposed an agent-based instance selection algorithm in order to deal with two class - imbalanced dataset problem. They first perform a similarity-based clustering algorithm in order to determine sub-clusters within each considered class independently. Then, the prototypes of each cluster are used in order to get a balanced dataset. Classification phase was performed by using Agent-based population learning algorithm. Researchers compared their framework with the results which are obtained by employing traditional classification methods such as C4.5 and k-NN algorithms on unreduced, imbalanced dataset. However, the effects of the proposed instance selection method on the performance of traditional classification algorithms is not revealed. Besides, the relationship of the reduction ratio with the training model size, test duration complexity and classification accuracy are not discussed which are essential parameters for energy-constrained devices.

Instance Selection Based on Sub-activity Characteristics

Sensor data of the tasks such as transport mode detection and activity recognition involves multiple sub-classes within each class as a matter of course. For instance, samples within "car" class of the transport mode data, can be divided into multiple sub-classes due to the different car activities including acceleration, deceleration, turning, and/or characteristics of route/driver. The proposed instance selection algorithm is developed by taking these features of the daily activity data into account.

The block diagram of the proposed system is illustrated in Figure 1. The algorithm starts with separating the train set into k+1 folds. Data within each fold is divided into sub-classes by using unsupervised clustering methods. In this study, X-means clustering which is an extended version of K-means algorithm is employed in order to form sub-classes. X-means constitutes optimum cluster number by repetitively partitioning the data [50].

At the second step, initial parameters of the model which represent the sub-classes of each class are calculated by using the data within $fold_0$. The new values of these parameters are transmitted to the instance selection module in order to select the instances of $fold_1$. After inclusion

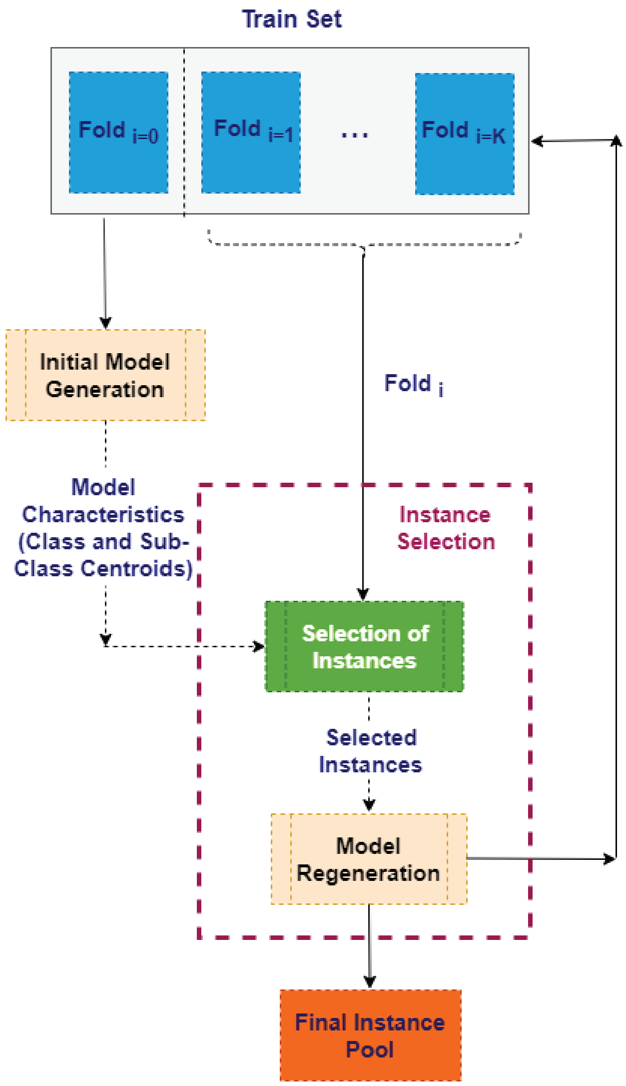


Figure 1. The block diagram of the proposed system.

of new instances, model parameters are recalculated. This iterative process is repeated for each of the folds. The working principle of instance selection module is illustrated in Figure 2.

Testing the instances of $fold_i$ is performed by using two threshold values namely class threshold and sub-class threshold. These thresholds are calculated by evaluating the sub-class distribution of each class within the instance pool, which is gathered from the instances that have been selected so far.

The detailed description of the whole proposed instance selection methodology is given below;

1. Separate train set into k+1 folds
2. Add instances within $fold_0$ to the instance pool
3. Apply xMeans to the each class in the instance pool
4. Calculate $\forall_j \in \{1, ..., S\}, \forall_n \in \{1, ..., N\}$ C_{nj} where S is the number of sub-classes within $class_n$ and C corresponds to the centers of the sub-classes
5. Initialize $ThClass_n$ and $ThSubClass_n$ by evaluating centers of sub-classes
6. For the each remaining fold;
   a. For each sample X which belongs to $class_n$ within $fold_i$; $StatSubClass_n[x] = \forall j \in \{1, ..., S\}, \min(||X- C_{nj}||)$ $StatClass_n[x] = \sum_{j=1}^{S}(||X - C_{nj}||)$
   b. if $((StatSubClass_n\ [x] > ThSubClass_n)$ AND $(StatClass_n\ [x] < ThClass_n))$
      OR
      $((StatSubClass_n\ [x] < ThSubClass_n)$ AND $(StatClass_n\ [x] > ThClass_n))$
      Add X to the instance pool
   c. Apply xMeans to each class in the sample pool
   d. Update $\forall j \in \{1, ..., S\}, \forall n \in \{1, ..., N\}$ C_{nj}

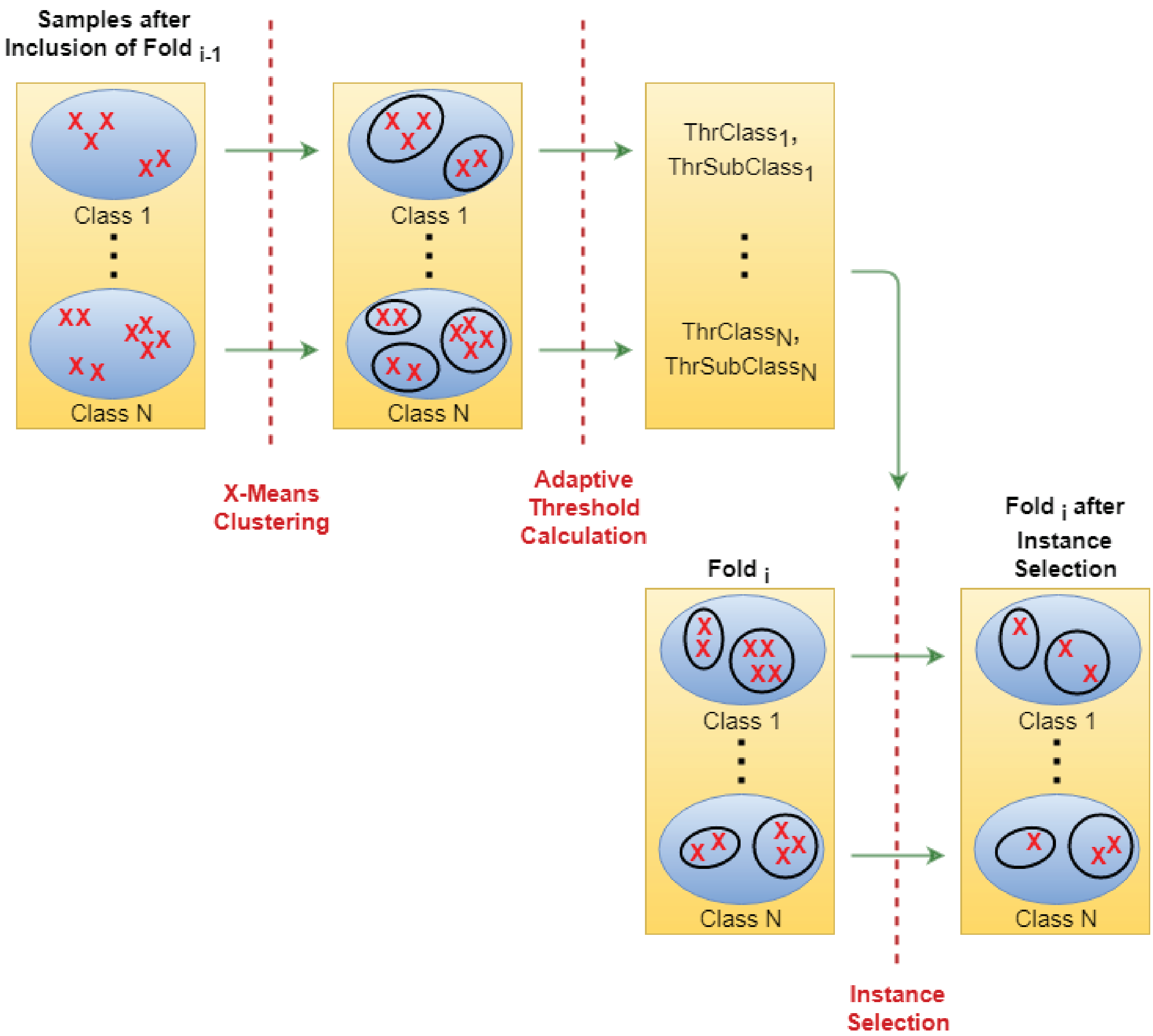


Figure 2. The illustration of the proposed instance selection algorithm.

e. $ThClass_n$=CalculateAdaptiveThr($StatClass_n$, α)
f. $ThSubClass_n$=CalculateAdaptiveThr($StatSubClass_n$, α)

CalculateAdaptiveThr () function returns α_{th} percentile of a given array. Adaptive calculation of the thresholds yields a system that is robust to different datasets while α parameter controls the elimination ratio.

The proposed instance selection algorithm decreases the size of the dataset while preserving the most representative samples. This is accomplished by eliminating the samples which are tend to be outlier or very similar to the existing class members. Step 6.b ensures the inclusion of sample X if it satisfies one of the following conditions;

- Sample X is not close enough to either of the sub-classes within the corresponding class while it closes to the class center. In that case, X might start to form a new sub-class.
- Sample X is close to one of the sub-classes, however it is not close enough to the class center. This situation points out that, the corresponding sub-class cannot be represented efficiently in that class and the weight of that sub-class should be increased to provide full-coverage of the class characteristics.

RESULTS AND DISCUSSIONS

In this study, we exploit two well-known datasets for evaluation of the proposed instance selection algorithm. HTC dataset [51] includes 11 transportation modes such as car, bicycle, train and etc., whereas Mobifall dataset [52] consists of 11 different daily activities including walking, running, climbing up and down, sitting and different fall types. The number of instances for both datasets are given in Table 1 and Table 2. Each instance is represented by 348 time-domain features [12] extracted from accelerometer, gyroscope and magnetometer raw data. All experiments were run for both HTC dataset and Mobifall dataset to show the performance of battery-friendly training models on test operations using our instance selection algorithm. Implementations were performed on a system that uses Intel Core i5-6200u with 12 GB RAM and Ubuntu 17.10 operating system.

Table 1. HTC Dataset

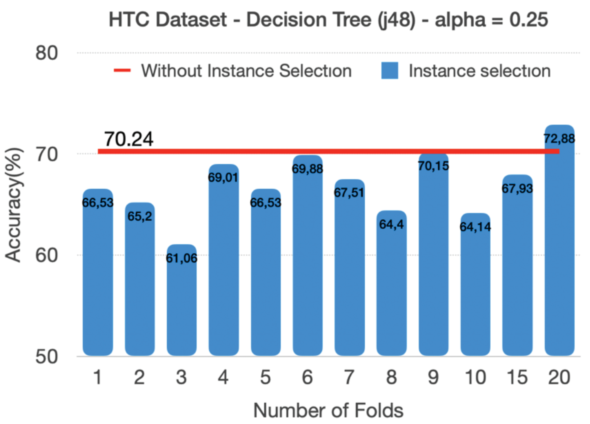
Activity Type	Number of Activities
Walking	8333
Metro	2949
Car	1890
Motorbike	8334
Train	8333
Bicycle	8333
High Speed Rail	8334
Running	6821
Still	8332

Table 2. Mobifall Dataset

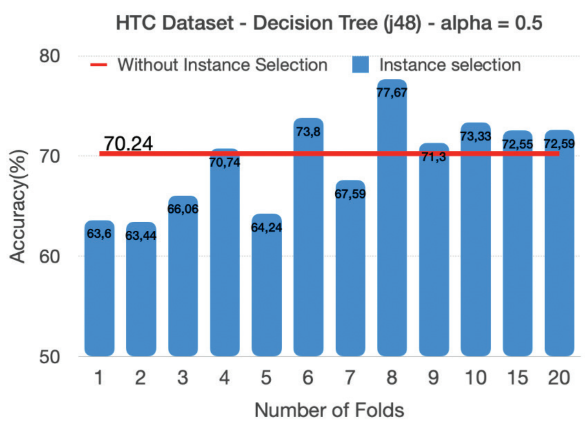
Activity Type	Number of Activities
Standing	1029
Walking	1020
Jogging	299
Jumping	297
Stair up	91
Stair Down	95
Sit Chair	53
Car Step-in	52
Car Step-out	52
Fall	744

There are several pre-processing techniques which would help for creating better training models, such as feature selection and instance selection. They generally improve the success rate and/or reduce the model size and the complexity of the model. On the other hand, most studies exploit the train dataset undividedly during learning phase. However, applying instance selection is an alternative approach for better model accuracy. In this chapter, we will demonstrate how our algorithm helps to build an efficient activity recognition model for energy constrained devices.

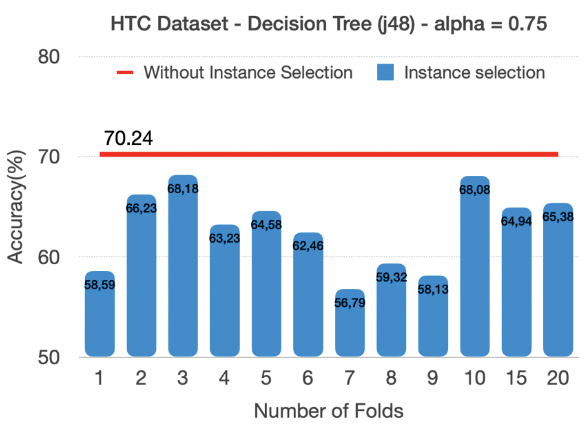
Since our main goal is to eliminate the unnecessary and redundant instances during training, we first examine the impact of the proposed instance selection algorithm for different number of folds, elimination ratio (α) on HTC dataset and Mobifall dataset. Number of folds refers to how many steps are the training model created in, whereas the parameter α indicates how many percentage of the instances might be discarded in each iteration to build a size-friendly classification model. Two different classification algorithms, k-NN and Decision Tree (J48) are selected for the comparison. Weka software was used for training the models and to perform tests [53]. Over a range of 1-7 which is covered in the determination of k in k-NN algorithm, best results occur at k=3. In order to establish a fair comparison, J48 was employed using default hyper parameter set on both datasets and for each elimination ratio. Figure 3, Figure 4, Figure 5, and Figure 6 show us whether the elimination process results in increasing the success rate compared to the model built without instance selection. For J48 algorithm, it is obvious that discarding instances could increase the success rate up to 11% as shown in Figure 3b. On the other hand, models built with the proposed instance selection algorithm for k-NN algorithm produce similar success rates to the traditional single-step model without instance selection.



(a)

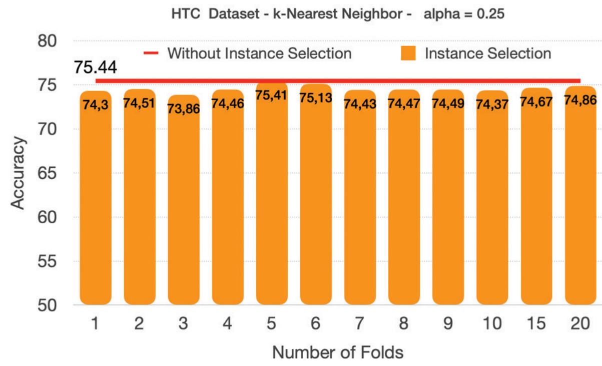


(b)

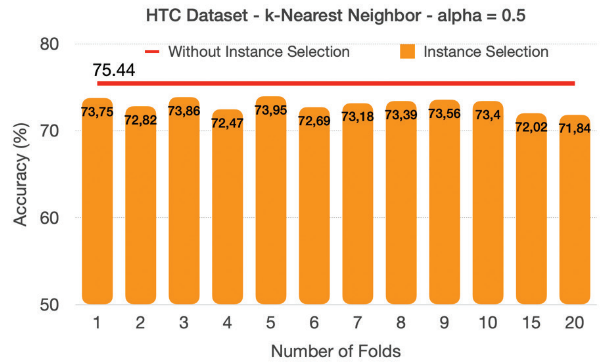


(c)

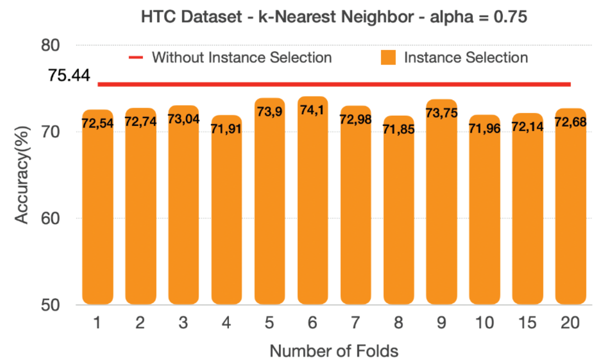
Figure 3. The performance of J48 Decision Tree on HTC Dataset (a) After eliminating approx. 25% of instances (b) After eliminating approx. 50% of instances (c) After eliminating approx. 75% of instances.



(a)



(b)

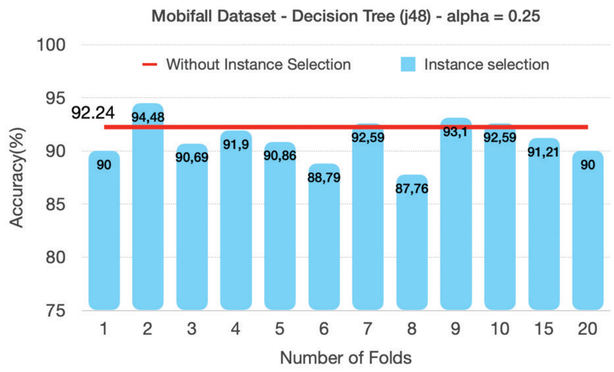


(c)

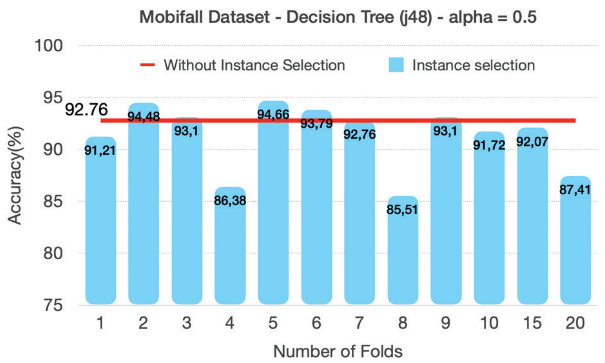
Figure 4. The performance of k-NN algorithm on HTC Dataset (a) After eliminating approx. 25% of instances (b) After eliminating approx. 50% of instances (c) After eliminating approx. 75% of instances.

Analyzing Figure 3,4,5,6 indicates that dividing the training set into 5 or 6 pieces and then applying our instance selection algorithm for each piece gives the best result. However, the optimum number of folds might vary for different datasets and algorithms. All machine learning algorithms have certain hyper parameters that must be tuned based on the dataset and the expectations. Accordingly, the number of folds and the elimination ratio should be considered as hyper parameters for building battery-friendly tiny training models.

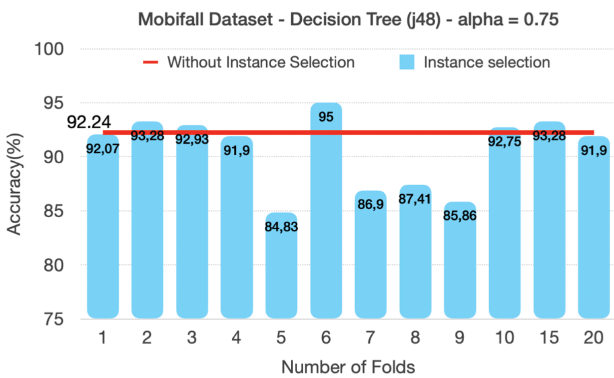
We obtain promising success rates even after eliminating quite high number of instances. However, our main



(a)

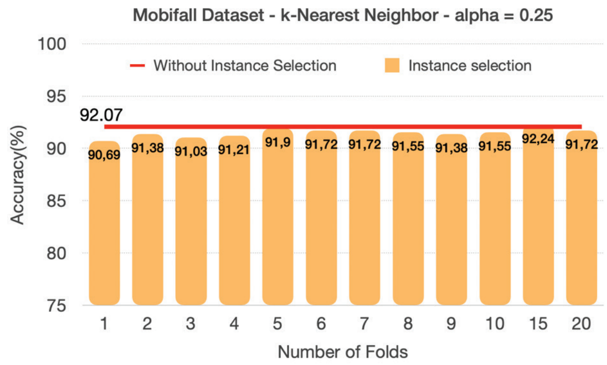


(b)

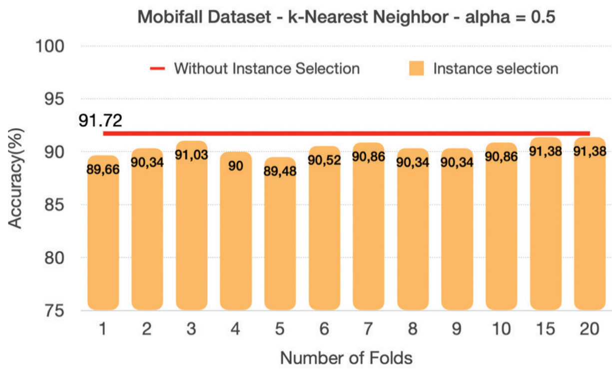


(c)

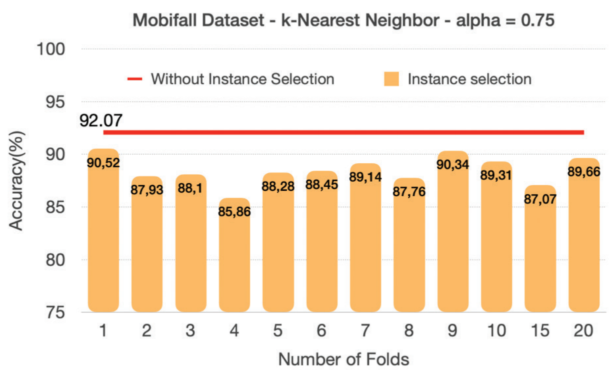
Figure 5. The performance of J48 Decision Tree on Mobifall Dataset (a) After eliminating approx. 25% of instances (b) After eliminating approx. 50% of instances (c) After eliminating approx. 75% of instances.



(a)



(b)

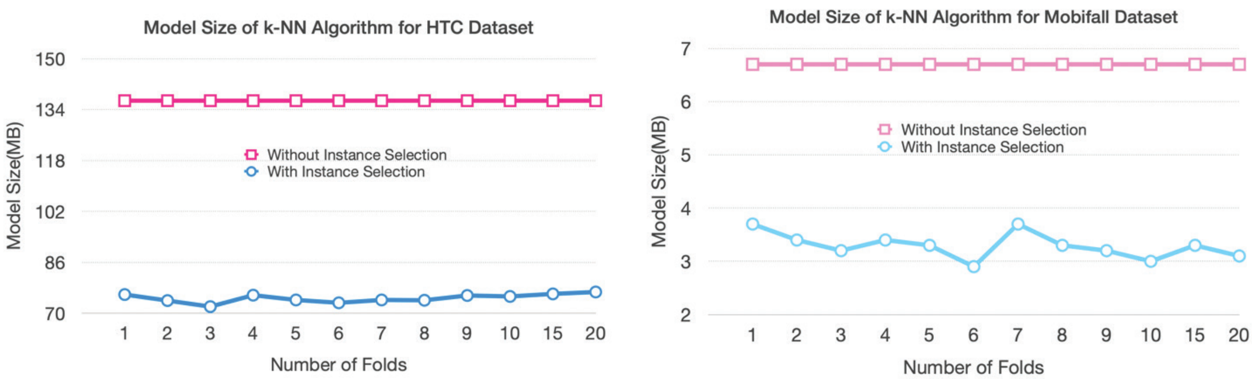


(c)

Figure 6. The performance of k-NN algorithm on Mobifall Dataset (a) After eliminating approx. 25% of instances (b) After eliminating approx. 50% of instances (c) After eliminating approx. 75% of instances.

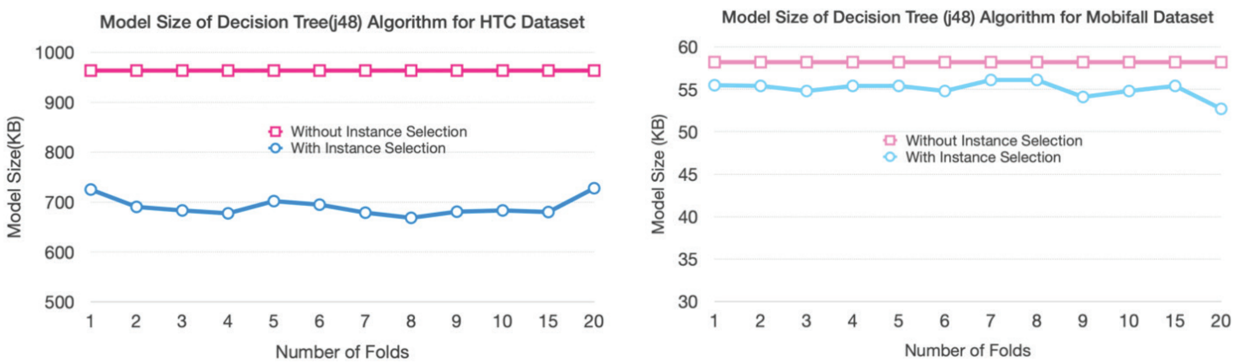
goal with designing such an instance selection algorithm is to reduce the model complexity and test duration of the decision process, i.e. energy consumption, especially for energy-constrained mobile devices. Our test results, shown in Figure 7 and Figure 8 show that with the help of our algorithm we could create up to 30% and 58% smaller models than the traditional single-step training approach without instance selection for J48 and k-NN algorithms, respectively. Our algorithm performs better for k-NN classifier than Decision Tree (J48) algorithm in terms of creating smaller models. However, at the worst case, selecting instances with our algorithm creates 9% smaller models.

It is very critical to determine how many and which instances should be discarded out of the model during the training phase. To find an optimal value, we conducted



(a) (b)

Figure 7. The comparison of model sizes for k-NN algorithm (a) The comparison of model sizes of k-NN algorithm applying with/without instance selection algorithm for Mobifall dataset (b) The comparison of model sizes of k-NN algorithm applying with/without instance selection algorithm for HTC dataset.



(a) (b)

Figure 8. The comparison of model sizes for J-48 Decision Tree algorithm (a) The comparison of model sizes of Decision Tree algorithm applying with/without instance selection algorithm for Mobifall dataset (b) The comparison of model sizes of Decision Tree algorithm applying with/without instance selection algorithm for HTC dataset.

several tests for different elimination ratios by changing the parameter α. In Figure 9 and Figure 10, we demonstrate the accuracy rates of the k-NN and Decision Tree(J48) algorithms in terms of the number of eliminated instances. The number of folds that used during training procedure was set as 5 during the experiments. For both datasets, 25% and 50% elimination rates give better results compared to 75% elimination of instances. In all cases except performing k-NN to HTC dataset, instance selection procedure not only created smaller models but also improved the classification accuracy in regard of the models obtained without instance selection. We can say that discarding 25% of instances than 50% performs a notch better for k-NN algorithm, whereas J48 algorithm gives more accurate results for the parameter $\alpha = 0.5$.

As an important metric for such solutions, we evaluated the test duration of the obtained classification model with/without instance selection per instance. In Table 3, we observe that the obtained classification model for HTC Dataset with instance selection approach could make decisions faster for both k-NN and J48 algorithms up to 61% and 13%, respectively. On the other hand, Table 4 shows that measured gain ratios of these algorithms are 62% and 10% after applying the proposed instance selection algorithm for Mobifall Dataset. In the tests performed using the two algorithms on both datasets, the gain ratios that increase in direct proportion with the instance selection rate are revealed. The achieved results indicate the consistency and reliability of the proposed instance selection algorithm.

On the other hand, our main contribution is building energy-efficient models for battery-constrained devices such as smartphones and smartwatches. To demonstrate the battery saving ability of the proposed algorithm, we run several tests on a smartphone, VESTEL Venus Z20 [54], for four different cases. Since the training process is executed

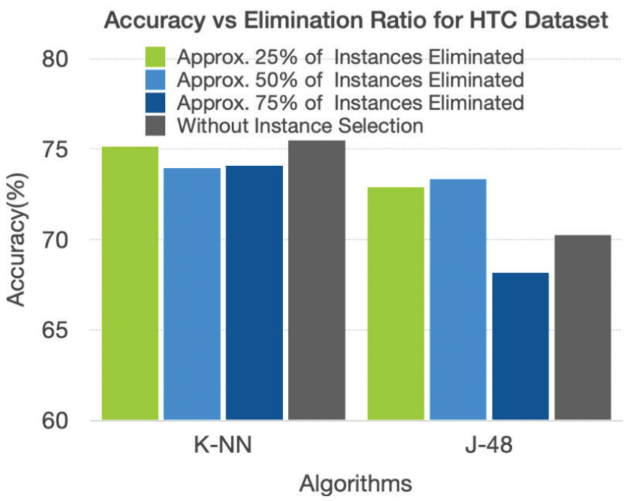


Figure 9. Accuracy for different Elimination Ratios on HTC Dataset.

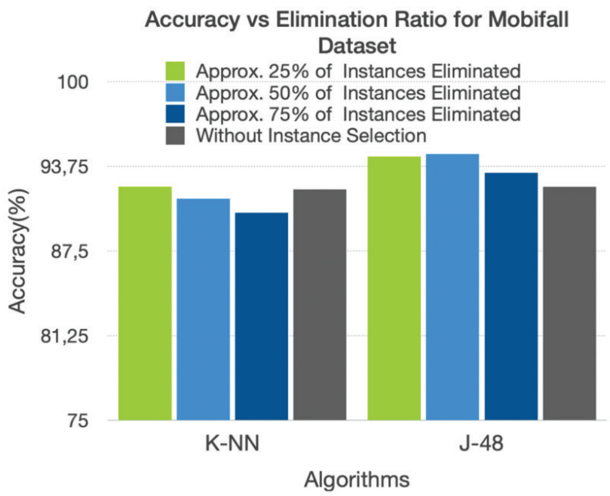


Figure 10. Accuracy for different Elimination Ratios on Mobifall Dataset.

Table 3: Test Duration for HTC Dataset

J48 Algorithm		
Instance Selection	**W/out Instance Selection**	**Gain Ratio (%)**
0.0126 ms (α = 0.25)		3.08%
0.0124 ms (α = 0.5)	0.013ms	4.62%
0.0116 ms (α = 0.75)		10.77%
k-NN Algorithm		
Instance Selection	**W/out Instance Selection**	**Gain Ratio (%)**
135.922 ms (α = 0.25)		16.23%
95.759 ms (α = 0.5)	162.261 ms	40.98%
62.500 ms (α = 0.75)		61.48%

Table 4. Test Duration for Mobifall Dataset

J48 Algorithm		
Instance Selection	**W/out Instance Selection**	**Gain Ratio (%)**
0.0064 ms (α = 0.25		4.48%
0.0062 ms (α = 0.5)	0.0067 ms	7.46%
0.0060 ms (α = 0.75)		10.45%
k-NN Algorithm		
Instance Selection	**W/out Instance Selection**	**Gain Ratio (%)**
14.56ms (α = 0.25)		22.22%
9.58 ms (α = 0.5)	18.72 ms	48.82%
7.10 ms (α = 0.75)		62.07%

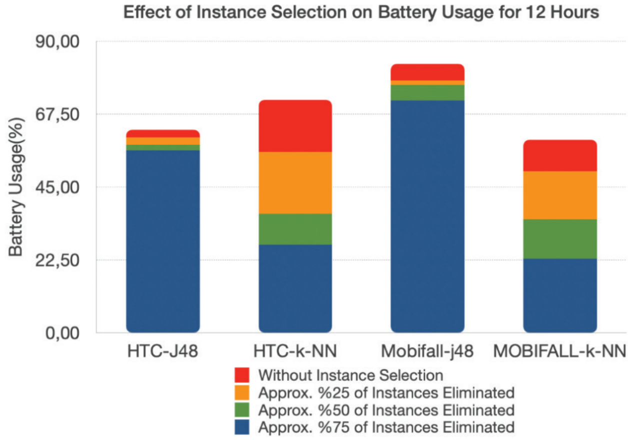


Figure 11. Battery Performance of the proposed Instance Selection Mechanism.

once on deployment of the system, testing performance matters in terms of daily usage. Therefore, we measured the battery usage of the system while the system is running on test mode. The results, shown in Figure 11, verify that, our proposed instance selection algorithm could save 6 to 44 points battery capacity compared to training without instance selection. Additionally, Figure 11 shows that energy saving for k-NN is far much more than J-48 algorithm as expected, since in k-NN, test duration is directly proportional to the number of instances whereas in J-48, both the number of instances and the correlation between them have an effect on the test duration and model size.

CONCLUSION

Widespread usage of energy-constrained mobile devices has increased the diversity in the mobile applications and their user profiles. In this study, an instance selection method which is designed by evaluating structural features of the daily activities is presented. The proposed method works in an iterative and cluster-based manner in order to increase training efficiency. The performance evaluation of the system is carried out for the combinations of two different datasets, two classifiers and three elimination ratios. It is observed that the proposed instance selection algorithm could increase the success rate of the system up to 11% while reducing the test duration and model size of the system by 62% and 58% respectively. Especially the accuracy results for J-48 algorithm is slightly better after instance selection both for HTC dataset and Mobifall dataset, whereas k-NN performance is just chasing the performance of without instance selection both for two datasets. Achieved results demonstrate that presented instance selection approach can contribute to the solutions to energy and resource constraints of daily activity recognition applications especially on energy saving. With the help of instance selection algorithm, the battery usage of a mobile phone for activity recognition applications could be reduced by up to 44 unit in terms of percentage. Battery saving is up to 12 units for J-48 algorithms, whereas the energy consumption could be saved more than half of without instance selection approach for k-NN algorithm. The proposed algorithm could be applied to different datasets by tuning the optimum fold number and elimination ratio parameters on validation set. On both datasets, elimination ratio of 0.25 and 0.5 generally gave better results than 0.75 in terms of accuracy. However, number of folds and elimination ratio could be described as two hyper-parameters for the introduced instance selection mechanism.

As a future work, we first aim at conducting the same experiments on deep learning algorithms. We then will investigate for a better model for deep learning algorithms. We also will try to optimize the hyper-parameters of the introduced mechanism.

NOMENCLATURE

S Number of sub-classes within $class_n$
C Centers of the sub-classes

Greek symbols
α Elimination ratio.

AUTHORSHIP CONTRIBUTIONS

Authors equally contributed to this work.

DATA AVAILABILITY STATEMENT

The authors confirm that the data that supports the findings of this study are available within the article. Raw data that support the finding of this study are available from the corresponding author, upon reasonable request.

CONFLICT OF INTEREST

The author declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

ETHICS

There are no ethical issues with the publication of this manuscript.

REFERENCES

[1] Shoaib M, Bosch S, Incel O, Scholten H, Havinga P. A survey of online activity recognition using mobile phones. Sensors 2015;15:2059–2085. [CrossRef]
[2] Nikolic M, Bierlaire M. Review of transportation mode detection approaches based on smartphone data. 17th Swiss Transport Research Conference, 17th Swiss Transport Research Conference Monte Verità / Ascona, May 17-19, 2017.

[3] Kansiz AO, Guvensan MA, Turkmen HI. Selection of time-domain features for fall detection based on supervised learning. Proceedings of the World Congress on Engineering and Computer Science 2013 Vol II WCECS 2013, 23-25 October, 2013, San Francisco, USA, 2013:2325.
[4] Zheng L, Wu D, Ruan X, Weng S, Peng A, Tang B, et al. A novel energy-efficient approach for human activity recognition. Sensors 2017;17:2064. [CrossRef]
[5] Prelipcean AC, Gidófalvi G, Susilo YO. Transportation mode detection-an in-depth review of applicability and reliability. Trans Rev 2017;37:442–464. [CrossRef]
[6] Quintella CA, Andrade LC, Campos CAV. Detecting the transportation mode for context-aware systems using smartphones. 2016, IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), pp. 2261-2266, IEEE. [CrossRef]
[7] Jahangiri A, Rakha HA. Applying machine learning techniques to transportation mode recognition using mobile phone sensor data. IEEE trans Intell Transp Syst 2015;16:2406–2417. [CrossRef]
[8] Fang SH, Liao HH, Fei YH, Chen KH, Huang JW, Lu YD, Tsao Y. Transportation modes classification using sensors on smartphones. Sensors 2016;16:1324. [CrossRef]
[9] Xiao G, Juan Z, Gao J. Travel mode detection based on neural networks and particle swarm optimization. Information 2015;6:522–535. [CrossRef]
[10] Xiao G, Juan Z, Zhang C. Travel mode detection based on gps track data and bayesian networks. Comput Environ Urban Syst 2015;54:14–22. [CrossRef]
[11] Lorintiu O, Vassilev A. Transportation mode recognition based on smartphone embedded sensors for carbon footprint estimation. 2016, IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), pp. 1976–1981. [CrossRef]
[12] Guvensan M, Dusun B, Can B, Turkmen H. A novel segment-based approach for improving classification performance of transport mode detection. Sensors 2018;18:87. [CrossRef]
[13] Zhou X, Yu W, Sullivan WC. Making pervasive sensing possible: Effective travel mode sensing based on smartphones. Comput Environ Urban Syst 2016;58:52–59. [CrossRef]
[14] Acharjee D, Mukherjee A, Mandal J, Mukherjee N. Activity recognition system using inbuilt sensors of smart mobile phone and minimizing feature vectors. Microsyst Technol 2016;22:2715–2722. [CrossRef]
[15] Cruciani F, Cleland I, Nugent C, McCullagh P, Synnes K, Hallberg J. Automatic annotation for human activity recognition in free living using a smartphone. Sensors 2018;18:2203. [CrossRef]
[16] Reyes-Ortiz JL, Oneto L, Samà A, Parra X, Anguita D. Transition-aware human activity recognition using smartphones. Neurocomputing 2016;171:754–767. [CrossRef]
[17] DAŞ DB, Birant D. Ordered physical human activity recognition based on ordinal classification. Turk J Electr Eng Comput Sci 2021;29:2416–2436. [CrossRef]
[18] Rahman S, Irfan M, Raza M, Moyeezullah Ghori K, Yaqoob S, Awais M. Performance analysis of boosting classifiers in recognizing activities of daily living. Int J Environ Res Public Health 2010;17:1082. [CrossRef]
[19] Guvensan M, Kansiz A, Camgoz N, Turkmen H, Yavuz A, Karsligil M. An energy-efficient multitier architecture for fall detection on smartphones. Sensors 2017;17:1487. [CrossRef]
[20] Fang SH, Fei YX, Xu Z, Tsao Y. Learning transportation modes from smartphone sensors based on deep neural network. IEEE Sensors J 2017;17:6111–6118. [CrossRef]
[21] Liang X, Wang G. A convolutional neural network for transportation mode detection based on smartphone platform. IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems (MASS). 2017, p. 338–342. [CrossRef]
[22] Vu TH, Dung L, Wang JC. Transportation mode detection on mobile devices using recurrent nets. Proceedings of the 24th ACM international conference on Multimedia. 2016, 392-396, ACM. [CrossRef]
[23] Hassan MM, Uddin MZ, Mohamed A, Almogren A. A robust human activity recognition system using smartphone sensors and deep learning. Future Gener Comput Syst 2018;81:307–313. [CrossRef]
[24] Ignatov A. Real-time human activity recognition from accelerometer data using convolutional neural networks. Appl Soft Comput 2018;62:915–922. [CrossRef]
[25] Sharma A, Singh SK, Udmale SS, Singh AK, Singh R. Early transportation mode detection using smartphone sensing data. IEEE Sensors J 2020;21:15651–15659. [CrossRef]
[26] Friedrich B, Lübbe C, Hein A. Analyzing the importance of sensors for mode of transportation classification. Sensors 2021;21:176. [CrossRef]
[27] Nawaz A, Zhiqiu H, Senzhang W, Hussain Y, Khan I, Khan Z. Convolutional LSTM based transportation mode learning from raw GPS trajectories. IET Intell Transp Syst 2020;14:570–577. [CrossRef]
[28] Muhammad AR, Aguiar A, Mendes-Moreira J. Transportation Mode Detection from GPS data: A Data Science Benchmark study. IEEE International Intelligent Transportation Systems Conference. 2021, pp. 3726–3731. [CrossRef]
[29] Bedogni L, Felice MDi, Bononi L. Context-aware android applications through transportation mode detection techniques. Wirel Commun Mob Comput 2016;16:2523–2541. [CrossRef]
[30] Shin D, Aliaga D, Tunçer B, Arisona SM, Kim S, Zünd D, Schmitt G. Urban sensing: Using smartphones for transportation mode classification. Comput Environ Urban Syst 2015;53:76–86. [CrossRef]

[31] Su X, Caceres H, Tong H, He Q. Online travel mode identification using smartphones with battery saving considerations. IEEE Trans Intell Transp Syst 2016;17:2921−2934. [CrossRef]

[32] Bloch A, Erdin R, Meyer S, Keller T, Spindler A De. Battery-efficient transportation mode detection on mobile devices. 16th IEEE International Conference on Mobile Data Management 2015;1:185−190. [CrossRef]

[33] Xu X, Liang T, Zhu J, Zheng D, Sun T, Review of classical dimensionality reduction and sample selection methods for large-scale data processing. Neurocomputing 2019;328:5−15. [CrossRef]

[34] Li X, Fang M, Zhang JJ, Wu J. Sample selection for visual domain adaptation via sparse coding. Signal Processing: Image Communication 2016;44:92−100. [CrossRef]

[35] Santiago-Ramirez E, Gonzalez-Fraga JA, Gutierrez E, Alvarez-Xochihua O. Optimization-based methodology for training set selection to synthesize composite correlation filters for face recognition. Signal Process Image Commun 2016;43:54−67. [CrossRef]

[36] Xuetong N. FCM-LSSVM based on training sample selection. Metallurgical & Mining Indust 2015;9:751−757.

[37] Yang Y, Ma Z, Nie F, Chang X, Hauptmann AG. Multi-class active learning by uncertainty sampling with diversity maximization. Int J Comput Vision 2015;113:113−127. [CrossRef]

[38] Guo Y, Ma L, Zhu F, Liu F. Selecting training samples from large-scale remote-sensing samples using an active learning algorithm,. International Symposium on Computational Intelligence and Intelligent Systems. 2015, pp. 40−51, Springer. [CrossRef]

[39] Kuncheva LI, Arnaiz-González Á, Díez-Pastor JF, Gunn IA. Instance selection improves geometric mean accuracy: a study on imbalanced data classification. Prog Artif Intell 2019;8:215−228. [CrossRef]

[40] Bi H, Perello-Nieto M, Santos-Rodriguez R, Flach P. Human activity recognition based on dynamic active learning. IEEE J. Biomed. Health Inform 2020;25:922−934. [CrossRef]

[41] Olvera-López JA, Carrasco-Ochoa JA, Martínez-Trinidad JF, Kittler J. A review of instance selection methods. Artif Intell Rev 2010;34:133−143. [CrossRef]

[42] Aslani M, Seipel S. Efficient and decision boundary aware instance selection for support vector machines. Inform Sci 2021;577:579−598. [CrossRef]

[43] Kavrin D, Subbotin S. Bagging-based instance selection for instance-based classification. Proceedings of The Fifth International Workshop on Computer Modeling and Intelligent Systems (CMIS-2020). 2020:769−783. [CrossRef]

[44] Rosales-Pérez A, García S, Gonzalez JA, Coello CAC, Herrera F. An evolutionary multiobjective model and instance selection for support vector machines with pareto-based ensembles. IEEE Trans Evol Comput 2017;21:863−877. [CrossRef]

[45] de Haro-García A, Pérez-Rodríguez J, García-Pedrajas N. Combining three strategies for evolutionary instance selection for instance-based learning. Swarm Evol Comput 2018;42:160−172. [CrossRef]

[46] Max EZ, Marcacini RM, Matsubara ET. Improving instance selection via metric learning. International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, 2018. pp. 1−6. [CrossRef]

[47] Kim Y, Riloff E, Meystre SM. Exploiting unlabeled texts with clustering-based instance selection for medical relation classification. AMIA Annual Symposium Proceedings. 2017;2017:1060, American Medical Informatics Association.

[48] Abdallah ZS, Gaber MM, Srinivasan B, Krishnaswamy S. Adaptive mobile activity recognition system with evolving data streams. Neurocomputing 2015;150:304−317. [CrossRef]

[49] Czarnowski I, Jędrzejowicz P. Cluster-based instance selection for the imbalanced data classification. International Conference on Computational Collective Intelligence. 2018, pp. 191−200, Springer. [CrossRef]

[50] Pelleg D, Moore AW. X-means: Extending k-means with efficient estimation of the number of clusters. Proceedings of the 17th International Conference on Machine Learning, June 2000, San Francisco, 2000;1:727−734.

[51] Yu MC, Yu T, Wang SC, Lin CJ, Chang EY. Big data small footprint: the design of a low-power classifier for detecting transportation modes. Proceedings of the VLDB Endowment. 2014;7,13:1429−1440. [CrossRef]

[52] Vavoulas G, Pediaditis M, Spanakis EG, Tsiknakis M. The mobifall dataset: An initial evaluation of fall detection algorithms using smartphones. 13th IEEE International Conference on BioInformatics and BioEngineering, 2013. pp. 1−4. [CrossRef]

[53] Garner SR. Weka: The waikato environment for knowledge analysis. Proceedings of the New Zealand Computer Science Research Students Conference, 1995. pp. 57−64.

[54] DroidChart. Vestel venus z20 device specifications. Available at: https://droidchart.com/en/vestel/vestel-venus-z20-z20-501 Last Accessed Date: 27.07.2023.