Masked Reconstruction Based Self-Supervision for Human Activity Recognition

Harish Haresamudram, Apoorva Beedu, Varun Agrawal, Patrick L. Grady, Irfan Essa, Judy Hoffman, Thomas Plötz
Georgia Institute of Technology
Atlanta, Georgia, USA
{harishkashyap,abeedu3,varunagrawal,patrick.grady,irfan,judy,thomas.ploetz}@gatech.edu

ABSTRACT

The ubiquitous availability of wearable sensing devices has rendered large scale collection of movement data a straightforward endeavor. Yet, annotation of these data remains a challenge and as such, publicly available datasets for human activity recognition (HAR) are typically limited in size as well as in variability, which constrains HAR model training and effectiveness. We introduce masked reconstruction as a viable self-supervised pre-training objective for human activity recognition and explore its effectiveness in comparison to state-of-the-art unsupervised learning techniques. In scenarios with small labeled datasets, the pre-training results in improvements over end-to-end learning on two of the four benchmark datasets. This is promising because the pre-training objective can be integrated "as is" into state-of-the-art recognition pipelines to effectively facilitate improved model robustness, and thus, ultimately, leading to better recognition performance.

CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile computing**; • **Computing methodologies** → **Artificial intelligence**; **Unsupervised learning**.

KEYWORDS

Activity recognition; Representation learning; Self-supervision

ACM Reference Format:
Harish Haresamudram, Apoorva Beedu, Varun Agrawal, Patrick L. Grady, Irfan Essa, Judy Hoffman, Thomas Plötz. 2020. Masked Reconstruction Based Self-Supervision for Human Activity Recognition. In *Proceedings of the 2020 International Symposium on Wearable Computers (ISWC '20), September 12–16, 2020, Virtual Event, Mexico.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3410531.3414306

1 INTRODUCTION

Machine learning models for sensor-based human activity recognition (HAR) generally rely upon the availability of annotated datasets that are used for supervised training [3]. While the ubiquitous availability of commodity wearable sensing platforms renders data collection straightforward, obtaining the required ground truth annotation remains a challenge resulting in small datasets for model training and evaluation, typically covering only few activities and participants. Small annotated datasets limit the complexity of analysis models and thus the capabilities of activity recognition systems.

Previous work has shown that unlabeled sensor data can effectively be used for deriving feature representations [2, 12, 27] that then can be integrated into activity recognition chains [3, 28]. We follow this general approach of utilizing unlabelled sensor data for pre-training (components of) human activity recognition systems.

The pre-training procedure is adopted from related domains of sequential data analysis, such as natural language processing or automated speech recognition. Frames of unlabeled sensor data are perturbed by randomly masking out portions of the sensor readings, and the model is trained to reconstruct only the masked portions. Previous work in non-HAR domains has shown that such training procedures effectively allow to learn temporal context, which is beneficial for time-series analysis. We exploit this idea to learn effective sensor data representations from unlabeled samples that form the basis for subsequent, supervised model training on small datasets from the particular target domains.

In order to perform the self-supervised pre-training, we employ a (variant of the) BERT model [5], which utilizes masking to perform bidirectional encoding using Transformer encoders [34]. As sensory data contain continuous values, we replace all sensory data at randomly chosen time steps with zeros (in contrast to BERT, which replaces some tokens with a specific mask token). Subsequently, we utilize mean squared error loss to reconstruct the masked data. This is similar to Wang *et al.* [36], where the network is trained to regress to the missing log mel energy values given an spectrogram with 'missing' (or masked) regions, in order to improve ASR performance via unsupervised pre-training. The pre-trained weights are then utilized for feature extraction and transfer learning, each of which is integrated into a conventional activity recognition chain for sensor-based HAR [3]. In summary, our contributions are:

- We introduce masked reconstruction to human activity recognition as a viable self-supervised pre-training objective.
- We apply the Transformer encoder architecture to continuous data from body worn sensors.
- On three out of four benchmark datasets, we show comparable if not better performance over both unsupervised learning and end-to-end training.
- On two out of four benchmark datasets, we observe improved recognition performance over end-to-end training when there is limited availability of labeled samples.

ISWC '20, September 12–16, 2020, Virtual Event, Mexico

ACM ISBN 978-1-4503-8077-5/20/09…$15.00
https://doi.org/10.1145/3410531.3414306

2 BACKGROUND

Our work is focused on deriving effective data representations to be used as part of the standard activity recognition chain (ARC) [3]. In what follows we summarize the related work for this field.

Feature extraction for HAR. Features play a crucial role in ARC-based human activity recognition. The state-of-the-art covers both engineered and learned data representations. For the former, typically heuristics are employed to extract time-domain features, such as statistical moments, or spectral representations [8]. Other approaches include utilizing classic dimensionality reduction techniques such as PCA [27] and distribution-based approaches [11].

Feature learning involves extracting representations by optimizing objective functions on raw sensor data, either supervised or unsupervised. Unsupervised methods have used Restricted Boltzmann Machines (RBMs) [27], and recently Haresamudram *et al.*[12] showed that feature learning with autoencoders in an ARC often outperforms end-to-end modeling. Ghods *et al.*[9] learned embeddings for activities of daily living (ADL) using a sequence-to-sequence model.

Self-supervision. Self-supervision utilizes domain expertise to design a prediction task—the "pretext" task—that generates supervisory signals related to the downstream recognition task, which is then used for targeted representation learning. For example, in computer vision, inpainting [25], predicting image rotations [10], colorization [15], temporal order verification [20], or odd sequence detection [7] are used as pretext tasks. In NLP, representations are learned in form of word and sentence vectors that have been computed using contextual feature embeddings such as Word2vec [19], GloVe [26] and more recently, BERT [5].

Self-supervision for representation learning in HAR has so far utilized multi-task setups to pre-train encoder weights. For example, in [30] eight data transformation techniques were defined and a network was trained to predict whether each transformation has been applied or not (i.e., multi-task setting). The encoder is common to all tasks while each task has its own specific fully connected layers. This work utilizes only accelerometer signals, while our work is generic w.r.t. modalities and thus has broad applicability.

In other domains, similar approaches have been introduced. For example, Wang *et al.*[36] target automated speech recognition (ASR). They use bidirectional recurrent networks and randomly perform masking on groups of timesteps as well as frequencies on log mel filterbank energies to formulate the pretext task. In contrast, our approach randomly masks out specific number of single timesteps, while not masking any sensors completely across time-steps. Other works like Schneider *et al.*[31] learn convolutional representations to improve the ASR performance with smaller labelled datasets.

Transformers for time series data. Recently, Transformer networks [34] have been introduced for processing sequential information, and have been primarily applied in natural language processing. They model sequential information by solely employing self-attention mechanisms and dispense entirely with recurrence and convolutions, therefore making the networks more parallelizable and requiring significantly less time to train [34]. Their ability to model long sequences has been leveraged in time series forecasting [16]. Transformer encoders have been utilized to classify activities in HAR [17], yet their main application is still elsewhere (e.g., in ASR [6, 21], and generic time-series forecasting [16, 32, 37]).

3 PRE-TRAINING FOR HAR

In this paper, we utilize masking of sensory data at random timesteps as a pre-training objective. The encoder is then forced to reconstruct the masked out sensor readings, thereby processing the sequences both from left to right and the other way around. We leverage the idea that such bidirectional encoding incorporates temporal context, and is beneficial towards learning representations for time series data. In what follows, we first detail the pretext task, followed by the explanation of the model architecture for the encoder. Finally we also describe the classification backend that is utilized to compute the performance of the proposed approach.

3.1 Self-Supervision Pipeline

Figure 1 details the self-supervision pipeline. It consists of two steps: *(i)* **pre-training**, where we utilize the unlabeled data to learn the encoder weights via self-supervision; *(ii)* **fine-tuning**, where we subsequently use the learned encoder weights for feature extraction as part of the activity recognition chain (ARC) [3]. The performance of the representations is evaluated using an MLP classifier.

3.2 "Pretext" Task

Representation learning is based on analyzing frames that contain T consecutive N-dimensional sensor readings extracted using a standard sliding window procedure. Similar to BERT, we randomly pick x% of the samples in a frame to be masked out. In our case, we set the values across all sensors for each randomly chosen timestep to zero. The goal is to force the model to reconstruct the masked out parts and thus, to learn temporal patterns from context, which makes for a rich representation that is derived directly from data. In our study, we randomly choose 10% of the samples in every frame for masking.

Our pre-training approach is detailed in Figure 1. Each input frame F is perturbed by a binary mask M (same dimensions as F). This perturbed input is passed through the Encoder g consisting of Transformer encoder and embedding layers to obtain representations at each timestep. Max pooling is then applied to the representations in order to obtain the feature vector for the entire frame. Following the encoding, a set of fully connected (FC) layers h is used to match the dimensions of the input. Mean squared error (MSE) loss is computed between the input frame and the reconstructed input *only* on the masked portion (similar to BERT [5]). This is in contrast to denoising autoencoders, where the entire perturbed frame is reconstructed [35]. Similar to [36], the reconstruction loss used to update network parameters is defined as:

$$L(F, M; g, h) = ||(1 - M) \odot [X - h(g(M \odot X))]||^2_{Fro}$$

where $\odot$ denotes element-wise multiplication.

While the masking allows us to obtain bidirectional representations, it creates a mismatch between pre-training and fine-tuning [5]. Thus, we utilize the strategy detailed in [5] as follows: for the i^{th} timestep chosen, we replace the sensory data with: *a)* zeros 80% of the time; *b)* the unchanged sensory data 10% of the time; and *c)* data from a random time-step within the frame 10% of the time.

3.3 Encoder Architecture

We detail the Encoder architecture in Figure 2. In this paper, we utilize a multi-layer Transformer encoder based on [34] as the encoder for the pre-training. The input sensory data is transformed

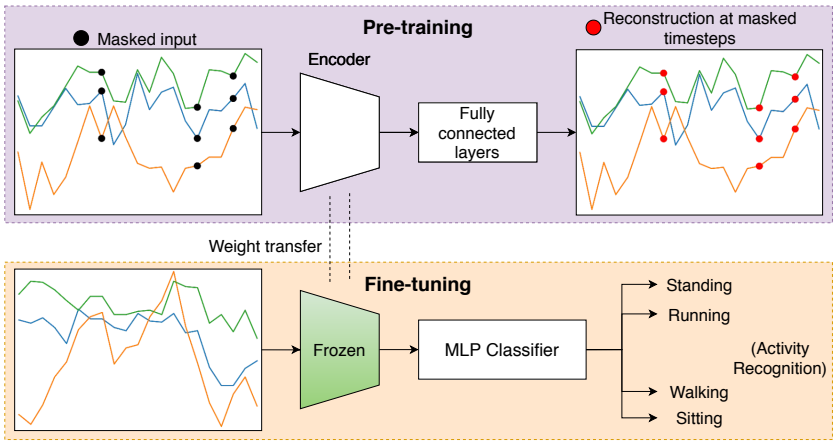

Figure 1: The self-supervision pipeline.

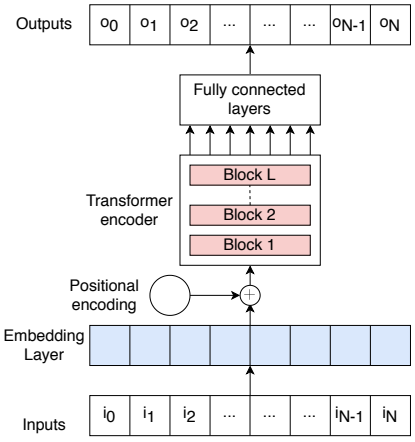

Figure 2: Encoder architecture.

to embeddings of 128 dimensions by utilizing 1D convolutions. As in Russwurm *et al.* [29], we also utilize LayerNorms throughout. The Transformer encoder contains no convolutions nor recurrence, and thus cannot make use of the temporal order of data. In order to inject positional information about the sequence of sensor readings, we use sinusoidal position embeddings [34]. The positional embeddings are designed to have the same dimensions as the input embeddings (128), and the two are summed before being input to the Transformer encoder.

3.4 Fully-Connected Layers and Recognizer

We use similar network architectures for the FC layers and the recognition backend. The first two layers have 256 and 128 units. For pre-training, the final layer matches the input dimensions whereas for fine-tuning, the last layer performs the softmax operation used for classification. We also apply ReLU activation [22], batch normalization [13] and dropout [33] with $p = 0.2$ between the layers.

4 EXPERIMENTS

4.1 Setup

Our experiments were based on four benchmark datasets (accelerometer plus gyroscope): Mobiactv2 [4], Motionsense [18], USC-HAD [38], and UCI-HAR [1]. The datasets cover locomotion activities such as walking, jogging, and standing. Mobiact, Motionsense and UCI-HAR were chosen since they were evaluated in Multi-task self-supervision [30] (which is one of our baselines), and contain data from both accelerometer and gyroscope. USC-HAD was also utilized since it comprises of similar activities and contains recordings from both sensors. This allows for transfer learning between the datasets as the number of dimensions is the same. For all but the USC-HAD dataset, we utilized the protocol from [30], where 20% of the participants were randomly chosen as test set. Out of the remaining participants, 20% were chosen randomly as validation set, and the rest was used for training. For USC-HAD, we followed the protocol from [12] with data from participants $1-10$ for training, from participants 11 and 12 for validation, and from participants 13 and 14 for testing. The data were downsampled to 33Hz. We used sliding window segmentation to obtain frames of 1s length with 50% overlap between subsequent windows.

We implemented the proposed method and related baselines using PyTorch [24]. For pre-training, we used the Noam optimizer [34] with a warmup of 4,000 steps and trained for 150 epochs. The feedforward dimensions for the Transformer encoder were set to 2,048 and dropout [33] was applied with $p = 0.1$. We tuned the number of heads and layers for each dataset. Fine-tuning was performed for 150 epochs with cross entropy loss. We utilized the Adam [14] optimizer and tuned over the learning rates $\in \left[10^{-3}, 10^{-5}\right]$ and L2 regularization $\in \left[10^{-2}, 10^{-4}\right]$. The learning rate was decayed by a factor of 0.8 every 25 epochs.

4.2 Baseline Recognition Experiment

Table 1: Recognition performance (test mean F1) of proposed approach compared to state-of-the-art unsupervised learning (‡), and to supervised learning (*), i.e., DeepConvLSTM (for reference). Results for [30] comparable to original publication, yet not identical due to original implementation not being released, and details being omitted in [30].

Method	Mobi -act	Motion -sense	USC -HAD	UCI -HAR
DeepConvLSTM* [23]	82.40	85.15	44.83	82.83
Transformer classifier*	80.96	83.30	43.84	82.61
Multi-task self sup.‡ [30]	75.41	83.30	45.37	80.20
CAE‡ [12]	79.58	82.50	48.82	80.26
Proposed‡	76.81	88.02	49.31	81.89

In this experiment, we evaluated the effectiveness of the self-supervision pretext task for representation learning. Once the pretext task was trained, the encoder weights were transferred to a randomly initialized activity recognition network for fine-tuning (see Figure 1). The learned weights were frozen, and cross entropy loss was utilized to update the weights of the classifier layers.

We compared our approach to state-of-the-art unsupervised learning techniques, i.e., convolutional autoencoders (CAE) [12] and multi-task self-supervision [30], and, for reference only given that the main focus of this paper is on unsupervised approaches, to supervised learning pipelines (DeepConvLSTM [23] and Transformer classifier). We used the CAE architecture from [12], with a bottleneck size of 128 dimensions, in order to match the output dimensions of the Transformer encoder. CAE representations were evaluated with the same classifier network as in the proposed technique. Our implementation of multi-task self-supervision [30] varies as we pre-process the data to a lower sampling rate.

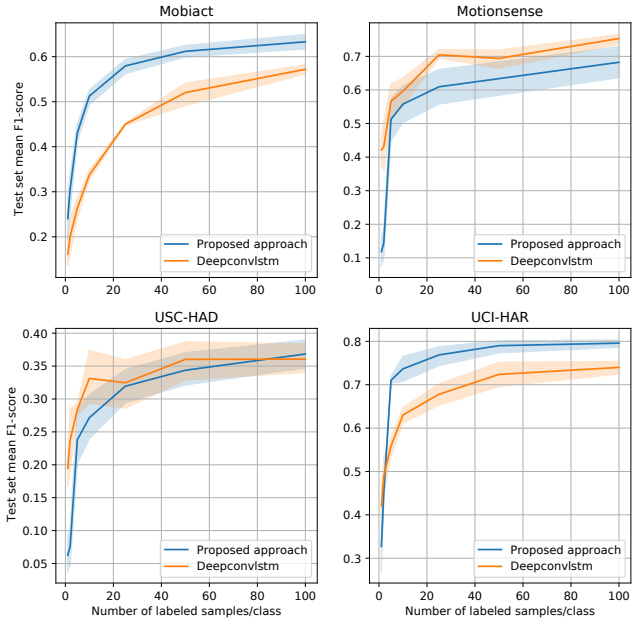

Figure 3: Results for fine-tuning with limited labeled data.

Thus, we utilize a kernel size of 5 across all three convolutional layers in the encoder. Further, the Transformer classifier utilizes the same network architecture used for fine-tuning, albeit it is initialized randomly and all weights (including the encoder) are trained end-to-end. Results are tabulated in Table 1.

Relative to the CAE, we see improvements in performance on Motionsense, USC-HAD and UCI-HAR. While the CAE has around $7M$ parameters, our proposed approach only has $\sim 1.5M$ parameters, which is approximately 20% of the autoencoders parameters. Even under such mismatched conditions, the proposed method outperforms the CAE. Our proposed method also consistently outperforms the multi-task self-supervision approach on all the datasets. Furthermore, the proposed approach performs comparably to end-to-end learning with DeepConvLSTM, outperforming it by approx. 4.5% on USC-HAD and by $\sim$ 3% on Motionsense. We see stronger performance against the Transformer classifier trained in a supervised manner. In contrast to the proposed approach (which has frozen encoder weights), the Transformer classifier updates the weights on all layers. The positive impact of the learned weights can clearly be seen as the proposed approach outperforms the Transformer classifier, while utilizing a fraction of the learnable parameters.

4.3 Fine-Tuning on Small Annotated Datasets

Next, we considered the main target scenario for our work, where limited numbers of annotated samples are available for fine-tuning, yet a large unlabeled dataset is available for pre-training. This scenario is particularly important as data *collection* is typically straightforward whereas *annotation* is often challenging.

For each dataset, we utilized the entire training set (without labels) for pre-training. Subsequently, we randomly sampled x labeled samples per class where $x \in [1, 2, 5, 10, 25, 50, 100]$ for fine-tuning. We performed five runs and plotted the test mean F1-score in Figure 3. The effectiveness of the learned encoder weights is compared to DeepConvLSTM. As before, DeepConvLSTM was trained end-to-end, while the learned encoder weights were frozen. On the Mobiact dataset, we observe improvements over DeepConvLSTM even when just one labeled sample per class is available. For UCI-HAR however, the boost in performance seen when there are atleast 5 labeled samples per class. The proposed approach requires 25 samples or more per class to obtain performance comparable to DeepConvLSTM on USC-HAD, while at 100 samples per class, we observe modest improvement. Although the proposed approach does not outperform on Motionsense, we observe that the learned weights (even when frozen) can perform well in limited annotated data settings.

Table 2: Transfer learning performance (Test mean F1-score) of the proposed approach against unsupervised learning (‡) and supervised learning (*) baselines.

Method	Motion -sense	USC -HAD	UCI -HAR
DeepConvLSTM* [23]	69.12	25.57	73.68
Transformer classifier*	80.75	47.34	81.24
Multi-task self sup.‡ [30]	79.30	31.35	73.89
CAE‡ [12]	84.97	51.66	84.15
Proposed‡	79.86	46.19	81.37

4.4 Transfer Learning

Here we evaluated the performance of the proposed approach for transfer learning. We begin with pre-training using Mobiact as it contains the largest number of participants. Subsequently, the frozen learned weights were used for fine-tuning the classifier on the remaining datasets. Similarly, for both DeepConvLSTM and the Transformer classifier, the encoder was kept frozen and the classifier network was optimized on the target datasets.

Table 2 compares the performance of the proposed approach to other unsupervised approaches (and, again for reference, to a supervised approach, DeepConvLSTM and a Transformer classifier). We note that the CAE significantly outperforms the supervised transfer learning approaches. Similarly, the proposed approach shows improved performance over using DeepConvLSTM for transfer. We observe comparable performance to using the Transformer classifier. Note that the unsupervised approaches perform better at transfer learning. This validates our hypothesis that utilizing a large unlabeled body of data improves downstream performance.

5 CONCLUSION

We have introduced masked reconstruction as a viable self-supervised pre-training objective for application to human activity recognition pipelines. On three out of four benchmark datasets, we demonstrated improved performance over state-of-the-art unsupervised learning approaches including convolutional autoencoders. On two out of four benchmark datasets, we demonstrated improvements when finetuning on limited labelled data. This result is of particular practical importance as it allows us to effectively utilize unlabeled data. *Collecting* large amounts of data using wearables is straightforward, yet *annotating* these is often very challenging. We explored how to alleviate the reliance on large-scale labeled datasets.

ACKNOWLEDGMENTS

We would like to acknowledge Nvidia for generously gifting a GPU, and Google for granting a faculty research award to TP.

REFERENCES

[1] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, and Jorge Luis Reyes-Ortiz. 2013. A public domain dataset for human activity recognition using smartphones.. In *Esann*.

[2] Sourav Bhattacharya, Petteri Nurmi, Nils Hammerla, and Thomas Ploetz. 2014. Using Unlabeled Data in a Sparse-coding Framework for Human Activity Recognition. *Pervasive and Mobile Computing* (2014).

[3] Andreas Bulling, Ulf Blanke, and Bernt Schiele. 2014. A tutorial on human activity recognition using body-worn inertial sensors. *ACM Computing Surveys (CSUR)* 46, 3 (2014), 1–33.

[4] Charikleia Chatzaki, Matthew Pediaditis, George Vavoulas, and Manolis Tsiknakis. 2016. Human daily activity and fall recognition using a smartphone's acceleration sensor. In *International Conference on Information and Communication Technologies for Ageing Well and e-Health*. Springer, 100–118.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[6] Linhao Dong, Shuang Xu, and Bo Xu. 2018. Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 5884–5888.

[7] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. 2017. Self-supervised video representation learning with odd-one-out networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3636–3645.

[8] Davide Figo, Pedro C Diniz, Diogo R Ferreira, and Joao MP Cardoso. 2010. Preprocessing techniques for context recognition from accelerometer data. *Personal and Ubiquitous Computing* 14, 7 (2010), 645–662.

[9] Alireza Ghods and Diane J Cook. 2019. Activity2Vec: Learning ADL Embeddings from Sensor Data with a Sequence-to-Sequence Model. *arXiv preprint arXiv:1907.05597* (2019).

[10] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. 2018. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728* (2018).

[11] Nils Y Hammerla, Reuben Kirkham, Peter Andras, and Thomas Ploetz. 2013. On preserving statistical characteristics of accelerometry data using their empirical cumulative distribution. In *Proceedings of the 2013 international symposium on wearable computers*. 65–68.

[12] Harish Haresamudram, David V Anderson, and Thomas Plötz. 2019. On the role of features in human activity recognition. In *Proceedings of the 23rd International Symposium on Wearable Computers*. 78–88.

[13] Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167* (2015).

[14] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[15] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. 2017. Colorization as a proxy task for visual understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6874–6883.

[16] Shiyang Li, Xiaoyong Jin, Yao Xuan, Xiyou Zhou, Wenhu Chen, Yu-Xiang Wang, and Xifeng Yan. 2019. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. In *Advances in Neural Information Processing Systems*. 5244–5254.

[17] Saif Mahmud, M Tonmoy, Kishor Kumar Bhaumik, AKM Rahman, M Ashraful Amin, Mohammad Shoyaib, Muhammad Asif Hossain Khan, and Amin Ahsan Ali. 2020. Human Activity Recognition from Wearable Sensor Data Using Self-Attention. *arXiv preprint arXiv:2003.09018* (2020).

[18] Mohammad Malekzadeh, Richard G Clegg, Andrea Cavallaro, and Hamed Haddadi. 2018. Protecting sensory data against sensitive inferences. In *Proceedings of the 1st Workshop on Privacy by Design in Distributed Systems*. 1–6.

[19] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[20] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. 2016. Shuffle and learn: unsupervised learning using temporal order verification. In *European Conference on Computer Vision*. Springer, 527–544.

[21] Abdelrahman Mohamed, Dmytro Okhonko, and Luke Zettlemoyer. 2019. Transformers with convolutional context for ASR. *arXiv preprint arXiv:1904.11660* (2019).

[22] Vinod Nair and Geoffrey E Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*. 807–814.

[23] Francisco Javier Ordóñez and Daniel Roggen. 2016. Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. *Sensors* 16, 1 (2016), 115.

[24] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Eds.). Curran Associates, Inc., 8024–8035. http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

[25] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. 2016. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2536–2544.

[26] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1532–1543.

[27] Thomas Plötz, Nils Y Hammerla, and Patrick L Olivier. 2011. Feature learning for activity recognition in ubiquitous computing. In *Twenty-second international joint conference on artificial intelligence*.

[28] Thomas Plötz, Paula Moynihan, Cuong Pham, and Patrick Olivier. 2011. Activity recognition and healthier food preparation. In *Activity Recognition in Pervasive Intelligent Environments*. Springer, 313–329.

[29] Marc Rußwurm and Marco Körner. 2019. Self-attention for raw optical satellite time series classification. *arXiv preprint arXiv:1910.10536* (2019).

[30] Aaqib Saeed, Tanir Ozcelebi, and Johan Lukkien. 2019. Multi-task Self-Supervised Learning for Human Activity Detection. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 2 (2019), 1–30.

[31] Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. 2019. wav2vec: Unsupervised pre-training for speech recognition. *arXiv preprint arXiv:1904.05862* (2019).

[32] Huan Song, Deepta Rajan, Jayaraman J Thiagarajan, and Andreas Spanias. 2018. Attend and diagnose: Clinical time series analysis using attention models. In *Thirty-second AAAI conference on artificial intelligence*.

[33] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research* 15, 1 (2014), 1929–1958.

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

[35] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. 2008. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*. 1096–1103.

[36] Weiran Wang, Qingming Tang, and Karen Livescu. 2020. Unsupervised pre-training of bidirectional speech encoders via masked reconstruction. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 6889–6893.

[37] Neo Wu, Bradley Green, Xue Ben, and Shawn O'Banion. 2020. Deep Transformer Models for Time Series Forecasting: The Influenza Prevalence Case. *arXiv preprint arXiv:2001.08317* (2020).

[38] M. Zhang and A. Sawchuk. 2012. USC-HAD: a daily activity dataset for ubiquitous activity recognition using wearable sensors.