Tackling the SHL Challenge 2020 with Person-specific Classifiers and Semi-supervised Learning

Stefan Kalabakov
stefan.kalabakov@ijs.si
Jožef Stefan Institute
Ljubljana, Slovenia

Simon Stankoski
simon.stankoski@ijs.si
Jožef Stefan Institute
Ljubljana, Slovenia

Nina Reščič
nina.rescic@ijs.si
Jožef Stefan Institute
Ljubljana, Slovenia

Ivana Kiprijanovska
ivana.kiprijanovska@ijs.si
Jožef Stefan Institute
Ljubljana, Slovenia

Andrejaana Andova
andrejaana.andova@ijs.si
Jožef Stefan Institute
Ljubljana, Slovenia

Clement Picard
clement.picard@ens-rennes.fr
ENS Rennes
Rennes, France

Vito Janko
Jožef Stefan Institute
Ljubljana, Slovenia

Martin Gjoreski
Jožef Stefan Institute
Ljubljana, Slovenia

Mitja Luštrek
Jožef Stefan Institute
Ljubljana, Slovenia

ABSTRACT

The Sussex-Huawei Locomotion Challenge 2020 was an open competition in activity recognition where the participants were tasked with recognizing eight different modes of locomotion and transportation with smartphone sensors. The main challenges were that the training data was recorded by a different person than the validation and test data, and that the smartphone location in the test data was unknown to the participants. We tackled the first challenge by attempting to identify the persons with clustering, and then performed cluster/person-specific feature selection to build a separate classifier for each person. The smartphone location appears not to make much difference. We also used semi-supervised learning to classify the test data. Internal tests using this methodology yielded an accuracy of 81.01%.

CCS CONCEPTS

• **Computing methodologies** → **Supervised learning**; *Semi-supervised learning*; *Unsupervised learning*.

KEYWORDS

Activity recognition, machine learning, feature extraction, competition, smartphone, semi-supervised learning, clustering

ACM Reference Format:

Stefan Kalabakov, Simon Stankoski, Nina Reščič, Ivana Kiprijanovska, Andrejaana Andova, Clement Picard, Vito Janko, Martin Gjoreski, and Mitja Luštrek. 2020. Tackling the SHL Challenge 2020 with Person-specific Classifiers and Semi-supervised Learning. In *Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and the 2020 International Symposium on Wearable Computers (UbiComp/ISWC '20 Adjunct), September 12–16, 2020, Virtual conference.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

UbiComp/ISWC '20 Adjunct, September 12–16, 2020, Virtual conference

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Smartphones, smart watches and other wearables have become ubiquitous. By analyzing sensor data acquired via such devices, one can develop applications and services that contribute to safety, health, comfort and overall quality of life. For this reason, activity recognition with wearable devices is a research topic studied by many researchers.

The Sussex-Huawei Locomotion (SHL) Challenge is an activity recognition challenge that has for three years in a row presented different challenges for recognizing eight modes of locomotion and transportation from the inertial sensor data of a smartphone. In 2020, the task was to recognise modes of transportation in a user-independent manner with an unknown phone position. The test data was composed of data from two users that were not included in the train data, and only a little bit of validation data from these two users was given. The location of the phone for the users in the test data was also not given, although only one location was used.

So far, many approaches for activity recognition (AR) with wearable devices have been developed. A classical approach to detect activities using sensor data would be to either use machine learning [3][9], or deep learning [11]. A more challenging approach would be to use semi-supervised learning [8][10]. Some authors have even tried using unlabeled sensor data to recognise human activities [8][7].

However, the main problem in this challenge is that the phone location and the users in the test data were unknown. Thus, a major part of the challenge was to figure out which phone location did the users in the test data use, and which sample in the test data belonged to which user. To identify the phone location, we used classical machine learning methods, which showed that the users in the test data were holding the phone on their hips. To identify to which user the samples from the test set belonged we used unsupervised clustering method.

Table 1: Summary of the provided datasets. "X" marks if the data is available for a particular user or if the data includes the activity label.

	User 1	Users 2&3	Locations	Labels	Days
SHL-Training	X		4	X	59
SHL-Validation		X	4	X	6
SHL-Test		X	1 (unknown)		40

2 SHL CHALLENGE DATA

The goal of 2020 SHL challenge was to recognize eight modes of locomotion and transportation (car, bus, train, subway, walk, run, bike and still) by using inertial sensor data of a smartphone. The data was originally recorded using four smartphones worn at different on-body locations (hips, torso, bag, hand) by three different users; however only a subset of all the data was provided by the challenge organizers.

The provided data came in three sets - *Train*, *Validation*, *Test*. To distinguish original sets from train, validation and test sets in specific experiments we use the *SHL* prefix. *SHL - Train* was the largest and it contained data from one user (user 1) and all four phone locations. The *SHL-Validation* set was much smaller and it contained mixed data from the other two users (user 2 and user 3) again for all four locations. Finally, the *SHL-Test* only contained data from users 2 and 3 and one unknown phone location. This set was unlabeled - correctly labeling it was the competition's goal. Overall, the challenge data comprised of 4 x 59 days of *SHL-Training* data (59 days of data for each of the four locations), 4 x 6 days of *SHL-Validation* data and 40 days of *SHL-Test* data – as summarized in Table 1.

The raw sensor data was sampled at a frequency of 100 Hz and it included data from the following sensors: acceleration (x, y and z axis), gravity (x, y and z), gyroscope (x, y and z), linear acceleration (x, y and z), magnetic field (x, y and z), orientation (x, y, z and w) and pressure. GPS, WiFi and other sensor data that could be used to identify the location of the user was omitted. The data was segmented using 5-second windows while the labels were provided per-sample.

The distribution of the activities for the *SHL-Training* and *SHL-Validation* data was very uniform, except for running activity, which was understandably under-represented.

3 PRE-PROCESSING AND FEATURES

In this work we opted for the use of classical machine learning, as it yielded better results then the deep learning. In order to employ it, we pre-processed the data and then calculated a large body of features from it.

3.1 Data ordering and split

Therefore, the unshuffling algorithm developed and used in the two previous SHL competitions [5][4] was not useful for classifying the *SHL-Test* data, but was still used to smooth the unsupervised clustering on *SHL-Validation* data.

3.2 Deriving data streams

SHL dataset contains 20 (counting all the axis) different sensor streams. From these original sensor streams it is possible to derive additional sensor streams that are useful for the AR. The subsequent steps treat these derived sensor streams like any of the original ones.

First derived sensor stream is the magnitude of the data. It was calculated for all the data that is coming from tree-axis sensors (acceleration, linear acceleration, gravity, magnetic field and angular velocity). Additional derived sensor streams were Euler angles, derived from quarternion data. Quarternions are better to avoid the "gimbal-lock", but Euler angles give better information on body's orientation.

3.3 Features

In order to use classical machine learning we calculated features on each five-second window of data – this window size was the largest possible, given the competition limitations, and our previous experience [5][4] on a similar problem showed that larger windows outperform the smaller ones, presumably due to infrequent activity transitions. Labels were calculated for each window as the most frequent per-sample label in that window.

Calculated features can be roughly categorized as being frequency-domain, time-domain and those calculated using the *tsfresh* package. The following three subsections describe each category respectively. Altogether 1124 features were calculated.

3.3.1 Frequency-domain features. These features were calculated using the power spectral density (PSD) of the signal, based on the fast Fourier transform (FFT). PSD characterizes the frequency content of a given signal and can be estimated using several techniques. Two of the most widely used and commonly considered are a simple periodogram, which is obtained by taking the squared-magnitude of the FFT components and the Welch's method, which is a bit more complex but also superior to periodogram.

In our work we used the Welch's method to obtain the PSD. We have implemented the same frequency-domain features as in the previous competitions [5][4] – three largest magnitudes of the FFT components, entropy of the the normalized FFT components and their energy.

3.3.2 Time-domain features. We have used time-domain features, that have proven themselves in our previous work [1][2] and previously won competitions [5][6][4]. These features were designed for accelerometer data and most of them were calculated only on the acceleration (and its derived) data streams. Some of the features were also calculated on the gyroscope data streams, however, some features such as *linear velocity* were left out as they have no semantic interpretation when calculated on non-acceleration data.

3.3.3 TSfresh features. We also extracted a subset of time-domain features from the tsfresh library that were not included in the previous set of expert features. These features were: the signal minimum, maximum, standard deviation, the number of times the signal is above/below its mean, the signal's mean change/absolute change, and its different autocorrelations (correlations of the signal with a delayed version of itself, for three different delays). All these features were designed for the accelerometer and gyroscope data (and their derived data streams).

4 METHOD

The main difference and at the same time the main challenge of this competition compared to the previous years is that we were required to recognise the mode of transportation in a user-independent model using data from one unknown location. To increase the chance for recognition of transportation modes on *SHL-Test* set, we tried to recognize the unknown location and additionally to separate the *SHL-Validation* and *SHL-Test* test into two clusters, which would hopefully indicate two users.

4.1 Recognizing the unknown location

We explored whether it is possible build a classifier for phone location detection, which can be used for investigating the advantage of location-dependent activity recognition models.

The sensor signals recorded while performing dynamic activities, such as walking and running, are highly dependent on the location of the sensors on the body, mainly because different body parts vary in degrees of freedom and have different movement patterns. On the other hand, the signals recorded while a person is using one of the different transportation modes (train, bus, subway) are very similar to each other, regardless on the location of the sensors on the body. With this consideration, we performed the detection of the phone location in two steps: i) walking/running detection step; ii) location detection step. In the walking/running detection step, we used data from all activities and all four locations (bag, hand, hips and torso) for the training of the classification model. The task was formulated as binary classification - all walking and running instances were labeled as class 1, and the others were labeled as class 2. In the location detection step, only instances referring to the walking and running activity were used to train the location classifier.

The same feature set was used in the activity recognition and location detection step, i.e. we used all extracted features (a total of 1,124 features) for both tasks. Random Forest was used for both walking/running identification and phone location detection.

4.2 Person clustering

We decided to explore the possibility of clustering the *SHL-Validation* and *SHL-Test* sets in order to separate the two users in both sets. This would allow us to build user-specific models and hopefully increase the performance of our model.

First we tried to cluster *SHL-Validation* data using the K-means algorithm on all possible features. Only a few consecutive samples were put in the same cluster, which suggested that the clusters did not correspond to persons. Additionally, clustering data from different locations returned different clusters.

Next we tried to determine which features could be used for distinguishing between different persons. We used 25% of *SHL-Train* set (containing user 1) and the whole *SHL-Validation* set and tried to separate user 1 from users 2 and 3. We ranked the features for each location and selected the first 50 most important features for each location. We tried to cluster the data using only the selected features on each location. Clustering worked much better and the clusters were almost identical on all locations, except the torso.

Finally we smoothed out the clusters, so that all the consecutive samples belonged to the same cluster, by using the unshuffling algorithm we developed at previous two SHL competitions [5][4]. Using the smoothed labels, we built a classification model to be used on the unlabeled *SHL-Test* set.

4.3 Feature selection

Since we computed a large number of features, we selected the most relevant ones with a three-step procedure. In the first step, the mutual information between each feature and the label was estimated, where larger mutual information means a higher dependency between the feature and the label. In the second step, Pearson correlation coefficient was computed for pairs of features. If the correlation was higher than a threshold, the feature with the lower mutual information with the label was discarded. In the final step, features were selected using a greedy wrapper approach. A Random Forest classifier was first trained using only the best scoring feature on the training set. The trained model was used to predict labels for the validation set and the prediction accuracy was calculated. Then the second-best feature was added and the model was trained again. If the accuracy on the validation set was higher than without using this feature, the feature was kept. This procedure was repeated for all the remaining features.

Feature selection is typically used to select generally good features, but in our case we also used it to adapt the features to particular users. We first adapted the features to users 2 and 3 combined. Two procedures were used. The first procedure used the *SHL-Training* set (user 1) for training and *SHL-Validation* set (users 2 and 3) for validation. The second procedure used one half of the *SHL-Validation* set for training and the other half for validation, and repeated this with the halves switched. We used the intersection of both selections as the output of the second procedure. Finally, we used the union of the outputs of both procedures as the final feature set for users 2 and 3 combined.

We also selected features specific for the two clusters obtained with person clustering (hopefully representing two different persons – user 2 and user 3). The *SHL-Validation* was split into two clusters/users as described in the previous section. Afterwards, the feature selection was done in the same manner as for users 2 and 3 combined: the two procedures were used on each cluster/user, and their union was the final feature set for that user.

4.4 Semi-supervised learning

As previously mentioned, the unlabeled *SHL-Test* data is entirely comprised of readings from user 2 and user 3, whilst their presence in the labeled *SHL-Train* and *SHL-Validation* sets is significantly smaller. To deal with the small number of examples from user 2 and user 3, we opted to use a semi-supervised learning scheme in order to leverage the knowledge contained in the unlabeled data. This process is independent of the use of clustering, as the labeled and unlabeled data could represent data from one user or a mixture of several different users.

In our semi-supervised approach, we first trained a classifier on the labeled data and we used that classifier to predict the unlabeled data. Once the predictions were obtained, we selected which instances from the unlabeled data would be transferred to the labeled training set and used to train another classifier. The selection process in our approach differed based on whether we used a single classifier in the prediction process or an ensemble of classifiers joined by a single majority voting classifier. In the case of a single classifier, the selection of an instance was done by comparing the prediction probability of the classifier for the given instance to a threshold value. On the other hand, when using a majority voting classifier, the selection was done by comparing the number of base classifiers which voted in the same way to a threshold value. A stricter version of this selection was also used in some experiments, where, for an instance to be selected, all base classifiers had to vote the same way and they all had to have a prediction probability higher than some threshold value.

It is important to mention that the unlabeled instances were transferred to the labeled training set by utilizing the predictions which our classifier produced. Finally, the process of training a classifier, predicting the unlabeled data, transferring instances back in the labeled training set and retraining the classifier can be repeated several times (from here on we refer to these repetitions as iterations) until convergence is achieved. The whole process is represented in Figure 1.

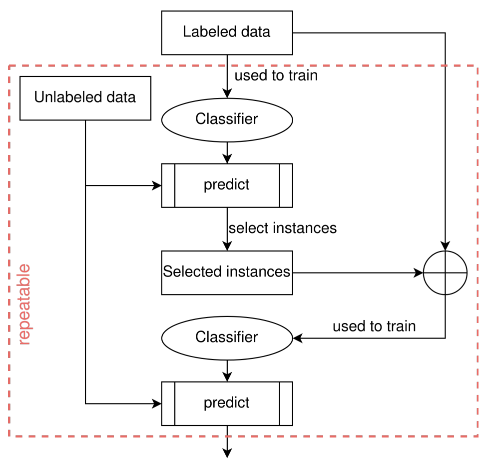

Figure 1: Semi-supervised learning scheme

4.5 Proposed pipeline

Our final approach is a combination of the methods previously described in this section and is shown in Figure 2. The pipeline starts with the creation of a labeled training set which consists of the data in the *SHL-Train* set (user 1) and the data from the *SHL-Validation* set (user 2 and user 3), regardless of the sensor location. Next, we divide the unlabeled *SHL-Test* data into two distinct clusters, which hopefully represent two different users. Once divided, we reduce the dimensionality of each cluster by using only those features that showed best results in the feature selection stage, for each cluster respectively. In addition to this we also create two versions of the labeled data, once with features selected for one of the clusters and once for the other. From there, we create a XGBoost classifier of each of the clusters and we start the process of semi-supervised learning, as described in Section 4.4, with just one iteration. The selection criterion in the semi-supervised learning stage is that the classifier predicts a label for the instance with a prediction probability greater than 70%.

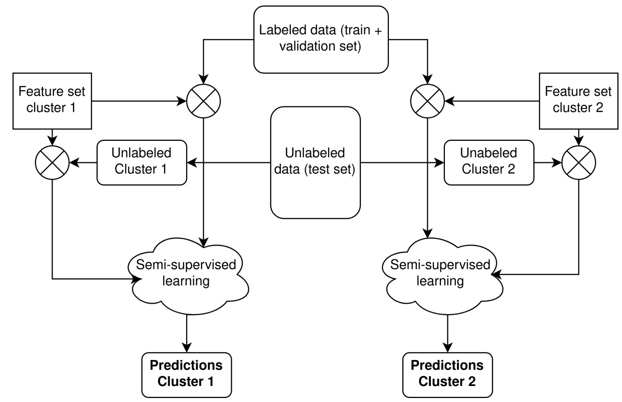

Figure 2: Proposed pipeline.

5 EXPERIMENTAL RESULTS

5.1 Recognition of unknown location

We evaluated the performance of the location detection on the validation set by performing 4-fold cross-validation, where each fold is exactly 1/4 of the validation set. The main reason for using only the validation set is that it provides more significant information about the position of the phones since it contains data from the same users as the test set.

The first step towards phone location detection was the identification of walking/running instances. The walking/running recognition showed an accuracy of 95.5%. However, for the purpose of phone location detection, the overall accuracy in this step was not the main concern. Instead, we were more interested in the number of false positives (instances falsely classified as walking/running activity), which might affect the performance of the location classifier. In total, roughly 2% of all instances from the validation set were falsely classified as walking/running activity. We assumed that this number is insignificant and would not affect the location detection process, considering the fact that nearly 90% of the instances identified as walking/running activity were true positives.

Afterwards, we tried to identify the location only on the instances that were classified as walking/running in the previous step. This includes the true positives (20,391 instances) and the false positives (2,532 instances). The highest location accuracy was noted for the hand location – 98.2% of the instances referring to this location were truly assigned to this location. The accuracy for the other locations was as following: 76.1% for the bag location, 74.2% for the hips location, and 84.2% for the torso location. These results encouraged us that it is possible to achieve a high classification accuracy in phone location detection, so we proceeded with phone location detection on the test set.

Eventually, we trained a final model for walking/running activity detection using the whole validation set and used it to classify the

Table 2: Location distribution on the test set.

	Bag	Hand	Hips	Torso
No. of instances	95	2,430	12,901	38

Table 3: F1-score for each location when using different training sets (either all data, or data from that location).

	Bag	Hand	Hips	Torso
Train on all locations	0.76	0.70	0.64	0.75
Train on specific location	0.74	0.66	0.64	0.67

instances from the SHL test set. The final results for the test set regarding the location detection are presented in Table 2. In total, 15,464 instances were classified as either walking or running activity. Later, 83.4% of them were classified as hips location, 15.7% as hand location, while less than 1% of the instances were classified as bag and torso location. Based on these results, we assumed that the unknown phone location of the test set is hips.

With this information, we further explored if location-dependent model might bring us higher classification accuracy than a general, location-independent model. More specifically, we investigated the influence of including only data from a specific location in the model's training set and including data from all four positions on the accuracy of the models, for each position separately. These experiments were also evaluated using a 4-fold cross-validation scheme on the *SHL-Validation* data. In each iteration, the model was trained using the *SHL-train* set + 3 folds of the *SHL-Validation* data, and evaluated on the remaining fold. The results from these experiments are presented in Table 3.

These results show that there was no advantage in training location-specific models using only data from the same location. The results for the hips location were the same with both setups, while for the other locations the accuracy decreased when using data only from the specific location. Therefore, we proceeded with location-independent models.

5.2 Person clustering

We performed the clustering as described in section 4.2 and shown in Figure 3.

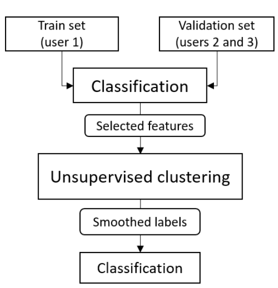

Figure 3: Clustering pipline

The classification problem of separating *SHL-train* set from *SHL-Validation* set was very straightforward. We used the Random Forest classifier and its integrated attribute to rank features by importance. We selected 50 most important features for each location. These features were then used to cluster the *SHL-Validation* set. We found two quiet clear clusters (silhouette score 0.58). To make sure that clustering didn't just cluster the activities, we run the clustering algorithms on each of the activities subsets and compared them with the clusters we got when running the clustering algorithm on entire *SHL-Validation* set.The clusters matched. On three locations (hand, hips and bag) clusters were very similar, as seen in the Figure 4. Clustering did not work as well on the remaining location (torso), but as we have predicted with a very high confidence that this is not the unknown test location.

We then smoothed the clusters by making sure that all of the consecutive samples were in the same cluster. The smoothed clusters were used to build a classifier (Random Forest) which predicted the clusters for the *SHL-Test* data. To check the "validity" we also tried to use the previously described unsupervised clustering on the *SHL-Test* data and compared it with the predicted clusters. There was a 72% match.

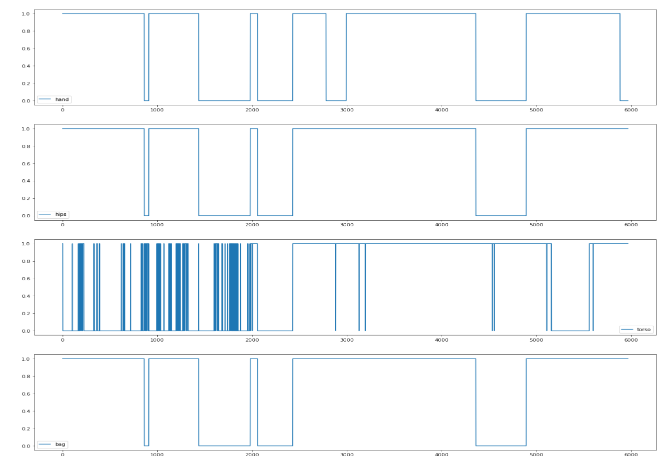

Figure 4: Example of clustered data on all four locations (from top down: hand, hips, torso and bag) for walking. The clustering works very similar for all other activities.

5.3 Proposed pipeline

In order to determine the effectiveness of using user-specific (cluster-specific) models which use semi-supervised learning, two different evaluation scenarios were devised. The first focused on obtaining an evaluation of a user-independent model, and the second on the evaluation of a user-dependent one. Since the results in Table 3 suggested that training a location-specific model worsens the results, both evaluation scenarios used examples from all four sensor locations.

The *user-independent models* were evaluated using a 4-fold cross-validation scheme on the *SHL-Validation* data, where each fold contained exactly 1/4 of the instances. In each of the iterations, a model was trained using the *SHL-Train* set + 3 folds of the *SHL-Validation* data (referred to as the training set) and evaluated on the fold which was left out when training. During the training and

Table 4: Evaluation results of different user-dependent and user-independent models

	XGB	MV	XGB + SSL	MV + SSL
User-independent	73.05	74.63	N/A	76.56
User-dependent	79.18	77.64	81.01	79.02

Table 5: Evaluation results of a XGBoost classifier with and without using feature selection

	XGB-noFS	XGB-withFS
User-independent	71.2	73.05
User-dependent	77.62	79.18

evaluation of these models, only the features selected as best on the whole validation set were used.

In the experiments which involved semi-supervised learning, the data in the fold which was left out during training was treated as unlabeled and instances from it were transferred back in the training set. The results, averaged across folds, of this evaluation can be found in Table 4.

The *user-specific models* were evaluated using a slight variation of the aforementioned 4-fold cross-validation scheme. This variation first split the *SHL-Validation* set into two clusters and further split each of those clusters into two halves. This way we ended up with four different subsets of data (which do not have the same number of instances) which were then used in the 4-fold cross-validation.

All the other aspects of the evaluation scheme are the same with the previously described one, except for the fact that when training and evaluating these models, we use the feature set selected for the cluster to which the left out fold belongs. The results, averaged across folds, of this evaluation can also be found in Table 4.

The use of feature selection in both of these evaluation scenarios is supported by the results in Table 5.

The first two columns of Table 4 serve as baseline and show results from training a single XGBoost classifier (XGB) and a majority voting classifier (MV) in both a user-dependant and independent fashion, without using semi-supervised learning. The next two columns represent the best results that both XGBoost and the majority voting classifier achieved when using semi-supervised learning (SSL).

In the case of the user-independent scenario, the best result is achieved by MV + SSL. In this case the selection strategy is based on all classifiers voting in the same way. One iteration was used in the SSL process. The results for XGB + SSL were not computed as we had no indication at that time that a single classifier might outperform majority voting.

Finally, in the case of the user-dependent models the best results were achieved by XGB + SSL. In this case the SSL process was repeated once and the selection strategy was based on the prediction probability being above 70%. The best score achieved by MV + SSL, used one iteration of the SSL process and a selection strategy based on all classifiers voting the same way.

6 CONCLUSION

The main idea behind the Sussex-Huawei Locomotion Challenge 2020 was to identify eight modes of transportation and locomotion using data recorded on sensors with an undisclosed location on the user's body. Another difficulty was that the subjects in the test set had an extremely low presence in the training and validation data.

Using machine learning, we were able to determine the actual location of the test data (hips). However, our experiments showed that using location-specific models did not help in improving the accuracy of the classification task. On the other hand, using clustering to identify different users in the validation and test sets helped us build user-specific models which greatly improved our classification ability. In particular, one use of the clustering was to select features for particular users, which reduced the computation complexity of our models and showed improved results in comparison to user-independent features. Also, the feature selection step helped us to adapt the features from user 1 to user 2 and 3. This step enabled us to use the whole training dataset (user 1) for training the final model, without worrying that we will overfit to that user. Finally, we discovered that using semi-supervised learning to train our models extracted valuable knowledge from the unlabeled data, which helped us tackle the problem of the small number of labeled instances for the users in the test set.

REFERENCES

[1] Božidara Cvetković, Vito Janko, and Mitja Luštrek. 2015. Demo abstract: Activity recognition and human energy expenditure estimation with a smartphone. In *2015 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*. IEEE, 193–195.

[2] Božidara Cvetković, Robert Szeklicki, Vito Janko, Przemyslaw Lutomski, and Mitja Luštrek. 2018. Real-time activity monitoring with a wristband and a smartphone. *Information Fusion* 43 (2018), 77–93.

[3] Wei-Chih Hung, Fan Shen, Yi-Leh Wu, Maw-Kae Hor, and Cheng-Yuan Tang. 2014. Activity recognition with sensors on mobile devices. In *2014 International Conference on Machine Learning and Cybernetics*, Vol. 2. IEEE, 449–454.

[4] Vito Janko, Martin Gjoreski, Carlo Maria De Masi, Nina Reščič, Mitja Luštrek, and Matjaž Gams. 2019. Cross-Location Transfer Learning for the Sussex-Huawei Locomotion Recognition Challenge. In *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers* (London, United Kingdom) *(UbiComp/ISWC '19 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 730–735. https://doi.org/10.1145/3341162.3344856

[5] Vito Janko, Nina Reščič, Miha Mlakar, Vid Drobnič, Matjaž Gams, Gašper Slapničar, Martin Gjoreski, Jani Bizjak, Matej Marinko, and Mitja Luštrek. 2018. A New Frontier for Activity Recognition: The Sussex-Huawei Locomotion Challenge. In *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*. ACM, 1511–1520.

[6] Simon Kozina, Hristijan Gjoreski, Matjaž Gams, and Mitja Luštrek. 2013. Efficient activity recognition and fall detection using accelerometers. In *International Competition on Evaluating AAL Systems through Competitive Benchmarking*. Springer, 13–23.

[7] Yongjin Kwon, Kyuchang Kang, and Changseok Bae. 2014. Unsupervised learning for human activity recognition using smartphone sensors. *Expert Systems with Applications* 41, 14 (2014), 6067–6074.

[8] Yonggang Lu, Ye Wei, Li Liu, Jun Zhong, Letian Sun, and Ye Liu. 2017. Towards unsupervised physical activity recognition using smartphone accelerometers. *Multimedia Tools and Applications* 76, 8 (2017), 10701–10719.

[9] Charissa Ann Ronao and Sung-Bae Cho. 2016. Human activity recognition with smartphone sensors using deep learning neural networks. *Expert systems with applications* 59 (2016), 235–244.

[10] Maja Stikic, Kristof Van Laerhoven, and Bernt Schiele. 2008. Exploring semi-supervised and active learning for activity recognition. In *2008 12th IEEE International Symposium on Wearable Computers*. IEEE, 81–88.

[11] Shaohua Wan, Lianyong Qi, Xiaolong Xu, Chao Tong, and Zonghua Gu. 2019. Deep learning models for real-time human activity recognition with smartphones. *Mobile Networks and Applications* (2019), 1–13.