# Large-Scale Pre-Training for Dual-Accelerometer Human Activity Recognition

Aleksej Logacjov[1][0000−0002−8834−1744], Sverre Herland[1], Astrid Ustad[2][0000−0001−7516−3259], and Kerstin Bach[1][0000−0002−4256−7676]

[1] Department of Computer Science, Norwegian University of Science and Technology, Trondheim Trøndelag 7034, Norway
{aleksej.logacjov,sverre.herland,kerstin.bach}@ntnu.no
[2] Department of Neuromedicine and Movement Science, Norwegian University of Science and Technology, Trondheim Trøndelag 7030, Norway
astrid.ustad@ntnu.no

**Abstract.** The annotation of physical activity data collected with accelerometers for human activity recognition (HAR) remains challenging despite the growing interest in large public health studies. Existing free-living accelerometer-based datasets are limited, hindering the training of effective deep learning models. To address this limitation, some studies have explored self-supervised learning (SSL), i.e., training models on both labeled and unlabeled data. Here, we extend previous work by evaluating whether large-scale pre-training improves downstream HAR performance. We introduce the SelfPAB method, which includes pre-training a transformer encoder network on increasing amounts of accelerometer data (10-100K hours) using a reconstruction objective to predict missing data segments in the spectrogram representations. Experiments demonstrate improved downstream HAR performance using SelfPAB compared to purely supervised baseline methods on two publicly available datasets (HARTH and HAR70+). Furthermore, an increase in the amount of pre-training data yields higher overall downstream performance. SelfPAB achieves an F1-score of 81.3% (HARTH), and 78.5% (HAR70+) compared to the baselines' F1-scores of 74.2% (HARTH) and 63.7% (HAR70+). Additionally, SelfPAB leads to a performance increase for activities with little training data.

**Keywords:** accelerometer· human activity recognition· self-supervised learning· machine learning· transformer.

## 1 Introduction

Accelerometer-based human activity recognition (HAR) is a research field focusing on predicting human postures and physical activities from accelerometer data [28]. Supervised machine learning (ML) is one of the most successful techniques to facilitate accelerometer-based HAR due to its ability to learn complex patterns in the data [24]. More recently, self-supervised learning (SSL), a form of semi-supervised learning, gained much attention in the ML community due to

its ability to extract useful representations from unlabeled data [11], thus omitting the costly task of annotating large-scale datasets. In general, SSL consists of two steps. First, a model is pre-trained on unlabeled data by defining an objective (auxiliary task) the model has to solve (upstream training). Second, the learned representations of the upstream training are leveraged to solve tasks that rely on annotated data (downstream training), like HAR [19]. The main goal is to improve the downstream performance, compared to purely supervised learning (i.e., no upstream training), based on the representations learned through self-supervised pre-training. SSL achieved state-of-the-art performances in many research fields [33,6,15]. But also the HAR research community started to investigate different SSL approaches (see Section 2). However, most works in SSL-based HAR use small labeled datasets for both upstream and downstream training. It has been shown in the research field of natural language processing, especially in large-language models, that the amount of training data plays a crucial role in a neural network's performance, with more data leading to better results [12]. None of the existing SSL-based HAR literature investigates large-scale datasets for pre-training and their influence on HAR performance. We fill this gap by making the following contributions:

**1)** We implement a self-supervised physical activity behavior representation learning method (SelfPAB). We pre-train a transformer encoder network [30] on the large-scale, unlabeled HUNT4 data corpus. The auxiliary task during pre-training is to reconstruct masked time windows and frequency bands in six spectrograms. The pre-trained network is used as a feature extractor during downstream HAR training on the two labeled HAR datasets, HARTH [18] and HAR70+ [29]. **2)** We experiment with different amounts of unlabeled data for pre-training. In particular, 10 hours, 100 hours, 1k hours, 10k hours, and 100k hours of the HUNT4 dataset. We show that only 10 hours of acceleration signals, less than many supervised datasets contain, are sufficient to achieve similar (on HARTH) and higher (on HAR70+) performances than purely-supervised methods. Using 100 hours shows better results in both datasets. Our experiments indicate that the amount of hours used for pre-training scales with the downstream performance. **3)** We show that especially the performance of activities with little data benefits from pre-training. Our experiments and pre-trained models are publicly available[3].

## 2 Related Work

There are various related works that have investigated different SSL strategies for HAR. Those can be grouped into three categories:

1) **Multi-task self-supervision**: In this category, multiple auxiliary tasks are defined at once. Related works focused on transformation-based multi-task self-supervision, hence, identifying what kind of transformation(s), if any, is applied to the input signal [22,26].

[3] `https://github.com/ntnu-ai-lab/SelfPAB` (accessed on 2023-10-16)

2) **Contrastive learning**: In contrastive learning, input representations are learned through comparing input samples [14]. The representations of "similar" samples (positive samples) need to be closer together than the representations of "dissimilar" samples (negative samples)[14]. How "similar" / "dissimilar" and the distance are defined depends on the used algorithm. Different contrastive approaches are proposed in [8,27,17,23,13,32,10,31].

3) **Masked reconstruction**: In masked reconstruction-based SSL, parts of the input signals are masked out (e.g., replaced with zeros), and the pre-training objective is the reconstruction of these parts to learn local temporal dependencies [9]. Related works focus on time-domain masked reconstruction only [7,25].

The authors of [9] made an in-depth investigation of the state-of-the-art in SSL-based HAR. They used the Capture-24 dataset [2] to pre-train various conceptually different SSL approaches. The mentioned related works for masked reconstruction-based SSL show some limitations. First, they consider only a single sensor even though many studies show that using more than one can increase the HAR performance [5,20]. Second, the former two works ([7,25]) use the labeled HAR datasets for both pre-training and downstream training. Due to the datasets' limited size, this aspect makes it difficult to investigate whether more pre-training data can improve the HAR results. The authors of [9] studied different pre-training data quantities with the unlabeled Capture-24 dataset. However, they focused on only one sensor, and Capture-24 has its limitations of 4000 hours and 151 participants.

## 3 Methods

The self-supervised physical activity behavior representation learning method (SelfPAB), used in this work, is illustrated in Figure 1. It is based on TERA [15], a speech representation learning technique. SelfPAB consists of two parts, an upstream (left) and a downstream part (right). First, an upstream network is pre-trained on unlabeled data to acquire potentially useful physical activity representations. Second, the resulting model is utilized as a feature extractor for downstream training (e.g., HAR). The goal is to improve the performance of a downstream model by leveraging the representations the upstream model acquires from the unlabeled data.

### 3.1 Upstream

**Acceleration Signals** We use three existing datasets (see Section 4), each recorded with two Axivity AX3 (Axivity Ltd., Newcastle, UK) [4] accelerometers attached to the participants' lower back and thigh. Each sensor records the acceleration in three spatial dimensions, resulting in six time signals. We compute spectrograms of each signal using the short-time Fourier transform (STFT) to get the frequency content over time. This is inspired by the research field of automatic speech recognition (ASR), where spectrograms are successfully utilized

[4] `http://www.axivity.com/` (accessed on 2021-06-29)

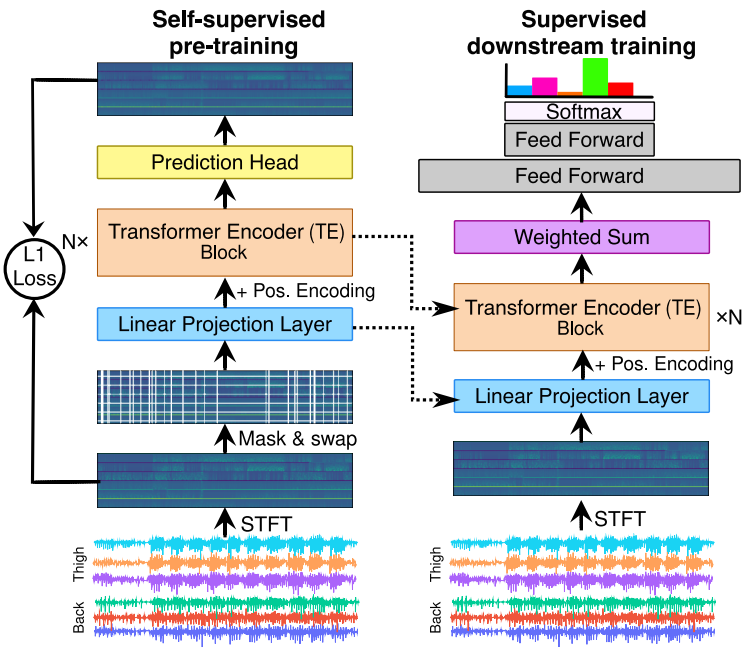



Fig. 1: Illustration of the SelfPAB method, consisting of two parts, the self-supervised pre-training (left) and the supervised downstream training (right).

for pre-training instead of raw time signals [15,16]. We stack the six resulting spectrograms on top of each other to create vectors for each time frame.

**Signal Alteration and Auxiliary Task** We utilize the masked reconstruction auxiliary task as it allows learning of temporal dependencies without much effort [9], which leads to useful representations of accelerometer signals for downstream tasks. Another benefit of masked reconstruction is that, compared to contrastive approaches, it does not suffer from the so-called sampling bias [4]. The primary strategy is to mask certain parts of the input and let the model learn to reconstruct these parts using the unmasked parts. As a result, we perform two alteration techniques on the input spectrograms.

1) Time domain alteration: As in [15], we define a time alteration percentage $P_T$, which determines the maximal amount of time frames to be altered in all six spectrograms. First, a number $T_{num} = \lfloor \frac{P_T \cdot L_T}{W_T} \rfloor$ of start indices are randomly chosen without replacement. $L_T$ is the total number of input time frames, and $W_T$ is a predefined window width to be altered. With a probability of 80%, the selected frames are replaced with zeros, with a probability of 10%, they are swapped with other frames in the input, and with a probability of 10%, they are not altered at all. The authors of [15] argue that the latter case tackles the train-test inconsistency problem. The white vertical lines in the spectrograms of Figure 1 illustrate the masking of time frames with zeros in all six spec-

trograms. Note that altered windows can overlap, leading to larger consecutive masked/swapped areas.

2) Frequency domain masking: Like in time domain masking, we compute a number of start indices $F_{num} = \lfloor \frac{P_F \cdot L_F}{W_F} \rfloor$ using frequency masking percentage $P_F$, frequency window width $W_F$, and the number of frequency bins of one sensor $L_F$. The same consecutive frequency bands are masked (zeroed out) in all six spectrograms. The white horizontal lines in the spectrograms of Figure 1 illustrate this masking.

Note that time domain alteration and frequency domain masking are combined in most cases. Like in TERA [15], we use the L1 reconstruction loss $l_1 = M \cdot |y - \hat{y}|$ between the upstream model's output $\hat{y}$ and the unmasked spectrograms $y$. M is a matrix with the same dimension as $y$ and $\hat{y}$ and contains ones where alteration (masking or swapping) is applied and zeros anywhere else. The multiplication with M ensures that the L1 loss is computed for the altered parts only. Initial experiments with the L2 loss and the Huber loss showed no benefits to the L1 loss.

**Upstream Architecture** The masked spectrograms are forwarded to a linear input projection layer. It is a single trainable feed-forward layer, mapping the input to a predefined embedding of dimension $d_{model}$. Sinusoidal positional encoding is used to preserve information about the order of the input sequence. Input embedding and positional encoding are summed together and forwarded to a transformer encoder network consisting of $N$ transformer encoder layers. Transformer models, proposed in [30], can learn relationships between a set of input vectors, in our case, between time windows in the stacked spectrograms, without the usage of recurrent or convolutional layers, making them efficient to train. The output of the last transformer encoder layer is forwarded to the prediction head, a feed-forward layer mapping the $d_{model}$-dimensional vectors back to the input dimension $d_{input}$ to make the model's output comparable to the unmasked spectrogram. Despite the similarity of SelfPAB to TERA [15], we want to highlight the differences here. 1) We work with standard spectrograms in contrast to log Mel spectrograms. 2) We consider a six-dimensional time series (two sensors, each having three axes) instead of a univariate time series. 3) The authors of TERA applied magnitude alteration by adding noise to the spectrograms, which we do not. The reason is that it did not provide a strong benefit in classification tasks [15].

### 3.2 Downstream

We use the pre-trained linear input projection layer and the transformer encoder upstream network to extract features from the input spectrograms. Like in [15], we use the weighted sum $\mathbf{F} = \sum_{l=1}^{L} \mathbf{F}_l \cdot w_l$ of each transformer encoder layer's output $F_l$ as input to the downstream network. $L$ is the number of transformer encoder layers and $w_l$ a trainable weight scalar. This technique allows the model to learn which layer in the upstream network is most important for the

downstream training. It is inspired by the authors of [3], who showed that using internal transformer encoder layers for feature extraction can lead to better speaker recognition and phoneme classification results.

The downstream architecture is a multilayer perceptron (MLP) with one hidden layer. It receives the upstream model's $d_{model}$-dimensional output as input. The output layer has the same dimension as the number of activities in the HAR downstream dataset. A ReLU activation is applied to the hidden layer's output and a softmax activation function to the output layer. Initially, the weights of the upstream model are frozen, and only the weighted-sum layer and downstream MLP are trained to prevent the initially large gradients from altering the carefully set parameters of the upstream model too much. However, we unfreeze the upstream model's weights after 75% of the total number of downstream steps, i.e., we perform fine-tuning. Fine-tuning upstream models showed promising downstream results in other works [15,3].

## 4 Experiments

We test our approach in experiments with three different datasets, the HUNT4 [1], the HARTH v1.2 [18], and the HAR70+ [29]. HUNT4 is an unlabeled dataset, and it is utilized for pre-training only. After pre-training, we investigate the HAR performance on the latter two datasets, which are both labeled.

### 4.1 Pre-training / Upstream

**HUNT4 Dataset (unlabeled)** In HUNT4, accelerometer data of approximately 35,000 participants were recorded [1]. Each participant wore two three-axial Axivity AX3 accelerometers for up to seven days. The sensors were attached to the participants' lower back and thigh, and recordings were made with a sampling rate of $50Hz$. HUNT4 consists of around 230 times more subjects with significantly more hours of data than the Capture-24 dataset [2]. Hence, it is a good candidate to investigate a large variety of hours used for pre-training.

**Data Pre-processing** Five-minute time windows ($15,000$ samples at $50Hz$), a frame length of $1sec$ ($= 50$ samples), an overlap of half a second ($= 25$ samples), and the Hann window function are used for STFT computation. This results in 26 frequency bins and 599 time frames for each axis. The six sensor spectrograms are stacked, resulting in $156 \times 599$-dimensional input matrices. We use the upstream dataset's mean and variance to normalize the input before pre-training.

**Hyperparameters** After initial experiments, the following hyperparameter assignments achieved the best results during pre-training. The linear projection layer transforms the input of dimension $d_{input} = 156$ to $d_{model} = 1500$. The transformer encoder network consists of four transformer encoder layers each having six attention heads, and a 2048-dimensional feed-forward layer. We use

AdamW with a weight decay factor of $1e^{-5}$ as the optimizer. Like in TERA [15], we perform a linear learning rate warm-up in the first 7% of training steps, leading to a peak learning rate of $1e^{-4}$. Afterward, a linear learning rate decay is applied with a final learning rate of $1e^{-6}$. We further compare the downstream performance when using upstream models trained on 10, 100, 1k, 10k, and 100k hours of acceleration data. To ensure that all five models take the same number of gradient steps, we train the 10 hours model for 500,000, the 100 hours model for 50,000, the 1k hours model for 5,000, the 10k hours model for 500, and the 100k hours one for 50 epochs, all with a batch size of 64. We randomly select five-minute time windows of the HUNT4 data corpus to collect the required amount of data. For creating the altered time frames, we define a time alteration percentage of $P_T = 0.15$ and the amount of consecutive time frames to alter is set to $W_T = 3$. We set the frequency masking percentage to $P_F = 0.2$ and the frequency masking width to $W_F = 3$.

### 4.2 Downstream

**Datasets (labeled)** This work considers two publicly available and labeled datasets for downstream training (i.e., HAR). Those are, to the best of our knowledge, the only two labeled and publicly available HAR datasets with the same sensor setup as HUNT4.

1) HARTH v1.2: The first is the HARTH v1.2 [18][5]. Twenty-two subjects were recorded for around 1.5 to 2 hours in a free-living setting. HARTH v1.2 has twelve different professionally annotated activities: walking, running, shuffling (i.e., standing with leg movement), stairs (ascending), stairs (descending), standing, sitting, lying, cycling (sit), cycling (stand), cycling (sit, inactive), and cycling (stand, inactive). The dataset contains around 2221.6 min ($\approx$ 37 hours) of acceleration data. HARTH v1.2 is highly imbalanced, making HAR a challenging task for ML approaches [18]. We combine the active and inactive cycling activities, resulting in ten activities, to make our experiments more comparable to the original HARTH experiments [18].

2) HAR70+: The HAR70+ contains 18 subjects, which are over 70 years old [29][6]. Seven activities were professionally annotated: standing, shuffling, walking, sitting, lying, stairs (descending), and stairs (ascending). HAR70+ consists of around 756 min (= 12.6 hours) accelerometer recordings. As in HARTH v1.2, a high class imbalance is observable [29].

**Downstream Training** We investigate three different settings to show the benefits of our two-stage approach. (1) SelfPAB: Our proposed downstream training as described in Section 3.2. The downstream MLP's hidden layer has a dimension of 1028 and the output layer a dimension of 10 and 7 depending on the used

[5] Dataset available at `https://github.com/ntnu-ai-lab/harth-ml-experiments/tree/v1.2/harth` (accessed on 2022-04-13)

[6] `https://github.com/ntnu-ai-lab/harth-ml-experiments/tree/main/har70plus` (accessed on 2023-03-24)

dataset's number of activities. (2) Spectrograms + MLP: We skip the pre-trained model and train the mentioned MLP directly on the stacked spectrograms. The same MLP hyperparameters as in setting (1) are used. (3) Spectrograms + TE: We train the upstream architecture (linear projection layer, transformer encoder, and prediction head) purely-supervised, i.e., without pre-training. The prediction head has a dimension of 10 or 7, depending on the used dataset. A softmax activation follows the prediction head. The remaining hyperparameters are the same as described in Section 4.1. Settings (2) and (3) will answer the question of whether the pre-training objective in combination with the proposed architecture is helpful for HAR or not. Since we normalize the HUNT4 data during pre-training, we do the same for HARTH v1.2 and HAR70+ using HUNT4's mean and variance before downstream training. This strategy showed a considerable performance improvement in previous work [9]. The first 20% of each subject's spectrogram is used as the validation set and the remaining 80% as the training set, leading to roughly the same activity distribution between the two sets. We randomly cut 32 (batch size) five-minute spectrograms (599 time bins) out of the training set in each training step. The models are trained on 2000 steps in total. We utilize the Adam optimizer, a learning rate of $1e^{-4}$, and exponential learning rate decay with a decay factor of 0.1. The categorical cross-entropy is used as the loss function. A leave-one-subject-out cross-validation (LOSO) is performed. Hence, the model is trained on $S-1$ subjects and tested on 1, with $S$ being the number of subjects. This is repeated $S$ times, each time with a different test subject. Averaged across all activities, we compute the harmonic mean of recall and precision, the average F1-score $= 2 \cdot \frac{Recall \cdot Precision}{Recall + Precision}$ for each test subject. In contrast to the accuracy, the F1-score takes class imbalances into account. Note that we create one-second predictions for five minutes (599 samples) at once. To make the results comparable, we replicate the resulting one-second predictions to 50 samples per second, hence, back to the original time domain dimension.

**Baselines** We compare our method to the best baseline approaches presented in [18], a support vector machine (SVM) and an extreme gradient boost (XGB). Additionally, we compare SelfPAB to the well-established DeepConvLSTM approach [21]. We ensure a fair comparison by performing a hyperparameter optimization with following LOSO for each baseline method. For the XGB and SVM, we compute the same 161 features of five-second time frames the authors in [18] used for training. Similar to the downstream experiments, we replicate these five-second predictions to 50 samples per second, ensuring comparability. We investigate DeepConvLSTM's performance on the raw time signals, as well as spectrograms, denoted as (TS) and (Spectr.), respectively.

## 5 Results

### 5.1 Overall Downstream Performance

The average F1-scores, together with the corresponding standard errors, of the HARTH v1.2 and HAR70+ LOSOs are shown in Table 1. The first four rows

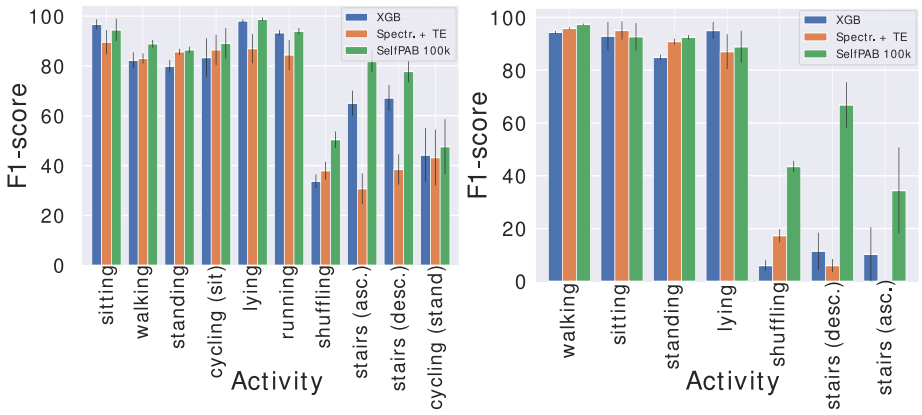
contain the results of the baselines, the fifth and sixth rows of setting (2) and (3), respectively, and the remaining row when using SelfPAB, pre-trained on 100k hours of HUNT4 data. With the highest F1-scores of 81.3% and 78.5% for HARTH and HAR70+, respectively, SelfPAB is the best model in our experiments, which is an improvement of around 7% compared to the best baseline model, the XGB. Furthermore, it generally attains a lower standard error than the baselines. The DeepConvLSTM trained on the time signals has the worst results in both datasets. A considerable improvement is observable when training DeepConvLSTM on spectrograms instead. However, it still shows the second-worst performance. Setting two (Spectrograms + MLP) has the third-worst F1-scores in both datasets. Setting three (Spectrograms + TE) outperforms the XGB in the HAR70+ dataset (64.5%) but not in the HARTH v1.2. Both, setting two and setting three, are considerably worse than SelfPAB, showing that the pre-training is important for good results.

Table 1: Average F1-score results of the leave-one-subject-out cross-validations on HARTH v1.2 and HAR70+. The best results are shown in bold letters.

| Approach | HARTH v1.2 (in %) | HAR70+ (in %) |
|---|---|---|
| SVM | 71.7 ± 2.0 | 64.3 ± 2.9 |
| XGB | 74.2 ± 1.9 | 63.7 ± 2.4 |
| DeepConvLSTM (TS) | 51.2 ± 6.2 | 54.7 ± 3.3 |
| DeepConvLSTM (Spectr.) | 60.2 ± 2.2 | 59.3 ± 1.9 |
| Spectr. + MLP | 60.5 ± 2.4 | 61.9 ± 2.5 |
| Spectr. + TE | 66.1 ± 2.0 | 64.5 ± 2.5 |
| SelfPAB (ours) | **81.3 ± 1.3** | **78.5 ± 2.1** |

Figure 2 shows the average F1-scores (with standard error) for each activity in the HARTH v1.2 (Figure 2a) and HAR70+ (Figure 2b) datasets. The SelfPAB, pre-trained on 100k hours, the XGB, and the Spectrograms + TE experiments are visible. The shown activities are ordered according to the number of samples in the dataset, with sitting being the most common activity for HARTH v1.2 and walking the most common for HAR70+. The well-represented classes (HARTH v1.2: sitting, walking, standing, cycling(sit), lying, and running; HAR70+: walking, sitting, standing, and lying) are largely dominated by good but similar results for all models. Cycling (sit) has a similar average performance across the models, considering the high standard error. Shuffling is an exception here for the HARTH v1.2 dataset. It has almost the same amount of samples as running but a much lower performance across all models. Nevertheless, SelfPAB has considerably better results than the baseline XGB for shuffling. The rare classes (HARTH v1.2: stairs (ascending), stairs (descending), and cycling(stand); HAR70+: shuffling, stairs (descending), and stairs (ascending)) have, in general, much poorer performance. Despite this, we observe that SelfPAB performs comparably well on these rare classes, especially on stairs (as-

cending) and stairs (descending). Cycling (stand), on the other hand, is similarly poor predicted by all models and shows a high standard error.




(a) HARTH v1.2 (b) HAR70+

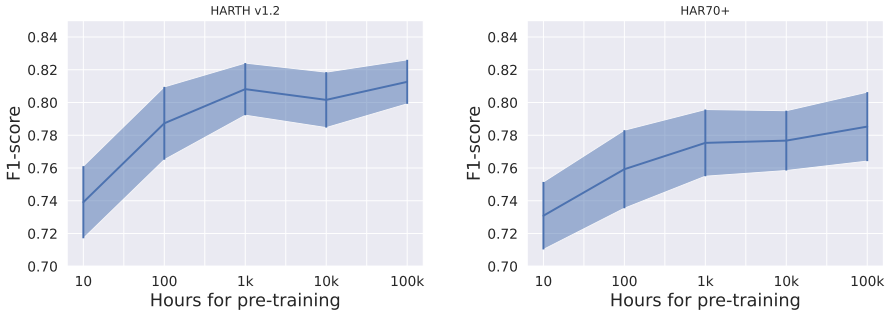
Fig. 2: Average F1-scores for each activity in the HARTH v1.2 (a) and HAR70+ (b) datasets. The black lines show the corresponding standard errors. The activities are ordered according to their amount of minutes in the dataset, with left being the most common activity. Spectr. is the abbreviation for Spectrograms.

### 5.2 Impact of the Amount of Unique Upstream Samples

The overall increase in the average F1-score with an increasing amount of hours is illustrated in Figure 3a for the HARTH v1.2 dataset and in Figure 3b for the HAR70+. For HARTH v1.2 (Figure 3a), a strong performance gain is achieved when training on 1k hours compared to 10 or 100 hours. Using more pre-training data improves the performance marginally, with 10k hours being worse than 1k hours. Similarly, in the HAR70+ experiments (Figure 3b), the F1-score increases with increasing hours used during pre-training, while the performance gain from 10 hours to 1k hours is stronger than from 1k hours to 100k hours. In both cases, the model pre-trained on the most hours of unique upstream samples, SelfPAB 100k, achieves the best average F1-score. Note that for HARTH v1.2 SelfPAB pre-trained on 10 hours has similar well results as the best baseline model, the XGB, with 73.9%. For HAR70+ SelfPAB pre-trained on 10 hours shows even better performance than all purely supervised baselines.

## 6 Discussion

Most activities benefit from our pre-training. However, while more frequent activities have a generally high performance for all models, less common activities




(a) HARTH v1.2 (b) HAR70+

Fig. 3: The average downstream F1-score for (a) HARTH v1.2 and (b) HAR70+ if SelfPAB is trained on 10 hours, 100 hours, 1k hours, 10k hours, and 100k hours of the HUNT4 data. The shaded areas represent the standard error, and the y-axis range is between 70% and 85%.

show a strong F1-score increase. This is beneficial, especially in current free-living datasets, which are relatively small and where certain activities are not performed that often. There are multiple explanations for this behavior. First, SelfPAB could be more robust against class imbalances, as observed in related works [9]. Second, the pre-training allows a more data-efficient training of downstream tasks. This is strengthened by the stairs (ascending/descending) results for both datasets. Third, certain activities are hard to distinguish from others due to their strong similarity. The poor shuffling performance in HARTH v1.2, with a sample count similar to running but notably lower performance, strengthens this explanation. Shuffling's "semantic" proximity to walking and standing can lead to misclassification. We further show that increasing the number of unique data samples for pre-training improves the HAR downstream performance, with 100k hours leading to the best results. A similar observation was made in [12] on transformer-based language models, where the loss scales with the amount of training data. SelfPAB enhances the performance of activities with limited data but requires longer training and higher power consumption, necessitating a trade-off between performance and training complexity. Nevertheless, we also show that already 10 hours of pre-training data are enough to achieve better/similar performances than the purely supervised baselines. Hence, our approach can learn useful representations even from small amounts of unlabeled data. Furthermore, the performance increase slows down after 1k hours, indicating a convergence. Hence, a limitation of our study is that we do not investigate more than 100k hours to examine whether an actual convergence occurs. Augmentation can be seen as an alternative to SelfPAB, since it can increase underrepresented class samples. However augmentation is considered less effective than SSL in related work [15]. Thus, a comparison with augmentation falls outside the scope of this study. Spectrograms + TE lags behind SelfPAB

pre-trained on just 10 hours, suggesting that weight freezing in SelfPAB serves as beneficial weight initialization procedure for downstream training. It remains an open question of how well SelfPAB performs compared to other SSL-based HAR approaches, like the ones presented in Section 2. Nevertheless, the focus of this work is not to create a novel state-of-the-art SSL approach for HAR but rather the investigation of the influence of the amount of pre-training data on the model's downstream HAR performance. Hence, we consider a comparison to other SSL approaches as out of the scope of this paper and refer it to future work.

## 7 Conclusion

Inspired by the recent success of self-supervised machine learning and the large-scale HUNT4 data corpus, we implement the SelfPAB method. SelfPAB learns physical activity representations by reconstructing masked parts of accelerometer signal spectrograms of the unlabeled HUNT4 dataset. SelfPAB achieves better HAR performances than purely supervised baselines, especially for activities with little data. Furthermore, we show that increasing the amount of unique pre-training samples leads to an increase in the downstream HAR performance. For future research, we recommend the investigation of a potential sensor location mismatch between pre-training and downstream data. It would reveal how robust SelfPAB is regarding sensor position. Furthermore, the fact that two separate sensors record the data can be used to design more innovative pre-training objectives. The ever-growing community of physical activity behavior research based on accelerometer (attached to the thigh and lower back) measurements will acquire new knowledge about the influence of physical activity behavior on public health by using our SelfPAB method.

## References

1. Åsvold, B.O., Langhammer, A., Rehn, T.A., Kjelvik, G., Grøntvedt, T.V., Sørgjerd, E.P., Fenstad, J.S., Heggland, J., Holmen, O., Stuifbergen, M.C., Vikjord, S.A.A., Brumpton, B.M., Skjellegrind, H.K., Thingstad, P., Sund, E.R., Selbæk, G., Mork, P.J., Rangul, V., Hveem, K., Næss, M., Krokstad, S.: Cohort Profile Update: The HUNT Study, Norway. International Journal of Epidemiology **52**(1), e80–e91 (Feb 2023). https://doi.org/10.1093/ije/dyac095
2. Chan Chang, S., Doherty, A.: Capture-24: Activity tracker dataset for human activity recognition. University of Oxford (2021)
3. Chi, P.H., Chung, P.H., Wu, T.H., Hsieh, C.C., Chen, Y.H., Li, S.W., Lee, H.y.: Audio Albert: A Lite Bert for Self-Supervised Learning of Audio Representation. In: 2021 IEEE Spoken Language Technology Workshop (SLT). pp. 344–350. IEEE, Shenzhen, China (Jan 2021). https://doi.org/10.1109/SLT48900.2021.9383575
4. Chuang, C.Y., Robinson, J., Yen-Chen, L., Torralba, A., Jegelka, S.: Debiased Contrastive Learning (Oct 2020). https://doi.org/10.48550/arXiv.2007.00224
5. Cleland, I., Kikhia, B., Nugent, C., Boytsov, A., Hallberg, J., Synnes, K., McClean, S., Finlay, D.: Optimal placement of accelerometers for the detection of

everyday activities. Sensors (Basel, Switzerland) **13**(7), 9183–9200 (Jul 2013). https://doi.org/10.3390/s130709183
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805 [cs] (May 2019)
7. Haresamudram, H., Beedu, A., Agrawal, V., Grady, P.L., Essa, I., Hoffman, J., Plötz, T.: Masked reconstruction based self-supervision for human activity recognition. In: Proceedings of the 2020 International Symposium on Wearable Computers. pp. 45–49. ISWC '20, Association for Computing Machinery, New York, NY, USA (Sep 2020). https://doi.org/10.1145/3410531.3414306
8. Haresamudram, H., Essa, I., Plötz, T.: Contrastive Predictive Coding for Human Activity Recognition. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies **5**(2), 65:1–65:26 (Jun 2021). https://doi.org/10.1145/3463506
9. Haresamudram, H., Essa, I., Plötz, T.: Assessing the State of Self-Supervised Human Activity Recognition Using Wearables. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies **6**(3), 116:1–116:47 (Sep 2022). https://doi.org/10.1145/3550299
10. Jain, Y., Tang, C.I., Min, C., Kawsar, F., Mathur, A.: ColloSSL: Collaborative Self-Supervised Learning for Human Activity Recognition. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies **6**(1), 17:1–17:28 (Mar 2022). https://doi.org/10.1145/3517246
11. Jaiswal, A., Babu, A.R., Zadeh, M.Z., Banerjee, D., Makedon, F.: A Survey on Contrastive Self-Supervised Learning. Technologies **9**(1), 2 (Mar 2021). https://doi.org/10.3390/technologies9010002
12. Kaplan, J., McCandlish, S., Henighan, T., Brown, T.B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., Amodei, D.: Scaling Laws for Neural Language Models (Jan 2020). https://doi.org/10.48550/arXiv.2001.08361
13. Khaertdinov, B., Ghaleb, E., Asteriadis, S.: Contrastive Self-supervised Learning for Sensor-based Human Activity Recognition. In: 2021 IEEE International Joint Conference on Biometrics (IJCB). pp. 1–8. IEEE, Shenzhen, China (Aug 2021). https://doi.org/10.1109/IJCB52358.2021.9484410
14. Le-Khac, P.H., Healy, G., Smeaton, A.F.: Contrastive Representation Learning: A Framework and Review. IEEE Access **8**, 193907–193934 (2020). https://doi.org/10.1109/ACCESS.2020.3031549
15. Liu, A.T., Li, S.W., Lee, H.y.: TERA: Self-Supervised Learning of Transformer Encoder Representation for Speech. IEEE/ACM Transactions on Audio, Speech, and Language Processing **29**, 2351–2366 (2021). https://doi.org/10.1109/TASLP.2021.3095662
16. Liu, A.T., Yang, S.w., Chi, P.H., Hsu, P.c., Lee, H.y.: Mockingjay: Unsupervised Speech Representation Learning with Deep Bidirectional Transformer Encoders. ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) pp. 6419–6423 (May 2020). https://doi.org/10.1109/ICASSP40776.2020.9054458
17. Liu, D., Abdelzaher, T.: Semi-Supervised Contrastive Learning for Human Activity Recognition. In: 2021 17th International Conference on Distributed Computing in Sensor Systems (DCOSS). pp. 45–53. IEEE, Pafos, Cyprus (Jul 2021). https://doi.org/10.1109/DCOSS52077.2021.00019
18. Logacjov, A., Bach, K., Kongsvold, A., Bårdstu, H.B., Mork, P.J.: HARTH: A Human Activity Recognition Dataset for Machine Learning. Sensors **21**(23), 7853 (Jan 2021). https://doi.org/10.3390/s21237853

19. Mao, H.H.: A Survey on Self-supervised Pre-training for Sequential Transfer Learning in Neural Networks (Jul 2020). https://doi.org/10.48550/arXiv.2007.00800
20. Narayanan, A., Stewart, T., Mackay, L.: A Dual-Accelerometer System for Detecting Human Movement in a Free-living Environment. Medicine & Science in Sports & Exercise **52**(1), 252–258 (Jan 2020). https://doi.org/10.1249/MSS.0000000000002107
21. Ordóñez, F.J., Roggen, D.: Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition. Sensors **16**(1), 115 (Jan 2016). https://doi.org/10.3390/s16010115
22. Saeed, A., Ozcelebi, T., Lukkien, J.: Multi-task Self-Supervised Learning for Human Activity Detection. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies **3**(2), 61:1–61:30 (Jun 2019). https://doi.org/10.1145/3328932
23. Saeed, A., Salim, F.D., Ozcelebi, T., Lukkien, J.: Federated Self-Supervised Learning of Multisensor Representations for Embedded Intelligence. IEEE Internet of Things Journal **8**(2), 1030–1040 (Jan 2021). https://doi.org/10.1109/JIOT.2020.3009358
24. Stewart, T., Narayanan, A., Hedayatrad, L., Neville, J., Mackay, L., Duncan, S.: A Dual-Accelerometer System for Classifying Physical Activity in Children and Adults. Medicine and Science in Sports and Exercise **50**(12), 2595–2602 (Dec 2018). https://doi.org/10.1249/MSS.0000000000001717
25. Taghanaki, S.R., Rainbow, M., Etemad, A.: Self-supervised Human Activity Recognition by Learning to Predict Cross-Dimensional Motion. 2021 International Symposium on Wearable Computers pp. 23–27 (Sep 2021). https://doi.org/10.1145/3460421.3480417
26. Tang, C.I., Perez-Pozuelo, I., Spathis, D., Brage, S., Wareham, N., Mascolo, C.: SelfHAR: Improving Human Activity Recognition through Self-training with Unlabeled Data. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies **5**(1), 1–30 (Mar 2021). https://doi.org/10.1145/3448112
27. Tonekaboni, S., Eytan, D., Goldenberg, A.: Unsupervised Representation Learning for Time Series with Temporal Neighborhood Coding. arXiv:2106.00750 [cs, stat] (Jun 2021)
28. Twomey, N., Diethe, T., Fafoutis, X., Elsts, A., McConville, R., Flach, P., Craddock, I.: A Comprehensive Study of Activity Recognition Using Accelerometers. Informatics **5**(2), 27 (Jun 2018). https://doi.org/10.3390/informatics5020027
29. Ustad, A., Logacjov, A., Trollebø, S.Ø., Thingstad, P., Vereijken, B., Bach, K., Maroni, N.S.: Validation of an Activity Type Recognition Model Classifying Daily Physical Behavior in Older Adults: The HAR70+ Model. Sensors **23**(5), 2368 (Jan 2023). https://doi.org/10.3390/s23052368
30. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention Is All You Need. arXiv:1706.03762 [cs] (Dec 2017)
31. Wang, J., Zhu, T., Chen, L., Ning, H., Wan, Y.: Negative Selection by Clustering for Contrastive Learning in Human Activity Recognition. arXiv:2203.12230 [cs] (Mar 2022)
32. Wang, J., Zhu, T., Gan, J., Chen, L., Ning, H., Wan, Y.: Sensor Data Augmentation by Resampling for Contrastive Learning in Human Activity Recognition. arXiv:2109.02054 [cs] (Mar 2022)
33. Yu, J., Wang, Z., Vasudevan, V., Yeung, L., Seyedhosseini, M., Wu, Y.: CoCa: Contrastive Captioners are Image-Text Foundation Models (May 2022). https://doi.org/10.48550/arXiv.2205.01917