2023 IEEE International Conference on Smart Computing (SMARTCOMP) | 979-8-3503-2281-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/SMARTCOMP58114.2023.00022

Combining Public Human Activity Recognition Datasets to Mitigate Labeled Data Scarcity

Riccardo Presotto
EveryWare Lab, University of Milan
Milan, Italy
riccardo.presotto@unimi.it

Sannara Ek
Univ. Grenoble Alpes, CNRS
LIG F-38000, Grenoble, France
sannara.ek@univ-grenoble-alpes.fr

Gabriele Civitarese
EveryWare Lab, University of Milan
Milan, Italy
gabriele.civitarese@unimi.it

François Portet
Univ. Grenoble Alpes, CNRS
LIG F-38000, Grenoble, France
francois.portet@imag.fr

Philippe Lalanda
Univ. Grenoble Alpes, CNRS
LIG F-38000, Grenoble, France
philippe.lalanda@imag.fr

Claudio Bettini
EveryWare Lab, University of Milan
Milan, Italy
claudio.bettini@unimi.it

***Abstract*—The use of supervised learning for Human Activity Recognition (HAR) on mobile devices leads to strong classification performances. Such an approach, however, requires large amounts of labeled data, both for the initial training of the models and for their customization on specific clients (whose data often differ greatly from the training data). This is actually impractical to obtain due to the costs, intrusiveness, and time-consuming nature of data annotation. Moreover, even with the help of a significant amount of labeled data, model deployment on heterogeneous clients faces difficulties in generalizing well on unseen data. Other domains, like Computer Vision or Natural Language Processing, have proposed the notion of pre-trained models, leveraging large corpora, to reduce the need for annotated data and better manage heterogeneity. This promising approach has not been implemented in the HAR domain so far because of the lack of public datasets of sufficient size. In this paper, we propose a novel strategy to combine publicly available datasets with the goal of learning a generalized HAR model that can be fine-tuned using a limited amount of labeled data on an unseen target domain. Our experimental evaluation, which includes experimenting with different state-of-the-art neural network architectures, shows that combining public datasets can significantly reduce the number of labeled samples required to achieve satisfactory performance on an unseen target domain.**

***Index Terms*—human activity recognition; mobile/wearable computing; transfer learning**

I. Introduction

Despite years of research and progress [1], the field of sensor-based Human Activity Recognition (HAR) on mobile or wearable devices remains a hot research topic. This is due to several reasons. First, the field has major social importance in the sense that it is necessary for the development of value-added services in the fields of health, well-being, or task monitoring in industry. Secondly, although constant progress can be reported, the problem is far from being solved and today solutions are difficult to be deployed in real-world scenarios [2]. Most recent approaches are based on machine learning techniques. More precisely, supervised learning leads to strong classification performances for HAR. This, however, requires large amounts of labeled data, both for the initial training of the models and for their customization on specific clients (whose data often differ greatly from the training data). This is actually a problem because of the costs, intrusiveness, and time-consuming nature of data collection and annotation. Even with the help of a number of labeled data, models deployed on heterogeneous clients face difficulties to generalize well on unseen data.

In other domains, such as Computer Vision or Natural Language Processing, learning a model on one large dataset [3], [4] makes it possible to transfer knowledge to new domains using only a limited amount of specific data [5]. However, in the HAR domain, transferring knowledge from a single dataset is not effective because of the lack of publicly available large and heterogeneous datasets, due to the above-mentioned problems. We call this the labeled data scarcity problem. Ideally, a pre-trained model should be able to cope with the variability of devices [6], the variability of applications, and also the behavioral differences of users [7]. In fact, there are a number of public datasets but they are generally small and acquired in scripted scenarios (e.g. Mobiact [8], MotionSense [9], PAMAP2 [10]). The few ones that are relatively large and acquired in the wild often include noisy data and incorrect annotations (e.g. ExtraSensory [11]).

Transfer learning from one dataset to another has been already attempted but existing works showed that such approaches lead to low recognition rates [12]. To further demonstrate the problem, Figure 1 presents the result of an experiment showing the challenge of training a model on one dataset and expecting it to perform well on a different dataset. This cross-dataset evaluation was performed using some of the most commonly used publicly available datasets for HAR and using a state-of-the-art model [13]. The figure clearly shows that the model performs well when trained and tested on the same dataset, but its performance drastically deteriorates in cross-dataset scenarios. These results highlight the challenges associated with developing models that can generalize well across diverse datasets and underscore the need for developing more robust approaches that can effectively leverage knowledge from pre-training on large, diverse, and representative

2693-8340/23/$31.00 ©2023 IEEE
DOI 10.1109/SMARTCOMP58114.2023.00022

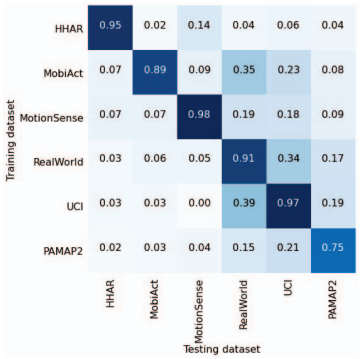


Fig. 1: Cross-dataset evaluation by using a state-of-the-art classifier [13]. A cell at coordinates i, j shows the F1 score obtained by training the model with 70% of dataset i on 30% of dataset j.

datasets.

In this paper, we hypothesize that, by combining several public HAR datasets that are often used as benchmarks by the HAR community, it is possible to pre-train models that can be fine-tuned on target clients with a limited amount of labeled data. Specifically, we propose a novel approach to combining datasets and we report our experience considering 6 public datasets to generate a pre-trained model. Also, we propose an evaluation strategy called Leave-One-Dataset-Out that effectively assesses the generalization capabilities of our approach.

Our experiments show that our pre-trained model can be effectively transferred to an unseen target HAR domain using a limited amount of labeled data for fine-tuning. To the best of our knowledge, this is the first work proposing to combine several heterogeneous HAR datasets to mitigate the labeled data scarcity problem. Precisely, the contributions of this paper are the following:

- We pre-trained models using a combination of diverse publicly available HAR datasets. We show that such models can be effectively fine-tuned efficiently with limited data of unseen target domains.
- We then propose a novel realistic and challenging evaluation methodology termed Leave-One-Dataset-Out (LODO), and we use it to extensively assess the effectiveness of our approach on different state-of-the-art neural network architectures.
- The pre-trained model, the corresponding training data, and the code are publicly available to make this research reproducible[1].

II. Related Work

Human activity recognition (HAR) using machine learning has traditionally relied on supervised learning approaches, which require a significant amount of labeled data for training to achieve high accuracy [14]. However, obtaining labeled activity data is often time-consuming, intrusive, costly, and hence unfeasible on a large scale [15]. To overcome the scarcity of labeled samples in the target domain, domain adaptation (DA) techniques have emerged to leverage labeled samples that are available from a different source domain [16]. Among the many DA techniques, transfer learning has shown valuable results in various domains, including natural language processing [17], [18]. However, adapting transfer learning to HAR is challenging due to device variability, body displacement, and behavioral differences among users, making it difficult to generate a generic model that can quickly adapt to diverse datasets [19], [20]. Several researchers have investigated how to apply transfer learning methods to solve cross-domain HAR problems, including cross-device [21], [22], cross-sensor-installation-position [20], [23], and cross-persons [17], [24]. However, publicly available datasets for HAR have limited labeled samples and classes, which restricts the effectiveness of cross-domain adaptation works in this area [25]. For instance, effective deep CNN frameworks have been proposed to enable the generation of a model that can efficiently adapt across diverse datasets [26], but they require that the classes between the source and the target domain overlap, hence limiting their effectiveness in real-world scenarios. Moreover, most of the studies in this field have been conducted by transferring knowledge from one single dataset to another, resulting in performance variations depending on the compatibility among the data across the considered dataset pairs [25]. Furthermore, due to the data scarcity problem in HAR, using only the data from one dataset to generate the model that needs to be transferred to another domain may not be sufficient. A possible solution to mitigate this problem is to pre-train the model on a different source domain before fine-tuning on the data-scarce target domain [12], [25]–[28]. The obtained results are promising even in the case very few data are used for fine-tuning the model in the target domain. However, this approach suffers from generalization across domains that differ from the target one. We argue that utilizing only a single dataset for the pre-training HAR models is not sufficient to find common features in the target domain, especially with the highly heterogeneous nature of smartphone data. Another approach to expand the dataset available for training may consist of using data augmentation techniques like GANs [29], [30]. GANs have been also used to facilitate cross-subject transfer learning [31]. However, GANs still require a significant amount of labeled samples to be trained.

In recent years, self-supervised learning (SSL) approaches have gained attention as a general framework for learning from unlabeled data through a pretext task [5], [32]. SSL has also been applied to sensor-based HAR, leading to promising results [33], [34]. However, even a good unsupervised or SSL model that permits adaptation using a short amount of data still requires a significant amount of data, as exemplified in other domains such as vision and speech [35]. In the HAR domain, even non-annotated data is difficult to obtain.

[1]https://github.com/getalp/SmartComp2023-HAR-Supervised-Pretraining

III. Combining datasets for model pre-training

In this section, we outline our strategy for combining datasets to create a pre-trained model that can generalize well across all datasets. We begin by providing an overview of the datasets we considered for this task. Next, we describe our proposed approach for combining these datasets to generate the pre-trained model.

A. Datasets

In the following, we describe the datasets we considered in this work. We note here that we explicitly seek datasets with both Accelerometer and Gyroscope data, as utilizing the two sensors together gave the best performance [36]. Information on the datasets is summarized in Table I.

1) Heterogeneity Human Activity Recognition (HHAR) [6] : The HHAR dataset consists of 4.5 hours of recorded activities from 9 participants. Each participant wore 8 Android smartphones in a tight pouch carried around on the waist, and 4 Android smartwatches while performing 6 different activities (Biking, Sitting, Standing, Walking, Upstairs, and Downstairs). All 12 devices recorded the activities using accelerometer and gyroscope measurements at their maximum sampling rates, which ranged from 50 Hz to 200 Hz. The HHAR dataset represents a heterogeneous learning environment due to the variety of devices used in the data collection process.

2) MobiAct [8] : This dataset includes labeled data from 61 different subjects with high variance in age and physical characteristics. The dataset contains data from a triaxial accelerometer, gyroscope, and magnetometer embedded into a Samsung Galaxy S3 smartphone carried by users while performing 9 physical activities. During the acquisition process, the users were left free to position the smartphone with a random orientation into one of their trousers' pockets. The physical activities included in this dataset are the following: Standing, Walking, Jogging, Jumping, Upstairs, Downstairs, Sitting, Car step in, Car step out. The adopted data acquisition frequency is the highest enabled by the sensors of the selected smartphone (i.e., at most $200Hz$).

3) MotionSense [9]: This dataset includes data from 24 subjects very heterogeneous in terms of gender, age, weight, and height. The six activities performed were Downstairs, Upstairs, Sitting, Standing, Walking, and Jogging/Running. The tests were conducted using an Apple iPhone 6s that was kept in the subject's front pocket and recorded data from accelerometer, attitude, and gyroscope sensors at a 50 Hz sampling rate.

4) RealWorld [37]: This dataset consists of 18 hours of recorded accelerometer and gyroscope data collected in 2016 from 15 subjects using a Samsung Galaxy S4 smartphone and an LG G Watch R placed at 7 different body positions: head, chest, upper arm, waist, forearm, thigh, and shin. The sampling rate was 50 Hz. The subjects performed various activities outdoors without any restrictions, and the data was labeled into 8 activities: Downstairs, Upstairs, Lying, Sitting, Standing, Walking, Jumping, and Running. This dataset aims to simulate the class imbalance that resembles the ones of realistic datasets. For instance, the "standing" activity represents 14% of the data while the "jumping" activity only accounts for 2%.

5) UCI Human Activity Recognition [38] : This dataset was collected using a Samsung Galaxy S II placed on the participant's waist, with a sampling rate of 50 Hz. The dataset comprises 3.6 hours of recorded activities from 30 participants with an age range of 19 to 48 years old. The six activities recorded are Walking, Upstairs, Downstairs, Sitting, Standing, and Lying, and the experiments were conducted in a controlled indoor lab environment.

6) PAMAP2 [10]: This dataset was recorded in a controlled setting where 9 participants carried out 12 activities of daily living, including domestic tasks and physical exercises. The activity data were recorded using three Colibri wireless IMUs (inertial measurement units) consisting of triaxial accelerometers, gyroscopes, and magnetometers attached to the participants' ankle, chest, and wrist. The activities monitored were: rope jumping, lying, sitting, standing, walking, running, cycling, nordic walking, ascending stairs, descending stairs, vacuum cleaning, and ironing. The sampling rate adopted for data acquisition was set to $100Hz$.

B. Datasets combination strategy

In the following, we describe how we pre-processed and put together the above-mentioned datasets. First, each dataset was downsampled to 50 Hz to align with the optimal frequency for HAR on smartphones as suggested by a recent survey [39]. Indeed, this survey suggests that a sampling rate between 20 Hz and 50 Hz is ideal for this task, and accelerometers and gyroscopes are the most suitable sensors. Using higher frequencies beyond 50 Hz would result in increased computation and memory costs with only marginal improvement in performance.

After downsampling, each dataset was standardized individually using sensor-wise z-normalization to center the data. We avoid standardizing all datasets together as a whole to avoid datasets with lower samples being under-presented when combined with many of the larger datasets (e.g. the large HHAR vs the small UCI dataset). In order to focus on the data scarcity problem and to limit influences on the results caused by domain shifts from different positions [36], from each dataset we use data only from the 'waist' position, which is usually the most common one and it is included in all of the datasets studied in this work.

Finally, based on past studies [40], the data was then segmented into instances using a window length of 128 samples (2.56 seconds) with a 50% overlap over all 6 channels from the accelerometer and gyroscope readings. In order to combine the datasets together, we performed a union of the physical activities involved. This combination led to a dataset with 147 subjects, 10 unique activities ('Downstairs', 'Upstairs', 'Running', 'Sitting', 'Standing', 'Walking',' Lying',' Cycling', 'Nordic Walking', 'Jumping'), and 213,289 data samples (≈ 151 hours of usable data), and 12 different devices for data

TABLE I: Summary of datasets characteristics

Dataset	# of samples	# of users	Adopted Devices	Sampling rate	Device position	Activities
HHAR	85,567	9	Smartphones: Samsung Galaxy S3 mini, Samsung Galaxy S3, LG Nexus 4, Samsung Galaxy S+ Smartwatches: LG watches, Samsung Galaxy Gears	from 50 Hz to 200 Hz	Smartphones: Waist Smartwatches: Wrist	Biking, Sitting, Standing, Walking, Upstairs, Downstairs
MobiAct	18,634	61	Samsung Galaxy S3	from 50 Hz to 200 Hz	Waist	Standing, Walking, Jogging, Jumping, Upstairs, Downstairs, Sitting, Car step in, Car step out
MotionSense	17,231	24	Apple iPhone 6s	50 Hz	Waist	Downstairs, Upstairs, Sitting, Standing, Walking, Running
RealWorld	356,427	15	Samsung Galaxy S4 LG G Watch R	50 Hz	Smartphones: Head, Chest, Upper arm, Waist, Thigh, Shin Smartwatches: Forearm	Downstairs, Upstairs, Lying, Sitting, Standing, Jumping, Walking, Running
UCI	10,299	30	Samsung Galaxy S II	50 Hz	Waist	Walking, Upstairs, Downstairs, Sitting, Standing, Lying
PAMAP2	15,177	8	Colibri wireless IMU sensors	100 Hz	Waist, Chest, Wrist	Rope Jumping, Lying, Sitting, Standing, Walking, Running, Cycling, Nordic walking, Upstairs, Downstairs, Vacuum cleaning, Ironing

acquisition. Despite considering only one on-body position, we believe that the wide variety of subjects and devices presents a challenging learning problem due to the heterogeneity of the combined data.

C. Pre-training

The set of labeled datasets can then be used to train a deep neural network model in a fully supervised learning fashion. Specifically, we consider the neural network as composed of two parts: the *feature extractor* and the *classification head*. During the pre-training phase, we train the whole network (i.e. *feature extractor* and *classification head*). We will refer to the *feature extractor* trained on several joint datasets as the *pre-training model*. By doing so, we aim to leverage the diversity and heterogeneity of the joint datasets to extract high-level and general-purpose features. These features should be applicable to a wide range of unseen domains considering various users, devices, and sensors. In other words, the pre-trained model is designed to capture the common underlying patterns and characteristics of the human activities that are invariant across datasets, so that it provides a robust feature extractor.

D. Considered Models

In this paper, we consider four alternative state-of-the-art neural networks which we use with their default parameters and implementation from their official source code repositories. For each model, we consider the last layer (softmax classification layer) as *classification head*, while the remaining layers as *feature extractor*.

1) **ISPL Inception** [13] This network has been designed by adapting the Inception-ResNet architecture proposed [41] to the HAR domain. This architecture represents a strong CNN architecture baseline for our study.
2) **DeepConvLSTM** [42] is a combination of convolutional and LSTM recurrent layers. The network architecture has been frequently used as a baseline for state-of-the-art studies in the wearable HAR domain community.
3) **HART** [36] A sensor-wise HAR Transformer (HART) architecture adapted from the successful transformer model from the vision domain [43]. The model incorporates lightweight components, able to be deployed on small devices capable of real-time performance.
4) **MobileHART** [36] An extension of HART that incorporates convolutional layers to learn spatial/temporal inductive biases that conventional transformers do not. The architecture reports state-of-the-art results on multiple benchmark datasets.

TABLE II: Fine-tuning ratios and the corresponding samples of training data for the considered datasets

	ratio of training data used for fine-tuning			
Dataset	**1%**	**5%**	**10%**	**70% (All Train Data)**
HHAR	733	3,680	7,366	51,581
MobiAct	385	1,987	3,991	28,028
MotionSense	153	1,067	2,148	15,092
RealWorld	520	2,634	5,274	36,953
UCI	180	506	1,026	7,241
PAMAP2	148	751	1,504	10,545

IV. Fine-tuning and evaluation

In this section, we outline our methodology for evaluating the pre-trained model that was learned from the data obtained by combining the considered datasets.

A. Fine-tuning Methodologies

A small amount of labeled data acquired on the target domain is used to fine-tune the pre-trained model. For the sake of this work, in this phase, we consider three strategies for using this small amount of data with the different models composed of the pre-trained *feature extractor* augmented with a randomly initialized *classification head*: a) using an unfrozen and randomly initialized *feature extractor* (denoted as Rd) as a baseline, b) using a frozen pre-trained *feature extractor* (P_F) to assess generalization capabilities on new tasks and c) using a pre-trained feature extractor with all layers unfrozen (P_U) to assess the impact of fine-tuning data on feature extraction.

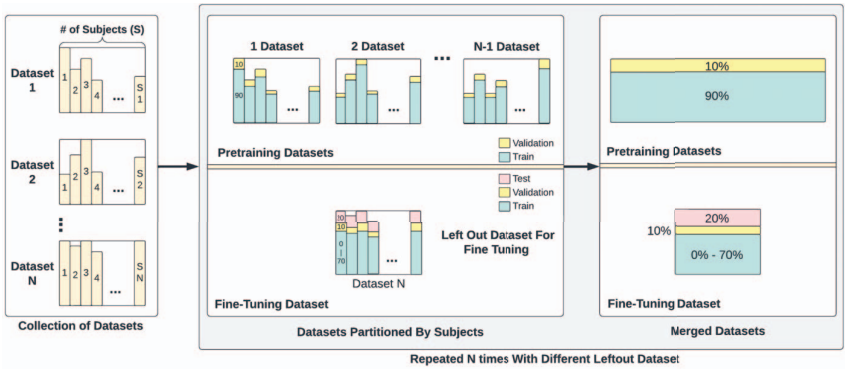


Fig. 2: Overview of LODO

B. Leave-One-Dataset-Out

In order to evaluate the effectiveness of our pre-trained models on unseen data, we introduce a new evaluation strategy called "Leave-One-Dataset-Out" (LODO). The data flow of this approach is depicted in Figure 2. This strategy takes into account our multi-dataset setting and provides a better assessment of the model's ability to transfer knowledge to an unseen target domain. At each fold, one dataset is considered as *left-out dataset*, while the *remaining* ones are used to create a pre-trained model. Specifically, the pre-training phase includes the pre-processing steps depicted in Section III-B For the partitioning, these datasets are class-stratified partitioned in a subject-wise manner, with 90% of the data used for training and 10% for validation. The training data from all remaining datasets are then merged (we will refer to this as the source/pre-training dataset), to learn the pre-trained the model. Similarly, the validation data from all remaining datasets are used as the validation set during the pre-training process. The left-out dataset (which we will refer to as the target dataset) is partitioned subject-wise with 20% set aside for testing and 10% for validation. The remaining data is used to create various training partitions simulating different labeled data scarcity scenarios for fine-tuning. Six ratios of fine-tuning data are considered: 0%, 1%, 5%, 10%, and 70%. The 0% ratio depicts a scenario where there is no data for fine-tuning, and it is useful to assess the capability of the pre-trained model to transfer knowledge to the target dataset. The 1% ratio assesses the model's performance when there is minimal training data. The 5% and 10% ratios demonstrate learning scenarios with limited data availability, while the full 70% ratio represents a conventional training pipeline. Table II shows the approximate amount of samples regarding each fine-tuning ratio for each of the considered datasets. We observed that, in extreme cases, the PAMAP2 and UCI datasets will have respectively less than 3 and 6 minutes of training data considering the 1% ratio.

C. Experimental Setup

Our experiments were conducted on a high-performance computing cluster with Intel Cascade Lake 6248 processors, 192GB of memory, and Nvidia Tesla V100 SXM2 16GB GPUs. All models were developed using TensorFlow [44].

For each experiment, we learn the pre-trained model by using 200 epochs with a batch size of 128 on the combined datasets mentioned in the previous section. We saved checkpoints of the pre-trained model during training, and then selected the one that achieved the highest accuracy on the validation set as the final pre-trained model. In the fine-tuning phase, we adopted 200 epochs with a batch size of 64 on the left-out datasets with their representative training scenarios mentioned in the previous section. The evaluation of the test set was performed on the fine-tuned model that achieved the highest accuracy on the validation set. We adopted Adam as the optimizer with a learning rate of 0.0005 for all experiments. The hyper-parameters of each specific network follow the standard configuration proposed by the authors of the corresponding papers. We used the macro F1 score metric to measure the performance of the classification task.

V. Results

In this section, we present the results of our evaluation using the evaluation methodology proposed in IV-B.

A. Main results

We first considered the worst-case labeled data scarcity scenario, where only 1% of fine-tuning data is available in the target domain. Figure 3 shows the F1 score obtained on each model with the different fine-tuning strategies, by averaging the recognition rate at each fold.

We observe that pre-training is particular effective for transformers-based architectures. For example, the F1 score of the randomly initialised HART model Rd is only 61%, whereas P_U achieves 74% of F1 score. With MobileHART, the benefits of pre-training are even more noticeable, with a 37% F1 score improvement when using P_U compared to Rd. These results suggest that the complexity of transformers-based networks requires a large amount of training data to extract meaningful features, leading to inadequate classification rates with randomly initialised models and limited labeled data in the target domain. Thus, our pre-training method enables transformers to learn high-level and general-purpose features

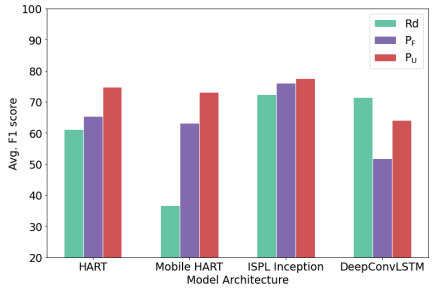


Fig. 3: The average F1-score obtained with the considered model architectures by using 1% of fine-tuning data

that work well on unseen data, which explains why they reach a high F1 score.

Considering ISPL Inception, the pre-trained model P_U outperforms Rd on average by 5% of F1 score. In this case, the advantage of pre-training is particularly evident on some datasets (e.g., $\approx +10\%$ when UCI and PAMAP2 are target datasets) and negligible in others (e.g., $\approx +0.4\%$ on MobiAct). This is likely due to the fact that the network underlying ISPL Inception is less complex than transformers and, in some cases, few training samples from the target domain are sufficient for model convergence. On the other hand, we observed that our pre-training strategy has a negative effect on DeepConvLSTM, where the F1 score obtained by fine-tuning Rd outperforms the one generated with P_U and P_F. This is likely due to the fact that DeepConvLSTM exhibited overfitting behaviors during pre-training, with a negative impact on generalization capabilities on unseen data.

B. More detailed results

In the following, we provide more details about these results, discussing the pros and cons of the proposed pre-training and fine tuning approach with each of the considered model architectures.

1) ISPL Inception: Table III shows the outcomes obtained by using the ISPL Inception with diverse ratios of fine-tuning data. When fine-tuning is not performed (i.e., fine-tuning ratio of 0%), P_U achieves an average F1-score of approximately 0.16 across all target datasets. This result suggests that, despite pre-training the model with a large number of diverse datasets, this model generally struggles to generalize well on data coming from unseen datasets.

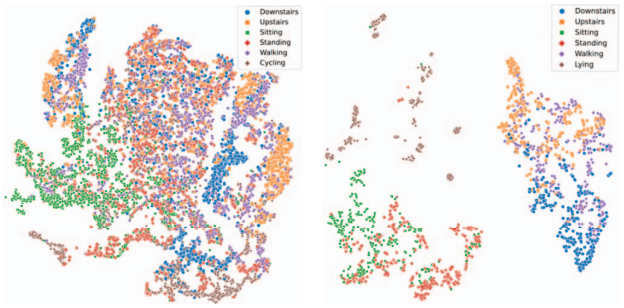
Nonetheless, as we previously mentioned, the effectiveness of pre-training varies depending on the specific of the target domain dataset. For example, the F1 score for HHAR and MobiAct with 0% fine-tuning is below 10%, while for RealWorld and UCI, it exceeds 20%. This observation is confirmed by analysing the T-SNE visual representation of the embeddings in Figure 4. Here, the embeddings from HHAR are cluttered and poorly separated, while those for UCI exhibit clear separations between static and dynamic activities. As previously discussed, the ISPL Inception model has the ability to quickly learn meaningful features from a small amount of

TABLE III: ISPL Inception evaluated using LODO

	ft ratio												
	0%	**1%**			5%			10%			All Train		
Target dataset	P_U	Rd	P_F	P_U	Rd	P_F	P_U	Rd	P_F	P_U	Rd	P_F	P_U
HHAR	09.00	80.59	83.33	86.12	91.26	92.04	92.39	93.24	93.25	94.20	95.70	95.28	95.34
MobiAct	06.67	57.46	59.09	57.83	85.50	83.67	83.98	87.68	88.09	88.28	89.63	90.32	88.53
MotionSense	12.67	83.46	83.25	84.14	95.34	95.9	95.58	96.78	96.73	97.36	98.06	98.13	98.07
RealWorld	21.26	81.98	85.25	85.9	87.05	88.17	88.35	88.15	89.72	88.99	91.02	91.31	91.11
UCI	27.41	76.06	86.86	87.13	91.14	94.20	94.08	92.58	95.81	95.42	97.45	98.14	97.76
PAMAP2	19.61	54.99	58.76	64.07	72.67	72.55	71.73	76.21	74.64	74.21	74.09	74.16	74.17
AVG	16.10	72.42	76.09	77.53	87.16	87.76	87.69	89.11	89.71	89.74	90.99	91.22	90.83

target data, which leads to the recognition rate significantly improve when only 1% of fine-tuning data is used. Similarly to what we observed with 0%, also in this case the advantage of pre-training is significant depending on the specific target dataset.

As the fine-tuning ratio increases, the benefits of pre-training decrease accordingly, confirming that pre-training is particularly important for labeled data scarcity scenarios.



(a) HHAR (b) UCI

Fig. 4: T-SNE visualization of the pre-trained ISPL Inception model's embeddings on HHAR and UCI datasets. The HHAR embeddings are mixed and cluttered, while the ones from UCI show a coarse separation among classes.

2) Transformer-based models: Table IV shows the results achieved with HART, while Table V illustrates the outcomes generated by MobileHART. Overall, the results are consistent with those obtained with the ISPL Inception network, but some important differences should be noted.

Firstly, the F1 scores achieved with HART and MobileHART by fine-tuning the randomly initialized model Rd with a small percentage of data (i.e., 1% and 5%) are generally lower than those obtained with the ISPL Inception network. This can be attributed to the higher complexity of the transformer architecture, which requires more training samples to extract representative features from the target dataset.

On the other hand, the performances in terms of F1 score obtained by fine-tuning P_U and P_F in these scenarios are significantly better than the F1 score achieved with Rd.

Furthermore, the advantages of pre-training are noticeable even when using 5% and 10% of the fine-tuning data.

3) DeepConvLSTM: Table VII shows that, in general, DeepConvLSTM does not benefit from pre-training contrary

TABLE IV: HART evaluated using LODO

	ft ratio												
	0%	1%			5%			10%			All Train		
Target dataset	P_U	Rd	P_F	P_U	Rd	P_F	P_U	Rd	P_F	P_U	Rd	P_F	P_U
HHAR	08.11	65.28	64.72	66.55	83.48	83.69	84.66	89.82	88.16	89.42	93.30	92.75	93.86
MobiAct	07.81	43.70	50.56	67.99	77.69	80.40	80.05	83.16	82.86	83.78	87.67	87.57	87.33
MotionSense	14.04	70.91	74.23	87.93	91.66	92.80	93.17	93.56	94.88	95.49	97.45	97.92	97.84
RealWorld	32.68	77.08	75.38	85.75	85.83	87.74	87.89	87.45	88.99	89.33	91.30	91.81	91.64
UCI	26.69	61.00	70.22	84.46	88.83	79.82	93.27	90.20	95.32	95.06	96.40	96.70	96.74
PAMAP2	15.39	49.17	57.82	56.31	63.04	64.82	64.62	64.42	66.82	66.16	69.10	68.13	67.91
AVG	17.45	61.19	65.49	74.83	81.76	81.55	83.94	84.77	86.17	86.54	89.20	89.15	89.22

TABLE V: MobileHART evaluated using LODO

	ft ratio												
	0%	1%			5%			10%			All Train		
Target dataset	P_U	Rd	P_F	P_U	Rd	P_F	P_U	Rd	P_F	P_U	Rd	P_F	P_U
HHAR	06.27	78.02	60.18	71.03	92.76	91.26	92.05	94.9	93.99	94.06	97.31	97.07	96.63
MobiAct	06.84	47.05	44.42	53.50	82.47	75.38	80.93	86.53	84.29	85.29	89.19	89.22	90.18
MotionSense	16.26	10.92	68.34	85.92	92.39	92.18	93.06	95.85	96.44	96.89	98.26	97.82	98.16
RealWorld	33.52	66.97	83.61	84.99	85.23	87.42	87.60	88.64	89.21	89.18	91.30	91.26	91.13
UCI	20.65	13.44	67.25	84.65	88.64	92.46	94.18	90.76	92.95	95.17	97.44	97.40	98.24
PAMAP2	12.58	04.25	55.7	58.53	60.03	65.73	68.48	65.01	67.70	70.34	71.10	71.89	73.98
AVG	16.02	36.78	63.25	73.10	83.59	84.07	86.05	86.95	87.43	88.49	90.77	90.78	91.39

to the other model architectures considered in this study. Specifically, the randomly initialized model Rd outperformed the pre-trained models P_U and P_F when a limited percentage of fine-tuning data was used (i.e., 1% and 5%). This behavior may be due to the fact that the DeepConvLSTM has significantly fewer parameters compared to the other networks (see Table VI for details), and therefore struggles to extract high-level features that generalize well over heterogeneous datasets. Nonetheless, there is only one case where our pre-training

TABLE VI: Models with their parameters and FLOPS

Model	**Parameters**	**FLOPS**
ISPL Inception [13]	1,327,714	338,528,096
DeepConvLSTM [42]	457,546	86,184,508
HART [36]	1,445,918	15,212,636
MobileHART [36]	2,542,942	19,809,292

strategy is effective for DeepConvLSTM, and that is when UCI is used as a target dataset. For these reasons, DeepConvLSTM resulted in not a particularly suitable model for the considered transfer-learning task.

VI. Discussion

The beneficial impact of the pre-training varies across different architectures and target datasets. This show that it is important to evaluate pre-training on different datasets and different types of models. Transformer-based architectures such as HART and MobileHART benefit more from the pre-training as opposed to smaller networks such as the DeepConvLSTM, which obtained ranging benefits from little to worsening performances when compared to starting randomly instead. As transformer-based networks generally require more training and data to start converging to superior performances, we argue that model initializing on other datasets is essential for this type of architecture. Contrarily, smaller networks are more prone to overfitting when starting with an initialized model that has been trained on a large amount of data.

In terms of datasets, reported performances vary. Fine-tuning the pre-trained model using datasets such as MotionSense, RealWorld, UCI, and PAMAP2 generally showed improved recognition rates. However, when the target dataset is too different from the other combined datasets model, the advantage of pre-training may be limited. For instance, considering the HHAR and MobiAct datasets, our results show that it is particularly challenging to build a pre-trained model capable of generalizing on them. These results are consistent considering all the neural networks by observing the results of transfer without any fine-tuning. This is likely due to the numerous devices used and the abundant number of users of the HHAR and MobiAct datasets respectively.

Finally, the results obtained by fine-tuning the three best model using a frozen feature extractor (P_F) are similar to the ones obtained with an entirely fine-tuned pre-trained model (P_U). This suggests that the pre-trained feature extractor has learned representations that are sufficiently robust to reach good performances simply by optimizing a classification head. Such results are particularly relevant considering training efficiency since the frozen-feature extractor scenario drastically lowers training costs in the fine-tuning stage.

TABLE VII: DeepConvLSTM evaluated using LODO

	ft ratio												
	0%	1%			5%			10%			All Train		
Target dataset	P_U	Rd	P_F	P_U	Rd	P_F	P_U	Rd	P_F	P_U	Rd	P_F	P_U
HHAR	05.58	80.37	60.00	66.26	83.28	79.03	82.12	86.15	87.71	89.42	88.48	94.04	94.87
MobiAct	06.55	51.94	28.21	42.14	85.46	72.38	77.71	85.46	79.79	81.11	88.72	87.47	87.62
MotionSense	14.71	87.67	42.67	67.54	95.60	76.6	91.64	96.71	94.60	95.26	98.17	98.01	98.20
RealWorld	28.07	84.64	66.93	77.39	88.13	86.22	87.62	89.50	89.48	89.39	91.28	91.28	91.52
UCI	37.13	68.88	64.65	79.43	89.81	76.81	91.05	92.29	93.3	95.52	97.53	96.50	97.20
PAMAP2	20.16	55.47	47.81	52.35	68.69	63.54	64.56	66.36	66.93	67.21	71.84	68.95	68.91
AVG	18.70	71.50	51.71	64.19	85.16	75.76	82.45	86.08	85.30	86.32	89.34	89.38	89.72

VII. Conclusion

This paper introduces a novel strategy to mitigate the labeled data scarcity problem for sensor-based HAR, that relies on combining public datasets. By proposing a new evaluation methodology, our results show that pre-training on multiple datasets greatly improves performance in labeled data scarcity scenarios. In future work, we aim at improving the pre-training mechanism by learning more robust features. Specifically, we intend to investigate self-supervised learning methods. Indeed, several studies in other domains (e.g. Computer Vision, NLP) have already shown that the trained model is able to learn task-agnostic features [45].

VIII. Acknowledgement

This work has been partially funded by Naval Group, by MIAI@Grenoble Alpes (ANR-19-P3IA-0003), and granted access to the HPC resources of IDRIS under the allocation 2023-AD011013233R1 made by GENCI.

Part of this research was also supported by projects SERICS (PE00000014) and by project MUSA – Multilayered Urban Sustainability Action, funded by the European Union – NextGenerationEU, under the National Recovery and Resilience Plan (NRRP) Mission 4 Component 2 Investment Line 1.5: Strengthening of research structures and creation of R&D "innovation ecosystems", set up of "territorial leaders in R&D".

REFERENCES

[1] F. Gu, M.-H. Chung, M. Chignell, S. Valaee, B. Zhou, and X. Liu, "A survey on deep learning for human activity recognition," *ACM Computing Surveys (CSUR)*, vol. 54, no. 8, pp. 1–34, 2021.

[2] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *Pattern recognition letters*, vol. 119, pp. 3–11, 2019.

[3] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, pp. 211–252, 2015.

[4] Evain *et* al., "Task Agnostic and Task Specific Self-Supervised Learning from Speech with LeBenchmark," in *NeurIPS 2021*, 2021.

[5] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, and J. Tang, "Self-supervised learning: Generative or contrastive," *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[6] A. Stisen, H. Blunck, S. Bhattacharya, T. S. Prentow, M. B. Kjærgaard, A. Dey, T. Sonne, and M. M. Jensen, "Smart devices are different: Assessing and mitigating mobile sensing heterogeneities for activity recognition," in *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, (New York, NY, USA), p. 127–140, 2015.

[7] G. M. Weiss and J. Lockhart, "The impact of personalization on smartphone-based activity recognition," in *Workshops at the Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.

[8] G. Vavoulas, C. Chatzaki, T. Malliotakis, M. Pediaditis, and M. Tsiknakis, "The mobiact dataset: Recognition of activities of daily living using smartphones.," in *ICT4AgeingWell*, pp. 143–151, 2016.

[9] M. Malekzadeh, R. G. Clegg, A. Cavallaro, and H. Haddadi, "Protecting sensory data against sensitive inferences," in *Proceedings of the 1st Workshop on Privacy by Design in Distributed Systems*, W-P2DS'18, (New York, NY, USA), pp. 2:1–2:6, ACM, 2018.

[10] A. Reiss and D. Stricker, "Introducing a new benchmarked dataset for activity monitoring," in *2012 16th International Symposium on Wearable Computers*, pp. 108–109, IEEE, 2012.

[11] Y. Vaizman, K. Ellis, and G. Lanckriet, "Recognizing detailed human context in the wild from smartphones and smartwatches," *IEEE pervasive computing*, vol. 16, no. 4, pp. 62–74, 2017.

[12] X. Qin, Y. Chen, J. Wang, and C. Yu, "Cross-dataset activity recognition via adaptive spatial-temporal transfer learning," *in ACM IMWUT 2019*, vol. 3, no. 4, pp. 1–25, 2019.

[13] M. Ronald, A. Poulose, and D. S. Han, "isplinception: An inception-resnet deep learning architecture for human activity recognition," *IEEE Access*, vol. 9, pp. 68985–69001, 2021.

[14] C. Jobanputra, J. Bavishi, and N. Doshi, "Human activity recognition: A survey," *Procedia Computer Science*, vol. 155, pp. 698–703, 2019.

[15] D. J. Cook, K. D. Feuz, and N. C. Krishnan, "Transfer learning for activity recognition: A survey," *Knowl. Inf. Sys.*, vol. 36, no. 3, 2013.

[16] A. Farahani, S. Voghoei, K. Rasheed, and H. R. Arabnia, "A brief review of domain adaptation," *Advances in Data Science and Information Engineering: Proceedings from ICDATA 2020 and IKE 2020*, 2021.

[17] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, "A comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2020.

[18] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," *arXiv preprint arXiv:2212.04356*, 2022.

[19] D. Cook, K. D. Feuz, and N. C. Krishnan, "Transfer learning for activity recognition: A survey," *Knowl. Inf. Sys.*, 2013.

[20] J. Wang, V. W. Zheng, Y. Chen, and M. Huang, "Deep transfer learning for cross-domain activity recognition," in *proceedings of the 3rd International Conference on Crowd Science and Engineering*, 2018.

[21] F. J. O. Morales and D. Roggen, "Deep convolutional feature transfer across mobile activity recognition domains, sensor modalities and locations," in *Proceedings of the 2016 ACM International Symposium on Wearable Computers*, pp. 92–99, 2016.

[22] M. Kurz, G. Hölzl, A. Ferscha, A. Calatroni, D. Roggen, and G. Tröster, "Real-time transfer and evaluation of activity recognition capabilities in an opportunistic system," *machine learning*, vol. 1, no. 7, p. 8, 2011.

[23] Y. Chen, J. Wang, M. Huang, and H. Yu, "Cross-position activity recognition with stratified transfer learning," *Pervasive and Mobile Computing*, vol. 57, pp. 1–13, 2019.

[24] C.-H. Lu, Y.-C. Ho, Y.-H. Chen, and L.-C. Fu, "Hybrid user-assisted incremental model adaptation for activity recognition in a dynamic smart-home environment," *IEEE Transactions on Human-Machine Systems*, vol. 43, no. 5, pp. 421–436, 2013.

[25] M. Gjoreski, S. Kalabakov, M. Luštrek, M. Gams, and H. Gjoreski, "Cross-dataset deep transfer learning for activity recognition," in *in adjunct ACM UbiComp 2019 and ACM ISWC 2019*, pp. 714–718, 2019.

[26] M. A. A. H. Khan, N. Roy, and A. Misra, "Scaling human activity recognition via deep learning-based domain adaptation," in *2018 IEEE international conference on pervasive computing and communications (PerCom)*, pp. 1–9, IEEE, 2018.

[27] A. R. Sanabria and J. Ye, "Unsupervised domain adaptation for activity recognition across heterogeneous datasets," *Pervasive and Mobile Computing*, vol. 64, p. 101147, 2020.

[28] Z. Zhou, Y. Zhang, X. Yu, P. Yang, X.-Y. Li, J. Zhao, and H. Zhou, "Xhar: Deep domain adaptation for human activity recognition with smart devices," in *2020 17th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, IEEE, 2020.

[29] J. Wang, Y. Chen, Y. Gu, Y. Xiao, and H. Pan, "Sensorygans: An effective generative adversarial framework for sensor-based human activity recognition," in *2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, IEEE, 2018.

[30] M. H. Chan and M. H. M. Noor, "A unified generative model using generative adversarial network for activity recognition," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–10, 2020.

[31] E. Soleimani and E. Nazerfard, "Cross-subject transfer learning in human activity recognition systems using generative adversarial networks," *Neurocomputing*, vol. 426, pp. 26–34, 2021.

[32] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A survey on contrastive self-supervised learning," *Technologies*, 2020.

[33] C. I. Tang, I. Perez-Pozuelo, D. Spathis, S. Brage, N. Wareham, and C. Mascolo, "Selfhar: Improving human activity recognition through self-training with unlabeled data," *arXiv preprint arXiv:2102.06073*, 2021.

[34] H. Haresamudram, I. Essa, and T. Plötz, "Assessing the state of self-supervised human activity recognition using wearables," *arXiv preprint arXiv:2202.12938*, 2022.

[35] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in neural information processing systems*, 2020.

[36] S. EK, F. Portet, and P. Lalanda, "Lightweight transformers for human activity recognition on mobile devices," *arXiv preprint arXiv:2209.11750*, 2022.

[37] T. Sztyler and H. Stuckenschmidt, "On-body localization of wearable devices: An investigation of position-aware activity recognition," in *2016 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pp. 1–9, 2016.

[38] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones," in *21st European Symposium on Artificial Neural Networks, ESANN 2013, Bruges, Belgium, April 24-26, 2013*, 2013.

[39] W. Sousa Lima, E. Souto, K. El-Khatib, R. Jalali, and J. Gama, "Human activity recognition using inertial sensors in a smartphone: An overview," *Sensors*, vol. 19, p. 3213, 07 2019.

[40] A. Ignatov, "Real-time human activity recognition from accelerometer data using convolutional neural networks," *Applied Soft Computing*, vol. 62, pp. 915–922, 2018.

[41] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proceedings of the AAAI conference on artificial intelligence*, 2017.

[42] F. J. Ordóñez and D. Roggen, "Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, p. 115, 2016.

[43] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[44] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[45] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, and J. Tang, "Self-supervised learning: Generative or contrastive," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 1, pp. 857–876, 2023.