Human Activity Recognition using Sensor Data
CSCI-P556: Applied Machine Learning
Spring 2019
Final Project Report

Bivas Maiti[1], Darshan Shinde[1], Virendra Wali[1]

Abstract— **Physical activity recognition - commonly known as Human Activity Recognition (HAR) - has emerged as a key research area in modern Computer Science fields. The main goal of the activity recognition is to help users with their tasks by providing information about their behavior. There are several domains in real-world such as sports, entertainment, health care sector, etc which have a number of applications based on activity recognition. In this project, we try to answer three basic questions related to activity recognition, by implementing different known Machine Learning Algorithms and experimenting with them to gain insights.**

I. INTRODUCTION

There are many consumer products which are based on activity recognition. Nintendo Wii, Xbox Kinect and Sony PS4 are few examples of gaming consoles which uses activity recognition like as gestures or body movements to improve game experience for the players. Although it was originally developed for entertainment sector, it has number of a applications in other sectors like as sports and health care, etc. [5]. Many of the fitness devices which are based on activity recognition has the main goal of providing early assistance to encourage humans to adopt a healthy lifestyle. For rehabilitation, to support medical diagnosis, it is also very important in monitoring daily activity to help patients with improvements after surgeries in chronic impairments.

Although lot of work already done in recognizing activities from on-body inertial sensors and in prototyping and developing a system to recognize activity [2] [3], developing HAR systems that meet application and user requirements remains a challenging task.

While there are multiple common research challenges in HAR and the more general field of pattern recognition, there are few unique challenges in HAR as mentioned in [1]. The most common problem is intraclass variability. This can occur because of a fact that same activity can be performed differently by different individual. This can also occur if an activity is performed by same individual. There is another common problem, interclass similarity. There are many activities which requires similar kind of movements but still are different in nature. So classifying these activities from each other is also one of the challenge. Our main goal of this project is to solve this problem. Through this project we are trying to answer following three question.

1) For a given individual, can we predict the activity based on provided 5 seconds of data coming from on-body sensors?
2) For a given activity, can we identify the subject based on provided 5 seconds of data coming from on-body sensors?
3) By using the knowledge learnt in previous two problems can we develop a more generalized system which can predict the activity without any prior information about the subject?

While activity recognition and some other fields like as computer vision, natural language processing and speech recognition has some challenges, there are few unique challenges in activity recognition and it requires a dedicated set of computational techniques in addition to those developed in computer vision and speech recognition. Activity recognition tasks mainly differs from other fields like computer vision and speech recognition on three fronts [1].

First, unlike others there is not common definition, language or clear and common problem statement. Computer vision has clear problem definition like as 'detect object in image' or and speech recognition has clear definition as 'detect spoken word in a sentence' and they both focus on a fixed and specific type of sensing system.

Second, HAR is very different than computer vision and speech recognition in terms of system design and implementation. Human activity is very diverse and needs careful selection of different types of sensors which differ in their capability and characteristics. Number of sensors and their positions are also one of the important aspects of the system which completely depends on the application requirement.

Finally, activity recognition typically requires specific evaluation metric to measure the quality of the system.

Rest of the paper is subdivided as follows: Section 2 describes the previous related work. Section 3 is about dataset we have used to train the model and for experiments. Section 4 talks in detail about data preprossessing, different models we have implemented for training, and problems we have faced. Section 5 is about experiments and evaluations and finally section 6 concludes the paper along with some discussions on the possible future work.

II. PREVIOUS WORK

Activity Recognition is an important field of research, and various previous work has been done on this subject. In [4], Casale P., Pujol O. and Radeva P. used wearable

[1]Bivas Maiti, Darshan Shinde and Virendra Wali are graduate students at the School of Informatics, Computing, and Engineering, Indiana University, Bloomington. All of them have equally contributed to this project.

accelerometer data to predict activities. They used feature engineering and extraction and Random Forest Classifier to obtain encouraging results. They used 1 second window of accelerometer data and extracted the Root Mean Squared value of integration and mean value of Minmax Sums of all the rows in the window. In contrast, our window based approach takes into consideration the entire data, and not just the engineered features. In [8] , Mannini and Sabatini presented a Hidden Markov model based approach to tackle this problem. In [7], Lester, Choudhury and Borriello propose a static + HMM classifier which also achieves decent precision and recall values.They have also done significant research on the issues of location sensitivity and spatial placement of the sensors on the subject bodies, the number of sensors required and intra-class variability. In 2012, Reiss published the PAMAP2 database [11] which contained more than just accelerometer data and bench marked accuracies for different machine learning models. Our work uses the same database to train and test our models. We also add human features like height, weight, resting heart rate, etc. along with the data to tackle the issue of intra-class variability.

III. DATASET & FEATURES

For this project we are using the PAMAP2 dataset [11] which is created and published by *Attila Reiss* in August 2012. This is a Physical Activity Monitoring dataset of 3 inertial sensors and a heart sensor for 18 different physical activities performed by 9 different subjects. Author of the dataset has used 3 Colibri wireless inertial measurement units (IMU) which has sampling frequency of 100 Hz and 1 HR-monitor (heart rate monitor): BM-CS5SR from BM innovations GmbH1 which has sampling frequency of 9 Hz. The data collection has been performed on 9 subjects which includes 8 male and 1 female candidates of age in between 25 to 32 years. Candidates use all three IMUs (1 on the wrist of the dominant arm, 1 on the chest and 1 on the dominant side ankle) during all experiments. Candidates perform 12 different activities as per the protocol.

Dataset has already been synchronized and labeled as per activities. Dataset contains 54 columns per row, column description is as follows:

- 1 time-stamp (s)
- 2 activityID
- 3 heart rate (bpm)
- 4-20 IMU hand
- 21-37 IMU chest
- 38-54 IMU ankle

Data coming from each IMU has 17 different columns as follows:

- 1 temperature (°C)
- 2-4 3D-acceleration data (ms-2), scale: $\pm$16g
- 5-7 3D-acceleration data (ms-2), scale: $\pm$6g
- 8-10 3D-gyroscope data (rad/s)
- 11-13 3D-magnetometer data (μT)
- 14-17 orientation (invalid in this data collection)

Activity id for all different 12 activities are as follows:

- 1 lying
- 2 sitting
- 3 standing
- 4 walking
- 5 running
- 6 cycling
- 7 Nordic walking
- 12 ascending stairs
- 13 descending stairs
- 16 vaccum cleaning
- 17 ironing
- 24 rope jumping
- 0 other

Author of the dataset has suggested that data for activity 0 should be discarded as it is nothing but the sensor reading for the time in between performing different activities. e.g., going from one location to next's activity location, or waiting for the preparation of some equipment. He has recommended to use data from first accelerometer (with the scale of +-16g), because another accelerometer is not precisely calibrated and gets saturated sometimes for certain movements. There are some missing data due to wireless data dropping which are represented by NaN. Dataset has a row for each 0.01 second (because IMU has sampling frequency of 100 Hz), but sampling frequency of HR monitor is only 9 Hz i.e., we will have only 1 value for every 11 rows of data from IMU. These missing data values are also represented by NaN. Data for some categories are completely missing for all subject due to problems in hardware setup.

IV. IMPLEMENTATION

A. PRE-PREOCESSING

Dataset is already synchronized, but still the dataset needs to be preprocessed before fitting different machine learning algorithms. As the author of the dataset has suggested, we should discard the data for activity 0, the rows of the dataset which are marked with activity id 0 is nothing but the sensor reading for the time in between performing different activities. Hence, we have discarded all rows having activity id as 0. There are 4 columns of orientation in data coming from each IMU. This data is marked as invalid for this data collection protocol by the author. There are a total of 12 such columns (4 columns for all 3 IMUs) in the dataset. Those columns are discarded from the dataset. The accelerometer with scale +-6g is not calibrated and hence 9 columns corresponding to the same has been discarded, as recommended by the author. We can have only one value for HR monitor for every 11 data rows coming from IMU. For all other rows the value of HR monitor is missing, so such values are interpolated based on nearby time-stamps. All other rows which has NaN data has been either dropped or interpolated to the value of similar entry.

We have two different questions in which the goal is to predict either the subject under the given condition of a known activity or an activity under the given condition of a known subject. For those problems we created two different types of data files from original complete dataset. For the question of predicting subject for given activity, we need data files for each activity which will have all data columns except activityId and will have an extra column of subjectId which will be treated as target label in this case. For the question of predicting activity for given subject, we need data files for each subject which will have all data columns and in this case activityId column will be treated as target label.

There is one very common issue in HAR. Human beings performs activity fluently and consecutive activities are combined with other rather than being clearly separated by pause. In many cases, the exact boundary of the activity is very hard to define. For example, a drinking activity will start with reaching for the cup or holding the cup ends after seeping or keeping cup on the table. Human beings performs all these activities very smoothly with a flow and in this case, it is very difficult to separate one activity from another with clear boundary. Hence to resolve this type of problem, we need to segment the dataset. There exists many processes to resolve the problem of segmentation. In our case, we have implemented a sliding window approach to segment the data. We created windows of 5.12 seconds over the data with the stride of 1 second, which means there is almost 4 seconds overlap between data from two consecutive windows. In such a way, we segmented data to resolve the issue of unclear boundary between two activities. This will increase the number of features by the factor of 512, with the total number of features reaching 15k.

Another issue with the dataset is that it is not balanced for all activities. For few activities it has more data while some activities has less amount of data. Hence, so tackle this problem we have calculated the class weights which are considered during model training.

B. MODELS AND METRICS

After preprocessing, we normalized the data and fit into different models. We have implemented the Following Models-Logistic Regression, Decision Trees, Naive Bayes, Boosted Trees and SVM. Here we discuss some of the key parameters that were tuned to get the best results for the third question. All the models were implemented in python language using Scikit-Learn package.

1) LOGISTIC REGRESSION: Logistic Regression is one of the simplest Linear Models used for classification purposes. It essentially creates a linear boundary in an n-dimensional space, and classifies according to that boundary. We choose 'liblinear'[9] solver and L2 Penalty Loss to find the decision boundary. We apply a maximum iteration of 400 after we didn't see any substantial improvement in the accuracy past that. Training and Testing Accuracies were comparable, suggesting against overfitting. Thus no regularization has been applied.

2) DECISION TREES: Decision Trees are a non-parametric machine learning algorithm where we divide the data into two parts based on the most dominant features, and in the process of which we form a tree, based on which decisions can be made about the target. We have implemented Scikit Learn's[9] Classification and Regression Trees, which is very similar to C4.5 Decision Trees [10] implemented by [11]. We have chosen Gini Impurity as the impurity measure. If y has values ranging from 0 to k-1 for node m, p_{mk} denotes the proportion of class 'k' observations in node m.

$$P_{mk} = 1/N_m \sum_{x_i \in R_m} I(y_i = k)$$

Gini Impurity is calculated in the following way:

$$H(X_m) = \sum p_{mk}(1 - p_{mk})$$

3) GAUSSIAN NAIVE BAYES: Naive-bayes is a probabilistic classifier that makes classifications using the Maximum A Posteriori decision rule in a Bayesian setting. It can also be represented using a very simple Bayesian network. The Naive Bayes Model assumes conditional independence between all the features and computes the probabilities for different values of target based on this assumption. i.e. For n features $x_1, x_2, ...x_n$,

$$P(x_i|y, x_1, x_2...x_i - 1, x_i + 1, ...x_n) = P(x_i|y)$$

Gaussian Naive Bayes assumes the likelihood of the features to be Gaussian. This model is the fastest to run, is less prone to overfitting, and and gives decent results in classification problems.

4) SUPPORT VECTOR MACHINES: A Support Vector Machine (SVM) is a discriminative classifier formally defined by a separating hyperplane. For multiclass classification, we use kernels. The polynomial kernel can be written as:

$$K(x, xi) = 1 + sum(x * x_i)^d$$

and exponential as

$$K(x, xi) = exp(-gamma * sum((x - x_i^2))$$

Polynomial and exponential kernels calculates separation line in higher dimension. This is called kernel trick. Kernel trick is the method of using linear classifier to solve non-linear problems. The kernel function is what is applied on each data instance to map the original non-linear observations into a higher-dimensional space in which they become separable. A margin is a separation of line to the closest class points. SVM tries to achieve a good margin. A good margin is one where this separation is larger for both the classes.

We have implemented the SVM model and scaled values of the features are used as gamma value. RBF Kernel was used with specified class weights.

Model	Accuracy
Logistic Regression	83.96%
SVM	91.34%
Decision Trees	89.96%
K Nearest Neighbors	93.83%
Nave Bayes	94.92%
Light GBM	97.13%

TABLE I
PREDICT ACTIVITY GIVEN SUBJECT

Model	Accuracy
Logistic Regression	74.85%
SVM	84.01%
Decision Trees	76.23%
K Nearest Neighbors	86.09%
Nave Bayes	89.81%
Light GBM	94.74%

TABLE II
PREDICT SUBJECT GIVEN ACTIVITY

5) BOOSTED-TREE(LIGHT GBM): Lightgbm [6] is a gradient boosted framework that uses gradient based boosting tree learning algorithm. It uses Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB). With GOSS, a lot of the data with small gradients are excluded during information gain.They also bundle mutually exclusive features to reduce the number of features. Although finding the optimal feature binding is NP-Hard, Light GBM works fairly well with a greedy approach. One drawback of Light GBM is, it is susceptible to over-fitting for small dataset. As our dataset is large enough, we are getting best accuracy in Light GBM. We have tuned the parameters of this algorithm separately for the different questions.

C. PROBLEMS FACED & SOLUTIONS

Segmentation of the data using sliding window resulted in increase of number of features by a factor of 512, since we were taking a window of 5.12 seconds, a stride of 100ms, with 100Hz as the sample frequency. This had adverse effects on the training, with frequent crashes of the system, with 12 GB of memory in Google Collaborator not being enough. The first solution we thought of was to down-sample the data by a factor of 10, by averaging out 10 rows of data to one single row. As it turned out, this had quite a negative impact in one of the experimental designs, so we decided against it. The solution was to increase the stride to 1 seconds, so the number of rows decreased, allowing us to train our models without any system issues.

V. EXPERIMENTS & EVALUATION OF MODELS

We now evaluate our models for all the three questions.

A. PREDICT ACTIVITY GIVEN SUBJECT

The experimental setup of this question is: given 5 seconds of data of a subject, what is the activity? The model was trained on only 1 subject, and tested on the single subject.

Model	Without Human Features	With Human Features
Decision Tree	84.76%	86.69%
SVM	91.49%	90.62%
Logistic Regression	75.48%	74.62%
Nave Bayes	91.26%	90.57%
Boosted Tree	97.55%	98.24%
KNN	93.76%	94.23%

TABLE III
PREDICT ACTIVITY: STANDARD CROSS VALIDATION

The results of this are expected to be high, since human beings tend to perform the same activities in the same manner. We implemented various machine learning algorithnms on the data. The testing was done by cross-validating with a 75-25 Train-Test Split. Each model was run for each of the subjects, and the average results of the experiment are shown in Table I. We are using accuracy as a metric for our experiment. We can see here that Boosted Tree Model performs the best between all the models implemented here.

B. PREDICT SUBJECT GIVEN ACTIVITY

In this setup, our goal is to predict the subject for a given activity. The motivation for this question is to check the correlation of a subject with the activity. We have dropped the 'Temperature' data present in each IMU, since the subjects might be gathering data in different temperatures, and it would affect our insight to the correlation between a subject and the activity. As per general assumptions, activities are closely correlated with the subjects, and that's why we get pretty high accuracy numbers for this experiment. The testing was done by cross-validating with a 75-25 Train-Test Split. Each model was run on each activity. The results are in Table II. We can see that the Boosted Tree Model outperforms all other models in this question.

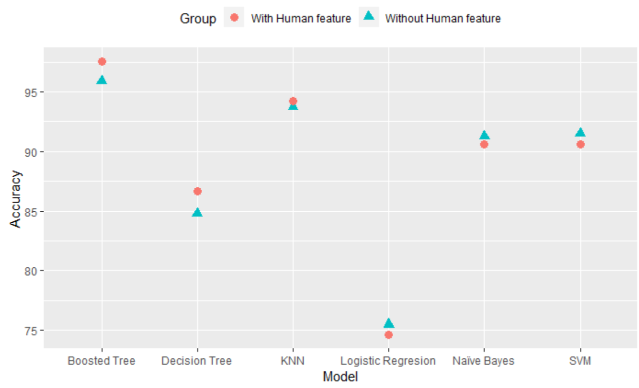

Fig. 1. Cross Validation: Accuracies

C. PREDICT ACTIVITY WITHOUT SUBJECT

The main goal of this project is to find the activity without any prior information about the subject, i.e. to build a model which is independent of the subject and can be applied

Model	Average		Std. Deviation	
	Without Human Features	With Human Features	Without Human Features	With Human Features
Decision Tree	57.88%	57.22%	12.76%	12.47%
SVM	72.13%	72.76%	16.73%	17.26%
Logistic Regresion	64.63%	64.61%	15.29%	15.26%
Nave Bayes	82.88%	82.78%	14.69%	14.75%
Boosted Tree	82.63%	80.92%	12.00%	11.99%
KNN	73.19%	73.23%	16.50%	16.48%

TABLE IV
PREDICT ACTIVITY:LOSO VALIDATION

Fig. 2. LOSO: Accuracies

for any new user. This problem is difficult, considering the correlation of a subject with the way it performs an activity, as we saw in the last question. We tried adding Human Features, like Height, Weight, Resting Heart Rate, Dominant Arm, etc. to the features to try and improve the models. Evaluation was done in two ways-Standard Cross Validation and 'Leave One Subject Out'(LOSO). In Standard Cross Validation, the entire data of all subjects is split into a 75-25 train-test cross validation split, and evaluated on that. The results of this experiement are illustrated in Figure 1 and the actual results are present in Table III. This is a big step toward subject independence. But,there is still some dependence on the subject, since the training and testing are done on the same subjects. Thus to truly test for subject-independence, we need to do a LOSO validation. In this design, out of all the subjects, training is done on every subject bar one, and testing is done on the one remaining subject. The results of this experiment are illustrated in Figure 2, and the mean accuracies and standard deviations are present in Table III. We can see that Boosted Trees has the best performance out of all the models in Cross Validation, while Naive Bayes Model has the best performance in the LOSO evaluation. We also notice here that adding human feature did not necessarily add any value to the model, and in a lot of cases the difference is negligible, or negative. Comparing our results with [11] , we see that for some algorithms, our models come close to their accuracies.

VI. CONCLUSION AND FUTURE WORK

In this paper, we have tried multiple machine learning algorithms to identify 12 different activities based on data collected from 3 IMUs. Unlike previous methods like [4] where data from each window is transformed, our method takes into account the entire data. We also take into account features intrinsic to the human and experiment our models accordingly .Our idea is human performs activities fluently and it is very hard to predict clear boundary of finishing of one activity and starting of next. We have implemented this idea by segmenting data using sliding window. We have used the dataset which is collected by using 3 different IMUs fitted at wrist, dominating ankle and chest and achieved comparable accuracy. The LOSO testing condition finds out how generalized our system is and how it will perform for any unknown user.

There are multiple new ideas as well which can be implemented in future like using deep network to predict activity for any random user. We can expand this for many other activities as well and try to get more accurate result.

REFERENCES

[1] Ulf Blanke Andreas Bulling and Bernt Schiele. "A Tutorial on Human Activity Recognition Using Body-Worn Inertial Sensors". In: *ACM Comput. Surv. 2014* (2014). DOI: 10.1145/2499621.

[2] Daniel Ashbrook and Thad Starner. "MAGIC: A motion gesture design tool". In: 2010, pp. 2159–2168.

[3] Manas Mittal Bjorn Hartmann and Scott R Klemmer. "Authoring sensor-based interactions by demonstration with direct manipulation and pattern recognition". In: 2007, pp. 145–154.

[4] Pierluigi Casale, Oriol Pujol, and Petia Radeva. "Human Activity Recognition from Accelerometer Data Using a Wearable Device". In: *Pattern Recognition and Image Analysis*. Ed. by Jordi Vitrià, João Miguel Sanches, and Mario Hernández. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 289–296. ISBN: 978-3-642-21257-4.

[5] C Ponce J Sung and A Saxena. "Human activity detection from RGBD images". In: 2011.

[6] Guolin Ke et al. "Lightgbm: A highly efficient gradient boosting decision tree". In: *Advances in Neural Information Processing Systems*. 2017, pp. 3146–3154.

[7] Jonathan Lester, Tanzeem Choudhury, and Gaetano Borriello. “A practical approach to recognizing physical activities”. In: *International conference on pervasive computing*. Springer. 2006, pp. 1–16.

[8] Andrea Mannini and Angelo Maria Sabatini. “Machine learning methods for classifying human physical activity from on-body accelerometers”. In: *Sensors* 10.2 (2010), pp. 1154–1175.

[9] F. Pedregosa et al. “Scikit-learn: Machine Learning in Python”. In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[10] J. Ross Quinlan. *C4.5: Programs for Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993. ISBN: 1558602402.

[11] Attila Reiss and Didier Stricker. “Creating and Benchmarking a New Dataset for Physical Activity Monitoring”. In: *Proceedings of the 5th International Conference on Pervasive Technologies Related to Assistive Environments*. PETRA ’12. Heraklion, Crete, Greece: ACM, 2012, 40:1–40:8. ISBN: 978-1-4503-1300-1. DOI: 10.1145/2413097.2413148. URL: http://doi.acm.org/10.1145/2413097.2413148.