Summary of the Sussex-Huawei Locomotion-Transportation Recognition Challenge 2020

Lin Wang
lin.wang@qmul.ac.uk
Centre for Intelligent Sensing
Queen Mary University of London, UK

Hristijan Gjoreski
hristijang@feit.ukim.edu.mk
Faculty of Electrical Engineering and Information Technologies
Ss. Cyril and Methodius University, MK

Mathias Ciliberto
m.ciliberto@sussex.ac.uk
Wearable Technologies Lab
University of Sussex, UK

Paula Lago
paula@mns.kyutech.ac.jp
Kyushu Institute of Technology, Japan

Kazuya Murao
murao@cs.ritsumei.ac.jp
College of Info. Sci. and Eng.
Ritsumeikan University, Japan

Tsuyoshi Okita
tsuyoshi.okita@gmail.com
Kyushu Institute of Technology, Japan

Daniel Roggen
daniel.roggen@ieee.org
Wearable Technologies Lab
University of Sussex, UK

ABSTRACT

In this paper we summarize the contributions of participants to the third Sussex-Huawei Locomotion-Transportation (SHL) Recognition Challenge organized at the HASCA Workshop of UbiComp/ISWC 2020. The goal of this machine learning/data science challenge is to recognize eight locomotion and transportation activities (Still, Walk, Run, Bike, Bus, Car, Train, Subway) from the inertial sensor data of a smartphone in a user-independent manner with an unknown target phone position. The training data of a "train" user is available from smartphones placed at four body positions (Hand, Torso, Bag and Hips). The testing data originates from "test" users with a smartphone placed at one, but unknown, body position. We introduce the dataset used in the challenge and the protocol of the competition. We present a meta-analysis of the contributions from 15 submissions, their approaches, the software tools used, computational cost and the achieved results. Overall, one submission achieved F1 scores above 80%, three with F1 scores between 70% and 80%, seven between 50% and 70%, and four below 50%, with a latency of maximum of 5 seconds.

CCS CONCEPTS

• **Computing methodologies** → **Activity recognition and understanding**; **Supervised learning by classification**.

KEYWORDS

Activity recognition; Deep learning; Machine learning; Mobile sensing; Transportation mode recognition

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
UbiComp/ISWC '20 Adjunct, September 12–16, 2020, Virtual Event, Mexico
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-8076-8/20/09…$15.00
https://doi.org/10.1145/3410530.3414341

ACM Reference Format:
Lin Wang, Hristijan Gjoreski, Mathias Ciliberto, Paula Lago, Kazuya Murao, Tsuyoshi Okita, and Daniel Roggen. 2020. Summary of the Sussex-Huawei Locomotion-Transportation Recognition Challenge 2020. In *Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers (UbiComp/ISWC '20 Adjunct), September 12–16, 2020, Virtual Event, Mexico.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3410530.3414341

1 INTRODUCTION

The mode of transportation or locomotion is an important contextual that enables applications such as activity and health monitoring, individual environmental impact monitoring, and intelligent service adaptation [17–25]. Several prior work looked at recognizing modes of transportation from smartphone sensors, including motion, GPS, sound, and image [26–30]. To date, most research groups assess the performance of their algorithms using their own datasets on their own recognition tasks. These tasks often differ in the sensor modalities used or in the allowed recognition latency. This makes it difficult to compare methodologies and to systematically advance research in the field.

Following on our successful 2018 and 2019 challenges [31, 32], which saw 22 and 14 submissions, respectively, we organized the third Sussex-Huawei Locomotion-Transportation (SHL) recognition challenge in the year 2020[1]. In contrast to the SHL 2018 and 2019, which focused on time-independent and placement-independent evaluation, respectively, the goal of this challenge is to recognize 8 modes of locomotion and transportation (the activities include: being still, walking, running, cycling, driving a car, being in a bus, train or subway) from the inertial sensor data of a smartphone in a user-independent manner. This paper introduces the dataset used for the challenge and the protocol for the competition, and summarizes and analyzes the achievements of the participants contributing to the challenge.

[1] http://www.shl-dataset.org/activity-recognition-challenge-2020/

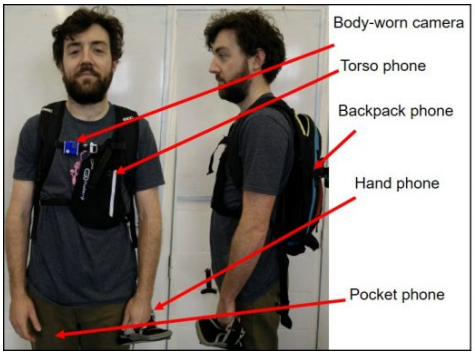

Figure 1: Smartphone positioning during data collection.

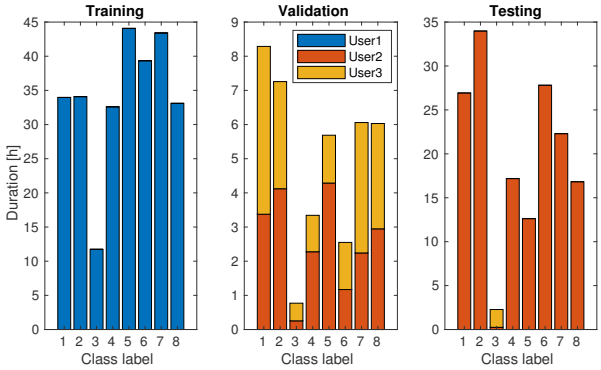

Figure 2: The duration of each class activity in the training and the testing dataset. The 8 classes are: 1 - Still; 2 - Walk; 3 - Run; 4 - Bike; 5 - Car; 6 - Bus; 7 - Train; 8 - Subway.

2 DATASET AND TASK

2.1 Dataset

The challenge uses a subset of the complete Sussex-Huawei Locomotion-Transportation (SHL) dataset [33, 34]. The SHL dataset was recorded over a period of 7 months in 2017 by 3 participants (called User1, User2 and User3) engaging in 8 different modes of transportation and locomotion in real-life setting in the United Kingdom, i.e. Still, Walk, Run, Bike, Car, Bus, Train, and Subway. Each participant carried four smartphones at four body positions simultaneously: in the hand, at the torso, in the hip pocket, in a backpack or handbag (see Fig. 1). The smartphone logged data from 16 sensor modalities. The complete dataset contains up to 2812 hours of labeled data, corresponding to 16,732 km travel distance, and is considered as one of the biggest dataset in the research community.

The SHL Challenge 2020 provided a training, testing and validation dataset[2]. The training dataset comprises 59 days of data from the 4 locations (Bag, Hips, Torso, Hand) indicated in Fig. 1 and from a single "Train" user called User1. The testing contains data from an "Test" user. This "Test" user is in reality a combination of data of User2 and User3, as none of these users could engage in all the activities, and this combination allows to obtain a balanced test dataset. The phone is placed at one location[3], which is unknown to the participants during competition. The validation dataset contains 6 days of data from the 4 locations and from User2 and User3[4]. Fig. 2 depicts the duration of each transportation activity in the training, validation and testing datasets. In total, we have 272×4 hours of training data, 40×4 hours of validation data and 160 hours of testing data, respectively.

The challenge dataset contains the raw data from 7 sensors, including accelerometer, gyroscope, magnetometer, linear acceleration, gravity, orientation, and ambient pressure, which yields a total of 20 sensor channels. The sampling rate of all these sensors is 100 Hz. The activity label (class label) of the training and validation data is provided. The class label for the testing data is invisible to the participants for evaluation.

[2] The exact dates for splitting the dataset will be released at the challenge website http://www.shl-dataset.org/activity-recognition-challenge-2020/.
[3] The testing position is "Hips".
[4] Note that the validation data is same as the previewed version of the SHL dataset. http://www.shl-dataset.org/download/#shldataset-preview.

2.2 Data Format

The training and validation data was generated by segmenting the whole data with a sliding window of 5 seconds with jumping size 5 seconds. The testing data was generated by segmenting the whole data with a sliding window of 5 seconds and jumping size 10 seconds. The frames for the training and validation data are consecutive in time. The frames in the testing data are randomly shuffled. The rationale for this is to force challenge participants to design algorithms which operate with a latency of a maximum of 5 seconds, which can be relevant in real-time interactive applications.

As shown in Table 1, the training and validation data contains the data from four positions: Bag, Torso, Hips and Hand; the testing data contains the data of one of the four position[5], and that position is unknown to the participants. Each position in the training/validation dataset contains 21 plain text files, including 20 sensor files and 1 label file. Each position in the testing dataset only contains the 20 sensor files and excludes the label file.

Each sensor data file in each position of the training set contains a matrix of size 196,072 lines $\times$ 500 columns, corresponding to 196,072 frames each containing 500 samples (5 seconds at the sampling rate 100 Hz). The data in the label file is of the same size (196,072×500), indicating sample-wise transportation activity. Similarly, each sensor data file in each position of the validation set contains a matrix of size 28,789 $\times$ 500. The label file is of same size as the sensor data. Each sensor data file of the testing set contains a matrix of size 57,573 ×500. The label file of the testing set will remain confidential until after the challenge. It is used for performance evaluation by the challenge organizer. The total size of the data in ASCII format are 76.9, 11.3 and 5.6 GB for the training, validation and testing set, respectively.

The 8 numbers in the label file indicate the 8 activities: 1 - Still; 2 - Walk; 3 - Run; 4 - Bike; 5 - Car; 6 - Bus; 7 - Train; 8 - Subway. [6]

2.3 Task and Evaluation

The task is to train a recognition pipeline using the training/validation dataset and then use this system to recognize the transportation

[5] The testing position is "Hips".
[6] Note that we removed all the 'null' class from the raw data.

Table 1: Data files provided by the SHL recognition challenge. Position: B - Bag; T - Torso; Hi - Hips; Ha - Hand; Un - Unknown.

Modality	File	Train (B/T/Hi/Ha)	Validation (B/T/Hi/Ha)	Test (Un)
Accelerometer	Acc_x.txt Acc_y.txt Acc_z.txt	✓	✓	✓
Gyroscope	Gyr_x.txt Gyr_y.txt Gyr_z.txt	✓	✓	✓
Magnetometer	Mag_x.txt Mag_y.txt Mag_z.txt	✓	✓	✓
Linear accelerometer	LAcc_x.txt LAcc_y.txt LAcc_z.txt	✓	✓	✓
Gravity	Gra_x.txt Gra_y.txt Gra_z.txt	✓	✓	✓
Orientation	Ori_w.txt Ori_x.txt Ori_y.txt Ori_z.txt	✓	✓	✓
Pressure	Pressure.txt	✓	✓	✓
Label	Label.txt	✓	✓	×

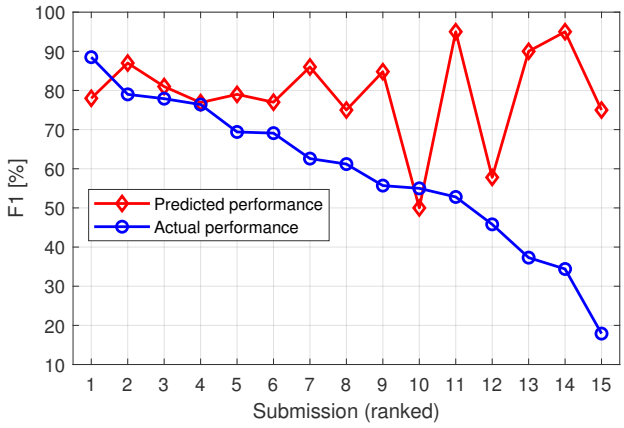

Figure 3: The submissions are ranked based on their F1 scores on the testing set (more details are given in Table 3).

mode from the sensor data in the testing set. The recognition performance is evaluated with the F1 score averaged over all the activities.

Let M_{ij} be the (i, j)-th element of the confusion matrix. It represents the number of samples originally belonging to class i which are recognized as class j. Let $C = 8$ be the number of classes. The F1 score is defined as below.

$$\text{recall}_i = \frac{M_{ii}}{\sum_{j=1}^{C} M_{ij}}, \quad \text{precision}_j = \frac{M_{jj}}{\sum_{i=1}^{C} M_{ij}}, \tag{1}$$

$$F1 = \frac{1}{C} \sum_{i=1}^{C} \frac{2 \cdot \text{recall}_i \cdot \text{precision}_i}{\text{recall}_i + \text{precision}_i}. \tag{2}$$

3 RESULTS

Thirty-three teams expressed interests in the initial registration stage. The teams had 2.7 months (05 April - 25 June 2020) to develop the methods and work on the challenge task. Eventually, 15 teams contributed 15 submissions in the final submission stage by the deadline of 25 June[7]. Table 2 shows the confusion matrices computed on the testing dataset and Table 3 summarizes the technical details of the 15 submissions.

Fig. 3 depicts the F1 score of each submission for the testing set. The submissions are ranked based on their performance on the testing set (Table 3). The performance of the submissions ranges from 17.8% to 88.5%. There is 1 submission achieving an F1 score above 80% on the testing set, 4 between 60% and 70%, 3 between 50% and 60%, and 4 below 50%.

Since each team employs a distinct strategy for cross validation, we request each team to predict their performance for the testing dataset based on the available dataset (i.e. training and validation). We report this predicted performance, as well as the actual performance on the test set in Fig. 3. The predicted result shows that the submissions 3 and 4 generalize well between the training/validation and the testing data. The submissions 1 and 10 shows moderate under-fitting. The other submissions suffer form over-fitting. We briefly introduce the approaches used by the top three contributions.

We-can-fly takes the first place with its F1 score of 88.5%, which is at least 9.5 percent point higher than other submissions [1]. After converting sensor readings (acceleration, linear acceleration, gyroscope, magnetometer, gravity, pressure) from phone-centred coordinate system to human-centred coordinate system, it employs a 1D DenseNet model working on the multi-channel sensor data simultaneously. Interestingly, the submission shows certain under-fitting, with the predicted performance only 78%.

IndRNN achieves the second highest F1 score of 79.0%. After de-rotating the sensor data from phone-centred coordinate system to human-centred coordinate system, the proposed method computes hand-crafted features in the time and frequency domains which are input to an RNN model to predict the labels of the sensor data. The method estimates the phone location in the testing set to be "Hips-Torso" and builds a position-dependent model. It future adopts transfer learning to accommodate user variation.

ThirdTime'sACharm takes the third place with its F1 score 77.9%. It employs a classical machine learning pipeline (XGBoost) working on hand-crafted features (1124 in total). The method trains a position-dependent model after estimating the phone location in the testing set to be "Hips". The method applies semi-supervised learning to estimate user information in the testing set, and improves the recognition performance with a user-dependent model.

4 SUMMARY OF APPROACHES

We categorize the 15 submissions into two families: classical machine learning pipeline (ML) and deep learning pipeline (DL). There are 6 ML submissions and 9 DL submissions.

Fig. 4(a) box-plots the F1 scores obtained by these two families. While ML has lower upper bound than DL, it has a higher bottom bound (excluding the outlier at 17.8%). The smaller dynamic range implies better robustness of ML, which utilizes hand-crafted features that can cope with user and position variation. In contrast, the features learned by DL does not always guarantee a good generalization. The best performance achieved by the DL approach

[7] Submission [13] contributed and evaluated in the SHL Challenge but not accepted for publication after peer review. Submission [14] is evaluated in this summary, but is ineligible for the prize competition due to conflict of interest.

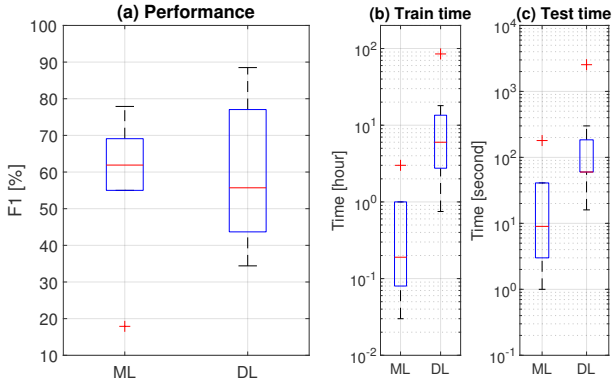

Figure 4: Comparison between machine learning and deep learning approaches. (a) F1 score for the testing data. (b) Training time. (c) Testing time.

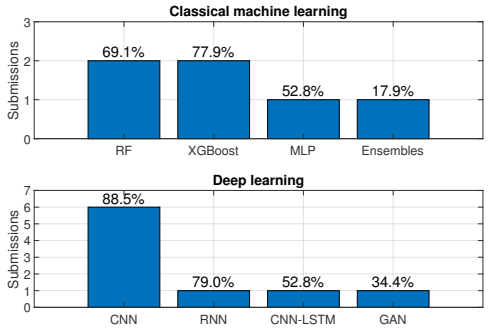

Figure 5: Classical machine learning and deep learning classifiers used by the submissions. The text on top of the bar indicates the highest F1 score achieved by each group of classifiers.

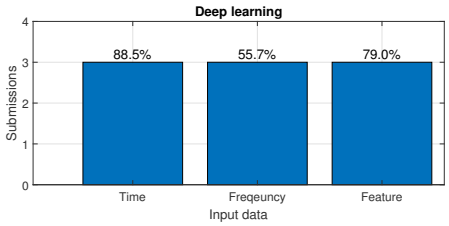

Figure 6: Type of input data to the deep-learning classifier. The text on top of the bar indicates the highest F1 score achieved by each type of input.

(We-can-fly [1], 88.5%) is 10.6 percentage points higher than the best ML approach (ThirdTime'sACharm [3], 77.9%). Fig. 4(b)-(c) show in box-plot the training and testing time by ML and DL approaches, respectively. DL takes much more time for training than ML, and also takes more time for testing.

Fig. 5 depicts the specific classifiers employed by ML and DL pipelines. ML involves four classifiers: extreme gradient boost (XGBoost), random forest (RF), multi-layer perceptron neural network with less than 2 hidden layers (MLP), and ensembles of classifiers (Ensembles). DL involves four classifiers: convolutional neural network (CNN), recurrent neural network (RNN), CNN+LSTM, and generative adversarial network (GAN).

For classical machine learning, RF and XGBoost each has two submissions while MLP and Ensembles each has one submission. Among these classifiers, XGBoost achieves the highest F1 score (77.9%), followed by RF (69.1%) and MLP (52.8%). For deep learning, CNN (6S - 6 submissions) is the most popular classifier, while the other three classifiers each has one submission. CNN achieves the highest F1 score (88.5%), followed by RNN (79.0%).

All the 4 ML approaches uses hand-crafted features as input to the classifier. DL may use different types of data as input to the classifier (Fig. 6): either in the time domain (3S), in the frequency domain (3S), or with hand-crafted features (3S). The time-domain input achieves the highest F1 score (88.5%), followed by hand-crafted features (79.0%), and frequency-domain raw data (55.7%).

4.1 Post-processing

Since the challenge aims to investigate the real-time recognition performance within 5 seconds, we randomly shuffle the temporal order of the testing frames. In SHL 2019, one submission recovered the temporal order of frames by looking at the correlation of sensor data [16]. This year we employed a special processing strategy to prevent this. Specifically, when generating the testing frames, we use a sliding window of length 5 seconds and a jumping size of 10 seconds, so that the correlation between framed data is minimized. Interestingly, this year, one submission still managed to perform temporal smoothing with a proposed nearest neighbour smoothing scheme [8]. The submission finally achieved a F1 score of 61.2%, the 8th position among all the submissions. It would be interesting to see if this scheme can really improve the recognition performance after the release of the ground-truth labels of the testing data.

4.2 Software Implementation

Fig. 7(a) summarizes the programming languages used by the submissions. For ML, Python (5S) is the most popular languages among 6 submissions, followed by Java (1S). No submission chose Matlab. For DL, Python is the only language used by all 9 submissions. Fig. 7(b) summarizes the machine learning libraries used by the submissions. For ML, Scikit-Learn (Python) is the mostly used library (5S), followed by Java AIT (1S). For DL, Keras (4S) is the most popular library, followed by Pytorch (3S) and Tensorflow (2S). Keras is a high-level library building on low-level libraries including Tensorflow, Theano and CNTK, where all the four submissions use the Tensorflow backend.

5 PERFORMANCE ANALYSIS

In Fig. 3, 11 out of 15 submissions achieve F1 scores between 50% and 90%. We analyze the results from the top 11 submissions.

Fig. 8 box-plots the recognition accuracy for each class activity (i.e. the diagonal elements of the confusion matrix in Table 2),

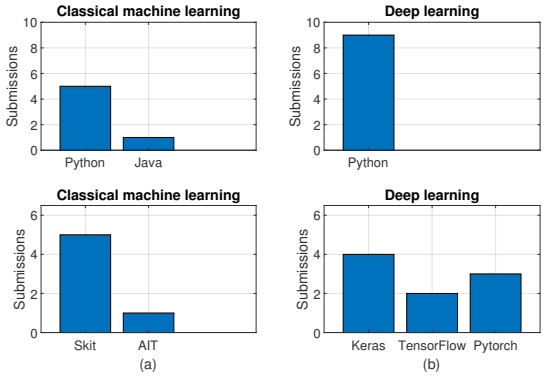

Figure 7: Programming languages and machine libraries used by the submissions for classical machine learning and deep learning. (a) Programming. (b) Library.

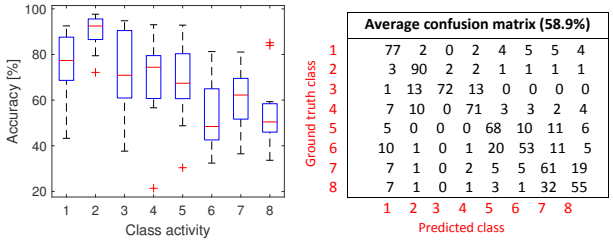

Figure 8: Recognition accuracy for each class activity by the top 11 submissions and the average confusion matrix. The 8 class activities are: 1 - Still; 2 - Walk; 3 - Run; 4 - Bike; 5 - Car; 6 - Bus; 7 - Train; 8 - Subway.

among the top 11 submissions, and also presents the average confusion matrix of their results. It can be observed from the box-plot that the class Bus and Subway are the two most difficult activities to recognize, followed by Train and Car. The first four activities (Still, Walk, Run, and Bike) are better recognized compared to the last four (Car, Bus, Train, and Subway). The motion of the smartphones during walk, run and bike is more distinctive than when the person is sitting or standing in the car, bus, train or subway, thus making the first four activities more distinctive than the last four. There is mutual confusion between the motor vehicles (Car vs Bus), and between the rail vehicles (Train vs Subway). The reason for this is the similar motion patterns during these activities. Some confusion between Still and the four vehicle activities (Car, Bus, Train and Subway) is also observed. This is similar to previous results reported in SHL 2018 [31], SHL 2019 [32] and in our baseline evaluation [34].

Baseline Performance

For reference, we present the baseline performance obtained with the baseline pipeline (CNN-freq) that was employed in SHL 2018 [35]. We simply retrain the same pipeline with this year's challenge data without fine tuning. When using the training set only for model training, we obtain an F1 score of 68.0% for the testing set. When using both the training and validation set for model training, we obtain an F1 score of 78.8%, which is 10.8 percentage points higher than the previous one. This demonstrates that the performance can be improved effectively by incorporating the validation data for model training. The confusion matrix for the highest F1 score (78.8%) is given in Table 2. This baseline result is slightly lower than the second best submission (79.0%).

6 DISCUSSION

The F1 scores reported in SHL 2020 are at a similar level to the ones reported in SHL 2019 [32]. SHL 2020 has 11 submission with F1 scores above 50%, with an average score among these of 58.9%. SHL 2019 has 13 submissions with F1 scores above 50%, with an average score among these of 61.6%. The best performance reported SHL 2020 (88.4%) is higher than the best one in SHL 2019 (78.4%). The average performance of SHL 2020 is slightly lower than SHL 2019. SHL 2019 focuses on position-independent evaluation while SHL 2020 focuses on both position-independent and user-independent evaluation. On the one hand, the user variation in SHL 2020 makes the recognition task more challenging. On the other hand, the recognition at "Hips" position in SHL 2020 is easier than the "Hand" position at SHL 2019. Taking these two issues both into consideration, it is reasonable that SHL 2019 and 2020 achieve a similar performance.

The participant teams have employed various techniques to tackle the challenge of achieving position-independence and user-independence. We summarize them as three schemes below.

Robust representation. Most submissions use orientation/position independent representation of the sensor data. For instance, the magnitude of sensor data, which is a combination of the data at three coordinates, has been widely used across the teams for feature computation or classifier training. Several submissions convert sensor data from phone-centred coordinate system to human-centred coordination system, which can increase the robustness to phone placement [1, 2, 6, 9, 10]. The submissions [1, 2] obtain the top two performance among all the participants.

Position-specific modeling. The location of the phone in the testing dataset is unknown, but it is known to be one of the four locations in the training dataset. Many submissions exploit this fact to predict the location of the phone first, and then do position-dependent training. Based on the summary in Table 3, 8 out of 15 submissions employ a machine learning scheme to estimate the location of the phone, where 7 out of these 8 submissions estimated the location correctly as "Hips" [2–4, 8, 9, 12, 14] and only one submission estimated the location to be "Hand" [6]. The highest F1 score achieved by these 8 submissions is 79.0%, which is ranked second among all the submissions [2]. The submission [14] employs generative adversarial networks to improve the recognition performance for a specific position. While this interesting idea achieved quite good results for the "Hip" position in the validation dataset (95.0%), the performance for the testing data is quite low (34.4%). A more in-depth investigation is needed for this approach.

User-specific modeling. Since the validation and the testing dataset contain the data from the same users (User2 and User3), the submission [3] exploits this fact to develop user-dependent model. It employs transfer learning techniques to train two user-dependent models with assistance from the validation dataset. The testing data

is clustered into two users and processed with two user-specific models accordingly. The method achieved an F1 score of 77.9%, which is ranked the third place among all the participants. The submission [3] also applied transfer learning to accommodate user variation, achieving the second highest F1 score 79.0%.

7 CONCLUSION

We reported the achievements obtained during the SHL recognition challenge 2020, where one submission achieved an F1 score between 80% and 90%, three submissions achieved F1 scores between 70% and 80%, four submissions between 60% and 70%, three between 50% and 60%. We summarized the approaches used by these submissions and analyzed their performance. Because the approaches are implemented by different research groups with varying expertise, the conclusions drawn will be confined to the submissions of the challenge.

The submissions can be divided into ML and DL pipelines. This year more submissions employed DL and led to higher performance. The highest performance is achieved by an DL approach (88.5%), which is 10.6 percentage points higher than the best ML approach (77.9%).

Various schemes have been employed by the participant teams to tackle the challenge of the variation of user and position, including robust representation, position-specific modeling, and user-specific modeling. The provides a good insight for developing novel algorithms for position-independent and user-independent activity recognition.

ACKNOWLEDGMENTS

This work was supported by HUAWEI Technologies within the project "Activity Sensing Technologies for Mobile Users".

REFERENCES

[1] Y. Zhu, et al. DenseNetX and GRU for the Sussex-Huawei locomotion-transportation recognition challenge. Proc. UbiComp/ISWC 2020.
[2] B. Zhao, et al. IndRNN based long-term temporal recognition in the spatial and frequency domain. Proc. UbiComp/ISWC 2020.
[3] S. Kalabakov, et al. Tackling the SHL Challenge 2020 with person-specific classifiers and semi-supervised learning. Proc. UbiComp/ISWC 2020.
[4] K. Yaguchi, et al. Human activity recognition using multi-input CNN model with FFT spectrograms. Proc. UbiComp/ISWC 2020.
[5] C. Naseeb, et al. Activity recognition for locomotion and transportation dataset using deep learning. Proc. UbiComp/ISWC 2020.
[6] M. S. Siraj, et al. UPIC: user and position independent classical approach for locomotion and transportation modes recognition. Proc. UbiComp/ISWC 2020.
[7] S. Brajesh, et al. Ensemble approach for sensor-based human activity recognition. Proc. UbiComp/ISWC 2020.
[8] P. Widhalm, et al. Tackling the SHL recognition challenge with phone position detection and nearest neighbour smoothing. Proc. UbiComp/ISWC 2020.
[9] R. Sekiguchi, et al. Ensemble learning for human activity recognition. Proc. UbiComp/ISWC 2020.
[10] Y. Tseng, et al. Hierarchical Classification Using ML/DL for Sussex-Huawei Locomotion-Transportation (SHL) Recognition Challenge. Proc. UbiComp/ISWC 2020.
[11] B. Friedrich, et al. Combining LSTM and CNN for mode of transportation classification from smartphone sensors. Proc. UbiComp/ISWC 2020.
[12] M. Hamid, et al. A multi-view architecture for the SHL challenge. Proc. UbiComp/ISWC 2020.
[13] Team-X, A data-fusion deep learning model for transportation mode detection on extracted features. (withdrawn)
[14] L. Gunthermann, et al. Smartphone location identification and transport mode recognition using an ensemble of generative adversarial networks. Proc. UbiComp/ISWC 2020.
[15] G. Dogan, et al. Where are you? Human activity recognition with smartphone sensor data. Proc. UbiComp/ISWC 2020.
[16] V. Janko, M. Gjoreski, C. M. De Masi, et al. Cross-location transfer learning for the Sussex-Huawei locomotion recognition challenge. Proc. 2019 ACM International Joint Conference and 2019 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, 2019, 730-735.
[17] J. Engelbrecht, M. J. Booysen, G. van Rooyen, F. J. Bruwer. Survey of smartphone-based sensing in vehicles for intelligent transportation system applications. IET Intelligent Transport Systems, 9(10): 924-935, 2015.
[18] Y. Vaizman, K. Ellis, G. Lanckriet. Recognizing detailed human context in the wild from smartphones and smartwatches. IEEE Pervasive Computing, 16(4): 62-74, 2017.
[19] E. Anagnostopoulou, J. Urbancic, E. Bothos, B. Magoutas, L. Bradesko, J. Schrammel, G. Mentzas. From mobility patterns to behavioural change: leveraging travel behaviour and personality profiles to nudge for sustainable transportation. Journal of Intelligent Information Systems, 2018: 1-22, 2018.
[20] D. A. Johnson, M. M. Trivedi. Driving style recognition using a smartphone as a sensor platform. Proc. IEEE Conference on Intelligent Transportation Systems, 2011, 1609-1615.
[21] W. Brazil, B. Caulfield. Does green make a difference: The potential role of smartphone technology in transport behaviour. Transportation Research Part C: Emerging Technologies, 37: 93-101, 2013.
[22] J. Froehlich, T. Dillahunt, P. Klasnja, J. Mankoff, S. Consolvo, B. Harrison, J. A. Landay. Ubigreen: Investigating a mobile tool for tracking and supporting green transportation habits. Proc. SIGCHI Conference on Human Factors Computing Systems, 2009, 1043-1052.
[23] C. Cottrill, F. Pereira, F. Zhao, I. Dias, H. Lim, M. Ben-Akiva, P. Zegras. Future mobility survey: Experience in developing a smartphone-based travel survey in Singapore. Transportation Research Record: J. Transportation Research Board, 2354(1): 59-67, 2013.
[24] S. C. Mukhopadhyay. Wearable sensors for human activity monitoring: A review. IEEE Sensors Journal, 15(3): 1321-1330, 2015.
[25] G. Castignani, T. Derrmann, R. Frank, T. Engel. Driver behavior profiling using smartphones: A low-cost platform for driver monitoring. IEEE Intelligent Transportation Systems Magazine. 7(1): 91-102, 2015.
[26] H. Xia, Y. Xiao, J. Jian, Y. Chang. Using smart phone sensors to detect transportation modes. Sensors, 14(11): 20843-20865, 2014.
[27] M. C. Yu, T. Yu, S. C. Wang, C. J. Lin, E. Y. Chang. Big data small footprint: the design of a low-power classifier for detecting transportation modes. Proc. Very Large Data Base Endowment, 2014, 1429-1440.
[28] S. Richoz, M. Ciliberto, L. Wang, P. Birch, H. Gjoreski, A. Perez-Uribe, D. Roggen. Human and machine recognition of transportation modes from body-worn camera images. Proc. Joint 8th Int. Conf. Informatics, Electronics & Vision and 3rd Int. Conf. Imaging, Vision & Pattern Recognition, 2019, 67-72.
[29] L. Wang, D. Roggen. Sound-based transportation mode recognition with smartphones. Proc. IEEE International Conference on Acoustics, Speech and Signal Processing, 2019, 930-934.
[30] S. Richoz, L. Wang, P. Birch, D. Roggen. Transportation mode recognition fusing wearable motion, sound and vision sensors. IEEE Sensors Journal, 20(16): 9314-9328, 2020.
[31] L. Wang, H. Gjoreski, K. Murao, T. Okita, D. Roggen. Summary of the Sussex-Huawei locomotion-transportation recognition challenge. Proc. 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, 2018, 1521-1530.
[32] L. Wang, H. Gjoreski, M. Ciliberto, P. Lago, K. Murao, T. Okita, D Roggen. Summary of the Sussex-Huawei locomotion-transportation recognition challenge 2019. Proc. 2019 ACM International Joint Conference and 2019 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, 2019, 849-856.
[33] H. Gjoreski, M. Ciliberto, L. Wang, F.J.O. Morales, S. Mekki, S. Valentin, D. Roggen. The university of Sussex-Huawei locomotion and transportation dataset for multimodal analytics with mobile devices. IEEE Access, 2018, 42592-42604.
[34] L. Wang, H. Gjoreski, M. Ciliberto, S. Mekki, S. Valentin, D. Roggen. Enabling reproducible research in sensor-based transportation mode recognition with the Sussex-Huawei dataset. IEEE Access, 2019, 10870-10891.
[35] L. Wang, H. Gjoreski, M. Ciliberto, S. Mekki, S. Valentin, D. Roggen. Benchmarking the SHL recognition challenge with classical and deep-learning pipelines. Proc. 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, 2018, 1626-1635.

Table 2: Confusion matrix (F1 score) of each submission for the testing dataset. The 8 class activities are: 1 - Still; 2 - Walk; 3 - Run; 4 - Bike; 5 - Car; 6 - Bus; 7 - Train; 8 - Subway.

Ground truth class	We_can_fly (88.47%)								IndRNN (78.98%)								ThirdTimesACharm (77.89%)								DSML-TDU (76.40%)							
1	93	2	0	1	1	0	2	2	88	2	0	0	2	0	4	4	86	4	0	1	2	1	3	3	90	4	0	0	1	0	3	3
2	3	95	0	1	0	0	0	0	3	96	0	0	0	0	0	0	1	98	0	0	0	0	0	0	2	98	0	0	0	0	0	0
3	0	2	95	3	0	0	0	0	0	7	93	0	0	0	0	0	0	8	92	1	0	0	0	0	0	13	87	0	0	0	0	0
4	3	2	0	93	1	0	1	1	4	17	0	78	0	0	0	0	4	9	0	76	0	2	1	8	3	3	0	93	0	0	0	0
5	4	0	0	0	87	6	2	2	5	0	0	0	67	14	11	2	3	0	0	0	93	1	3	0	3	0	0	0	71	11	12	2
6	7	1	0	0	8	81	2	1	6	1	0	0	18	67	6	2	11	1	0	0	17	62	6	3	8	2	0	0	16	65	6	2
7	6	0	0	1	1	1	81	10	6	1	0	0	2	1	59	30	8	1	0	0	4	3	64	20	16	1	0	1	3	4	49	26
8	4	0	0	0	0	0	10	85	5	0	0	0	1	0	10	84	6	1	0	0	1	0	35	56	16	1	0	1	2	2	17	59
	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
	Predicted class																															

Ground truth class	DL_Lock (69.40%)								RED_CIRCLE (69.10%)								ASIA (62.55%)								MDCA (61.19%)							
1	77	1	0	1	5	2	8	6	86	2	0	2	2	2	5	1	75	1	0	5	2	6	9	2	76	3	0	1	9	6	4	1
2	5	93	0	0	0	0	0	1	4	90	0	4	1	0	0	0	4	88	0	5	1	1	1	0	3	72	20	2	3	0	0	0
3	0	40	59	1	0	0	0	0	0	21	69	10	0	0	0	0	0	8	40	51	0	0	0	0	0	11	82	7	0	0	0	0
4	7	11	0	73	2	6	0	1	7	8	0	74	3	4	1	3	6	3	0	74	4	9	1	3	2	6	2	80	8	1	1	0
5	5	0	0	0	60	17	13	6	6	0	0	0	63	21	9	2	7	0	0	0	64	13	14	2	4	0	0	0	80	3	11	1
6	5	1	0	0	15	64	10	5	10	1	0	0	27	48	11	2	12	0	0	1	25	44	15	3	17	1	0	0	31	33	15	3
7	5	1	0	1	8	6	62	18	5	1	0	2	7	3	67	15	4	2	0	3	5	4	70	12	5	4	0	1	4	4	73	9
8	5	0	0	0	4	2	38	50	5	0	0	0	2	1	41	50	4	1	0	1	2	1	54	38	5	1	0	0	5	1	42	46
	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
	Predicted class																															

Ground truth class	TDU_BSA (55.66%)								SensingGO (54.97%)								103114102106 (52.80%)								Eagles (45.83%)							
1	67	3	0	1	4	1	11	13	43	2	0	4	9	36	5	1	65	5	0	7	4	3	5	12	83	1	0	2	4	3	5	3
2	4	79	1	3	1	1	10	1	1	86	0	5	2	3	3	1	4	92	0	1	0	1	1	1	7	86	0	4	0	1	1	1
3	5	15	71	0	1	0	5	3	0	0	38	62	0	0	0	0	1	23	67	10	0	0	0	0	0	30	27	43	0	0	0	0
4	19	2	0	57	1	2	3	17	1	1	0	57	16	9	7	9	26	44	1	21	0	3	1	5	10	66	0	4	3	9	3	6
5	6	0	0	0	30	4	23	37	2	0	0	0	80	9	6	2	12	1	0	2	49	12	12	11	26	0	0	0	15	36	20	3
6	8	1	0	1	23	32	25	9	11	0	0	1	32	42	9	3	12	3	0	2	9	43	13	18	22	1	0	0	5	49	21	2
7	10	2	0	2	5	3	50	27	3	1	0	3	9	13	56	13	13	1	0	3	9	9	36	29	8	0	0	1	5	4	61	21
8	8	3	0	1	2	1	33	52	3	0	0	1	7	4	51	34	17	1	0	2	5	3	26	46	7	0	0	0	1	1	45	47
	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
	Predicted class																															

Ground truth class	Team-X (37.25%)								Noname (34.44%)								Petrichor (17.84%)								Baseline (78.81%) [35]							
1	59	5	0	1	20	5	6	4	59	4	0	0	7	17	1	12	12	1	0	0	10	35	42	0	85	3	0	1	2	1	5	3
2	4	80	14	0	1	0	0	0	2	95	0	0	0	0	1	2	4	54	0	6	2	10	24	0	3	95	0	1	0	0	0	0
3	0	50	50	0	0	0	0	0	1	72	18	0	0	9	0	0	19	14	0	2	1	5	59	0	0	2	83	14	0	0	0	0
4	23	57	0	5	7	0	0	8	12	45	0	0	0	16	0	26	7	34	0	1	12	15	30	1	5	5	0	86	1	2	0	0
5	3	1	0	1	74	9	12	1	12	1	0	8	51	27	1	1	12	4	0	0	18	52	14	0	6	0	0	0	74	10	6	3
6	17	8	0	2	44	17	9	3	26	3	0	0	16	35	2	18	17	2	0	0	22	22	37	0	7	1	0	0	13	70	7	2
7	9	3	0	3	25	4	42	15	19	2	0	0	17	31	2	28	11	2	0	0	11	17	58	0	9	1	0	0	3	2	64	21
8	6	2	0	1	18	2	47	25	9	2	0	0	17	28	0	44	6	1	0	0	12	19	61	0	6	1	0	1	1	0	23	68
	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
	Predicted class																															

Table 3: Summary of the SHL recognition challenge 2020.

App.	Rank	Team	Classifier	Input	Location estimation	Sensor modality	Performance		Computational resource		Time		Implementation		Model size (MB)	Ref
							Predict	Test	CPU	GPU	Train [h]	Test [s]	Lang.	Library		
ML	3	ThirdTime's ACharm	XGBoost	Features	Hips	LAGMOPR	81.0%	77.9%	8-core@3.6GHz RAM-16G	RTX 2060	0.08	15	Python	Scikit-learn	60	[3]
	6	RED_CIRCLE	RF	Features	Hand	LAGMOPR	77.0%	69.1%	2-core@2.3GHz RAM-13G	/	1	41	Python	Scikit-learn	1825	[6]
	7	ASIA	RF	Features	/	LAGMP	86.0%	62.6%	8-core@2.3GHz RAM-16G	/	0.08	1	Python	Scikit-learn	278	[7]
	8	MDCA	MLP	Features	Hips	AGMOPR	75.0%	61.2%	8-core@1.9GHz RAM-16G	/	0.03	3	Java	AIT	0.2	[8]
	10	SensingGO	XGBoost	Features	/	LAGMP	50.0%	55.0%	12-core@2GHz RAM-128G	/	0.3	180	Python	Scikit-learn	0.7	[10]
	15	Petrichor	Ensemble	Features	/	LAGMP	75.0%	17.9%	16-core@1.9GHz RAM-256G	/	3	3	Python	Scikit-learn	410	[15]
DL	1	We-can-fly	CNN	Time	/	LAGMPR	78.0%	88.5%	14-core@2.6GHz RAM-128G	Tesla V100	6	120	Python	Pytorch	30	[1]
	2	IndRNN	RNN	Features	Hips-Torso	AGMOP	87.0%	79.0%	10-core@2.4GHz RAM-256G	Titan XP	18	2540	Python	Pytorch	43	[2]
	4	DSML_TDU	CNN	Features	Hips	LGM	67.9%	76.4%	8-core@3.5GHz RAM-128G	GTX 1080Ti	2	300	Python	Keras (Tensorflow)	103	[4]
	5	DL_Lock	CNN	Features	/	LAGM	79.0%	69.4%	6-core@3.5GHz RAM-32G	RTX 2080 Ti	0.75	16	Python	Keras (Tensorflow)	19.3	[5]
	9	TDU_BSA	CNN	Frequency	Hips	LAGMOP	84.8%	55.7%	6-core@3.2GHz RAM-32G	RTX 2060	12	60	Python	Keras (Tensorflow)	114	[9]
	11	103114102106	CNN + LSTM	Time	/	LAGMOPR	95.0%	52.8%	HPC Cluster	HPC Cluster	85	146	Python	Tensorflow	39	[11]
	12	Eagles	CNN	Time	Hips	LAGMOPR	57.8%	45.8%	56-core@2.2GHz RAM-96G	4 × Tesla K40M	3	60	Python	Keras (Tensorflow)	1	[12]
	13	Team-X	CNN	Frequency	/	AGMOR	90.0%	37.3%	?	Tesla P100	10	?	Python	Tensorflow	?	[13]
	14	Noname	GAN	Features	Hips	LAGMOPR	95.0%	34.4%	6-core@3.2GHz RAM-32G	GTX 2080	3	60	Python	Pytorch	14	[14]

Sensor modalitiy: L - Linear accelerometer; A - Accelerometer; G - Gyroscope; M - Magnetometer; O - Orientation; P - Pressure; R - Gravity.