arXiv:2003.09018v1 [cs.CV] 17 Mar 2020

# Human Activity Recognition from Wearable Sensor Data Using Self-Attention

**Saif Mahmud** [1] and **M Tanjid Hasan Tonmoy** [1] and **Kishor Kumar Bhaumik** [2] and **A K M Mahbubur Rahman** [2] and **M Ashraful Amin** [2] and **Mohammad Shoyaib** [1] and **Muhammad Asif Hossain Khan** [1] and **Amin Ahsan Ali** [2]

**Abstract.** Human Activity Recognition from body-worn sensor data poses an inherent challenge in capturing spatial and temporal dependencies of time-series signals. In this regard, the existing recurrent or convolutional or their hybrid models for activity recognition struggle to capture spatio-temporal context from the feature space of sensor reading sequence. To address this complex problem, we propose a self-attention based neural network model that foregoes recurrent architectures and utilizes different types of attention mechanisms to generate higher dimensional feature representation used for classification. We performed extensive experiments on four popular publicly available HAR datasets: PAMAP2, Opportunity, Skoda and USC-HAD. Our model achieve significant performance improvement over recent state-of-the-art models in both benchmark test subjects and Leave-one-subject-out evaluation. We also observe that the sensor attention maps produced by our model is able capture the importance of the modality and placement of the sensors in predicting the different activity classes.

## 1 Introduction

Human Activity Recognition (HAR) has drawn extensive attention in various areas of mobile health and context-aware computing such as recognition of Nurse care activities [9], assessment of the quality of physical activities or exercises performed by rehabilitation patients or athletes [18]. HAR is defined as the automated classification of the activities of specific subjects wearing heterogeneous sensors placed at different body locations. In other words, HAR takes the readings from different body-worn sensors as input and afterward, it segments and classifies the time-series sensor signal in accordance with the extracted features. Currently, the task of assessment of quality of physical activities or exercises performed by patients is usually performed by an expert physiotherapist. A HAR system can be used to perform this assessment in real-time and assist the healthcare professionals.

Although HAR is the core area of wearable and ubiquitous computing, it remains one of the most challenging ones. This is due to large number of sensor modalities, noisy signals, variation in the spatial and temporal dimension of the feature space across subjects and even when the same subject performs the same task at different times, and so on. Researchers from last decades introduced a number of hand-crafted signal processing equations to derive statistical features out of the time-series sensors data. Examples of statistical features are mean, variance, and Fast Fourier Transform coefficients. Then, they used several supervised classification techniques: Support Vector Machines, Decision Trees, Bayesian classifiers to classify the activities [3, 19]. Later, deep learning based techniques enabled the learning of feature representations for classification tasks without involving domain-specific knowledge. A number of researches have been performed with various architectures of Convolutional Neural Network (CNN) [5, 20]. Simple physical activities (e.g., walking or cycling) and postures (e.g., sitting or standing) are automatically recognized with good performance with the above-mentioned techniques. However, recognition of many complex activities (stair-up/down, running/jogging, watching TV, ironing) remain challenging. Moreover, sensor displacement and other sources of noise make the HAR more error-prone.

Recently, hybrid deep learning model consisting of CNN and Recurrent Neural Network (RNN) [26, 17] has achieved better performance with a considerable margin than the conventional CNN model for complex activities as these models consider the activity recognition as a sequence labeling problem. The convolution and recurrent layers (Gated Recurrent Unit or GRU [26] and Long Short-Term Memory [17]) together capture the temporal characteristics and relationship among the different sensor modalities. More recently, [16] explores attention module on top of the recurrent layers to improve the performance of the models. In [27, 15], the authors explore temporal attention on top of recurrent layers and attention on sensor modality and show that the models improve the performance of HAR on some benchmark human activity datasets.

The recurrent neural network based encoder-decoder architectures developed for natural language processing tasks such as Neural Machine Translation (NMT) [24] are unable to capture the context from all possible transformed feature combinations. To address this limitation, several research works proposed attention-based mechanism [2] for NMT where varying attention is given to different words of a sentence. However, recurrent networks are constrained by their sequential operations. These limitations have led the researchers towards innovating Transformer architecture [25] for NMT.

Transformer leverages self-attention [14] which enables the model to capture context within the sequence. Transformer avoids the sequential processing involved in recurrent architectures and depends solely on self-attention and positional encoding technique. Transformer also includes multi-headed architecture in order to capture self-attention from different perspectives. Thus, transformer architecture plays important role in capturing context through weight distribution in the temporal dimension and in computing attention in

[1] University of Dhaka, Bangladesh, email: {2015-116-815, 2015-116-770}@student.cse.du.ac.bd , {shoyaib, asif}@du.ac.bd; The first two authors have equal contributions.
[2] Independent University Bangladesh, email: {1621366, akmmrahman, aminmdashraful,aminali}@iub.edu.bd

natural language modeling. With the similar idea, we adopt self-attention architecture from NMT task for HAR and propose a model incorporating self-attention with sensor and temporal attention.

In this paper, we propose that sensor's data samples are equivalent to words and windows (time window) are analogous to sentence. Hence, the objective of this paper is to build an attention based end-to-end system where attention is utilized in different ways to create effective feature representation of sensor data. To do that, we introduce the first attention layer on the raw input. Secondly, we adopted self-attention and positional encoding from the transformer architecture [25] for HAR to capture spatio-temporal dependencies of sensor signals and their modalities. After a number of self attention blocks, we add another layer of attention that facilitates learning of global attention from the context. Finally, a fully connected layer is placed to classify the activity.

In this work, we have experimented with four benchmark human activity recognition datasets: PAMAP2 [21], USC-HAD [28], Opportunity [22], and SKODA [23] and compared our results with the current state-of-the-art techniques namely DeepConvLSTM [17] and Convolutional Autoencoder (ConvAE) [10] to demonstrate the effectiveness of the proposed approach. We observe that the proposed model outperforms the DeepConvLSTM and ConvAE for both sample-wise and window-wise experimental setup on benchmark test-cases from the aforementioned datasets. We also perform leave one subject out cross validation experiments to show the superiority of our proposed model for generalization across subjects. Hence, our contributions are enlisted below:

1. We propose a self-attention based non-recurrent neural network architecture for HAR.
2. We incorporate sensor modality attention and global temporal attention at different layers. The attention layers capture the spatio-temporal context in the sensor signal to construct feature representation for classification.
3. We compare our model with other state of the art models on four publicly available HAR datasets in terms of both benchmark test sets and leave one subject out tests. In addition, we analyze the impact of various window-size on the proposed and other existing models.
4. We construct sensor-level attention maps that are intuitively explainable and thus demonstrates interpretability of the modules of the architecture.

The rest of this paper is organized as follows. Section 2 introduces related works on HAR using deep learning and various attention models. We describe the technical details of our proposed model in Section 3. In Section 4 and 5, we describe the datasets and the setup of the experiments, respectively. Comparative results with existing models and other experiments are presented in Section 6. Finally, we conclude the paper in Section 7.

## 2 Related Works

Plethora of research has been conducted in the area of human activity recognition since 2000. Recently, Haresamudram et al. [10] presented a comprehensive review of deep learning based feature extraction and recognition models for HAR using sensor data. In the last decade, most of the wearable device-based HAR involves hand-crafted features from domain-specific knowledge in the case of shallow machine learning models. These models depend on statistical features [4, 19] and distribution-based features [13, 7]. Statistical features are calculated using different statistical characteristics equations [4]. There also has been efforts to use LSTM with attention on statistical and geometric features in the context of HAR based on 3D skeleton data [9, 29]. On the other hand, distribution-based feature representations are obtained from signal processing approaches [11] such as wavelet or Fourier transformation.

Recently, end-to-end deep learning based techniques for HAR have gained more popularity among the machine learning communities. As the deep learning based techniques find the most appropriate feature representations for the HAR with supervised fashion, they have eliminated human-intervened feature crafting and data representation tuning through simultaneous representation learning and classifier optimization [10]. Convolutional neural network [6] and its combination with recurrent networks (DeepConvLSTM) [17] have shown notable performance in capturing spatial-temporal features from the sensor signal data.

Furthermore, the utilization of attention mechanism for HAR has been explored in [16, 8] by combining it with recurrent networks. In particular, the DeepConvLSTM architecture proposed in [17] has been augmented with an attention layer in [16]. This layer learns parameters to compute the relative weights for the hidden state outputs of the preceding LSTM layer. Attention layer is used to create context vector using linear combination of past and current hidden states in contrast to [17] which uses the last hidden state as context. Integration of continuous temporal and modality attention with LSTM has been proposed in [27]. In the same way, the augmentation of attention in two capacities [15] is proposed to compute the relative weight of sensor modality for specific activity window and to encapsulate the temporal context of the salient features of specific sensor signal. This approach, based on attention augmented GRU and ConvNet architecture, makes use of overlapping sliding window of Fast Fourier Transformed spectrogram from sensor signals. This recurrent architecture based attention model [15] is referred to as AttnSense and obtains notable performance in temporal context capturing. In this regard, the existing attention models for HAR exhibits notable performance in adapting inter and intra activity class variance with adaptive duration of attention within activity sequence. Zheng et al. [30] has proposed uniqueness attention based LSTM architecture which captures atomic features in temporal context. However, no architecture has been proposed yet which incorporates self-attention to capture spatial context of the feature sequence along with temporal context capturing.

## 3 Proposed Self-attention Model

Our objective is to build a self-attention based model without any recurrent architectures. Hence, the proposed model foregoes recurrent networks and utilizes sensor modality attention, self attention blocks, global temporal attention to construct feature representation used for classification as illustrated in Figure 1. We briefly describe the model architecture below and provide detailed specification in the subsequent section.

The input to the model is a time-window of sensor values. Firstly, sensor modality attention is applied to the inputs to get a weighted representation of the sensor values according to their attention score. Thus, the learned attention score represents the contribution of each of the sensor modalities in the feature representation used by the subsequent layers. Afterwards, we convert the weighted sensor values to $d$ size vectors using 1-D convolution over single time-steps. Similar to [25], we encode positional information of the samples in the sequence by adding values based on sine and cosine functions to the

obtained $d$ size vectors. This enables the model to take the temporal order of samples into account. This representation is scaled by $\sqrt{d}$ and passed to the self attention blocks. Self attention blocks use dot product-based attention score to transform the feature value for each time step. The representation generated from the self attention blocks is used by global temporal attention layer. As shown in Figure 1, this layer learns parameters to set varying attention across the temporal dimension to generate the final representation which is used by the final fully connected and softmax layers. We discuss details of sensor modality attention, self attention blocks, global temporal attention modules in the following subsections.

### 3.1 Sensor Modality Attention

To capture the varying levels of contribution from sensors at different modalities for classification, we use the sensor modality attention layer. For example, in order to recognize the activity 'ironing', the sensors placed at the subject's ankle do not provide much meaningful information. Sensor attention layers learn such relationships by using 2-d convolution across time-step and sensor values to capture their dependencies.

Firstly, the input is reshaped to produce single channel image. Then, $k$ convolutional filters are applied to the input with padding which outputs image with k channels. This is then converted back to a single channel by applying $1 \times 1$ convolution. Sensor-wise softmax as defined in (1) provides the attention score for individual sensors. In addition to providing a weighted version of the input according to their learned importance for the self attention layer, this mechanism allows us to plot feature maps making the model more interpretable.

$$s_{\kappa}^{(t_i)} = \frac{\exp(q_{\kappa}^{(t_i)})}{\Sigma_{\kappa} \exp(q_{\kappa}^{(t_i)})} \tag{1}$$

$\kappa$ in (1) indicate individual sensors.

### 3.2 Self Attention Block

Each block consists of two sub layers - multi-headed self attention and position-wise feed forward layer. Self attention is used to determine relative weights for each time-step in the sequence by considering its similarity to all the other time-steps. Subsequently, these relative weights are used to transform the representation of each time-step with relevant information from other time-steps according to their importance.

$$f_{sa}^{(h_j)}(\mathbf{q}, \mathbf{k}, \mathbf{v}) = \left[ softmax\left(\frac{\mathbf{q} \cdot \mathbf{k}^T}{\sqrt{d_k}}\right)\mathbf{v} \right] \tag{2}$$

The terms (q, k, v) in (2) are learned linear transformation of the input to the layer and referred as key, query and value respectively. In this regard, the query can be considered to the transformed vector of a particular time-step that is compared to the key vector of every other time-step using dot product. Afterwards, the dot product value is scaled and softmax normalized which indicates the attention scores. Finally, the attention values are used to get a weighted representation of the value vectors for each of the time-steps. However, the entire operation is implemented as a matrix multiplication operation as indicated in (2).

Moreover, we utilize multi-headed self attention since different attention heads are able to capture distinct aspects of the input signal. In this regard, $h_j$ in (2) represents output from attention head $j$. For computing the key, query and value used in (2), each one of the $n$ attention heads use distinct parameters. The outputs from the distinct attention heads are concatenated and converted back to the dimension of single attention head using learned parameter $W_o$ as defined in (3).

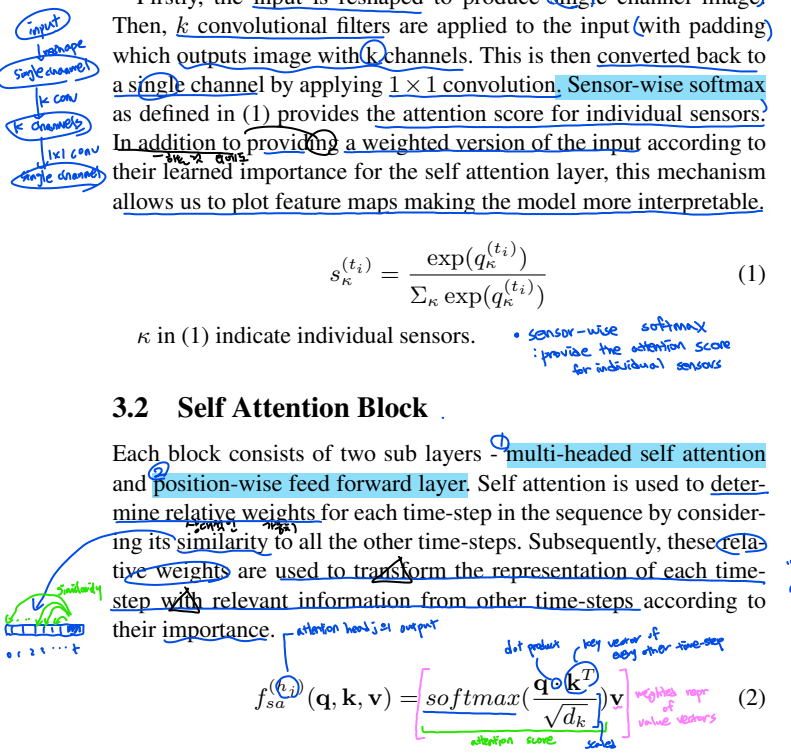



**Figure 1**: Attention based model incorporating self-attention and global temporal attention

$$\mathbf{s}_{mha} = \mathbf{W}_o \cdot \text{concat}(\mathbf{f}_{sa}^{(h_1)}, \ldots, \mathbf{f}_{sa}^{(h_{n-1})}, \mathbf{f}_{sa}^{(h_n)}) \tag{3}$$

Position-wise feed forward layer is applied independently to each position in a block. In this case, the weights are for each position in a block but different across the blocks.

Each of the sub-layer contains a residual connection and is followed by layer normalization.

### 3.3 Global Temporal Attention

We use the representation generated by the self attention blocks for each time-step and learn parameters to rank them according to their respective importance for predicting the corresponding class label for the window. The ranking (attention score) obtained in (5) is used to create a weighted average of the representations of all the time-steps in an activity "window" which is used as feature vector by the feed forward layers for classification.

$$\mathbf{g}^{(t_i)} = \tanh\left(\mathbf{W}_{ga} \cdot \mathbf{s}^{(t_i)} + \mathbf{b}_{ga}\right) \tag{4}$$

$$\alpha^{(t_i)} = \frac{\exp((\mathbf{g}^{(t_i)})^T \cdot \mathbf{g}_s)}{\sum_t \exp(\mathbf{g}^{(t_i)} \mathbf{g}_s)} \quad (5)$$

$$\mathbf{c}_i = \sum_t \alpha^{(t_i)} \mathbf{s}^{(t_i)} \quad (6)$$

The terms $W_{ga}$ and $b_{ga}$ in (4) refer to parameters learned during training to get a hidden representation from each of the vectors $s^{(t_i)}$ generated from self attention blocks. The parameter $g_s$ in (5) helps to capture temporal context while learning to compute the attention score. A weighted summation according to the relative importance of respective time steps is generated as the feature vector in (6).

For regularization, dropout has been used in the self attention blocks, the fully connected layers, and after the addition of positional encoding.

## 4 Dataset Description

We use four commonly used benchmark datasets [10] to evaluate the performance of our model and to compare it with that of state of the art models. However, we did not use Daphnet Freezing of Gait Dataset [1] as this is particular to specific gait recognition for patients with Parkinson's disease. Below We give brief description of the datasets used in our experiments.

**PAMAP2** dataset [21] incorporates the hardware setup of 3 Inertial Measurement Units (IMU) placed over the wrist of dominant arm, on chest and at ankle and the data has been sampled at the frequency of 100Hz. The whole data included annotated human activity class of 9 subjects with particular physical description. Majority of the subjects are male with right dominant hand. In fact, PAMAP2 contains only one female subject and one left handed subject with id 102 and 108 respectively. This benchmark dataset contains 18 human activity classes altogether. In our experiments, data from one of the accelerometers ($\pm$16g scale) and gyroscope contained in each IMU have been used.

**OPPORTUNITY** dataset [22] includes the annotated data of body-worn sensors and ambient sensors to specify particular human activity. The dataset has been formed incorporating the reading of motion sensors and classified with "modes of locomotion". The sensors have been able to capture 5 high-level human activity classes along with 17 mid-level gesture classes and 13 low-level actions. We focus on the mid-level gestures and remaining are considered null class which comprises more than 75% of the data making the dataset highly imbalanced in terms of class distribution.

**USC-HAD** dataset [28] incorporates six readings from body-worn 3-axis accelerometer and gyroscope sensor through Motion-Node device. The dataset has been created with equally distributed (7 each) 14 male and female subjects with defined physical specification and age. The sampling rate of sensor data is 100 Hz and includes one of 12 activity class labels for each time-step in the dataset.

USC-HAD dataset poses an inherent challenge in feature representation learning and segmentation due to the sensor placement and variation in the activity classes. Here, the single accelerometer and gyroscope reading is obtained from the motion node attached to the right hip of specific subject and thus does not contribute much in the feature space transformation. Moreover, the activity classes involve orientation such as walking forward or left or right and even elevator up or down which are generally not captured only through accelerometer and gyroscope reading.

**SKODA** dataset [23] is a special purpose dataset to track the activity of workers in the manufacturing assembly-line scenario. This dataset incorporates accelerometer reading from 10 different positions on the subject's arms and is labeled with specific activity class including a null class. Following the standard procedure, we use 80% of the data for training and 10% for validation and test respectively.

The benchmark test subjects and the summary of the datasets have been included in Table 1.

## 5 Experiment Setup

In this section we describe the preprocessing of the datasets, the architecture of the models, evaluation procedure, and performance measures used in our experiments.

### 5.1 Preprocessing

Since the datasets involved in the experiments have varied sampling rates, alignment of the frequencies through downsampling facilitates reasonable comparison of performance. Similar to the previous works in [10] and [27], we down-sampled PAMAP2, USC HAD and Skoda to close 30 Hz to align with the Opportunity dataset.

**Window based representation:** The proposed approach utilizes sliding window based feature extraction. Window size is the number of samples that is considered at a time to construct a feature representation used for classification. The activities under consideration are diverse in terms of duration and complexity which makes the choice of window size an important hyper-parameter. Likewise, the choice of how much overlap there should be between the consecutive windows is also an important factor to consider.

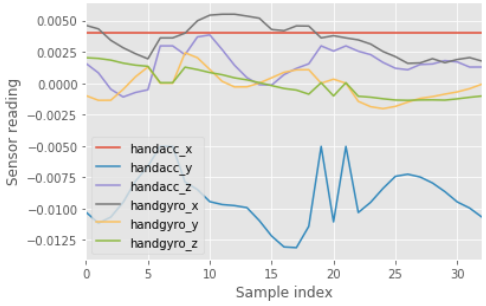



**Figure 2**: Activity window for walking activity in PAMAP2 dataset where timespan = 1 Sec

The activity recognition window is constructed like an image with time-steps and heterogeneous sensor readings as the two dimensions. The activity label is determined through majority voting in the samples constituting the window. The activity window with specific class label has been demonstrated in Figure 2.

The time-series sensor data indicates activities spanning varied time-span and to capture this sequence within the activity recognition window, the sliding of windows is done with a fixed percentage of overlap. The percentage of overlap has been tuned as a hyperparameter and deployed in the training accordingly.

### 5.2 Implementation of Existing Architectures

We perform extensive experiments with the Convolutional Autoencoder (ConvAE) and Deep Convolutional LSTM (DeepConvLSTM) [17]. Experiments presented in [10] show that these models perform well for different benchmark datasets. We also report the experimental results for newly published attention based HAR models from the

**Table 1**: Summary of experimental setup for the datasets. Here A = Accelerometer, G = Gyroscope, M = Magnetometer

| Dataset | Number of Activities | Benchmark Test Subject ID | Down-sampling | Sliding Window Overlap | Sensors Used |
|---|---|---|---|---|---|
| PAMAP2 | 12 | 106 | 1/3 | 50% | A, G |
| Opportunity | 18 | 2, 3 (Run 4 & 5) | 1 | 50% | A, G, M (upper body) & sensors in shoes |
| USC-HAD | 12 | 13, 14 | 1/3 | 50% | A, G |
| Skoda | 11 | 1 | 1/3 | 50% | A |

respective papers. Below we provide detailed description of the ConvAE and DeepConvLSTM models specified in [10] pertaining to our experiments.

**Convolutional Autoencoder (ConvAE)** includes an encoder and a decoder part with a bottleneck layer in between. The encoder consists of four convolutional blocks each containing two $3 \times 3$ convolution layers with the same number of filters followed by batch normalization. Each of the convolutional blocks contain a $2 \times 2$ max-pooling operation at the end. The output from the last convolutional block is flattened and passed to a fully connected bottleneck layer. The feature vector from the bottleneck layer is used by the decoder to reconstruct the input by inverting the encoding process sequentially. We followed the description of the decoder in [10] and used up-sampling, convolution and appropriate padding or cropping to match the input dimension during the inversion process. Similar to the encoder, we used the same number of $3 \times 3$ filters in the four blocks of the decoder. We used relu activation throughout the model and hyperbolic tangent activation for the output. The feature representation from the bottleneck layer is used by Multi-layer Perceptron (MLP) [10] to classify the activity label for the respective input window. The dimension of the bottleneck layer has been set to 500, 1000, 1500 and 2000 for PAMAP2, Opportunity, Skoda and USC HAD respectively since the best results for the respective datasets have been reported at that particular dimension in [10].

**DeepConvLSTM** has four successive convolution layers and two layers of LSTMs. Each convolutional layer has 64 filters with size of $5 \times 1$. The $5 \times 1$ filter is used to perform convolution along the time-steps. Using the $5 \times 1$ filter, multiple sensor information are kept separate. The output from the first convolution layer is fed to the 2nd convolution layer and so on. Then the output of the final convolution layer is applied to a two-layer LSTM, each with 128 hidden units. The final output vector is connected to a fully connected layer. After performing the softmax operation on fully connected layer output, activity class probability is available in the final output of the model. We use a dropout with probability of 0.5 in the fully connected layer.

Detailed description of the architectures of the attention based models namely, DeepConvLSTM with Attention, LSTM with continuous Attention and AttnSense can be found in [16, 27, 15] respectively.

## 5.3 Training and Test Procedures

For the segmentation of activity data, we have the choice of predicting class label each individual sample in a sequence or for a fixed-length time window. However, we need to analyze a sequence window of some length in both cases.

**Sample-wise:** During training, we slide the window by one time-step forward and provide the ground truth label for each time step. Then we slide the window right by one time step. During the testing, we follow the same technique. We take the output label of the window and set this output to corresponding last time-step of the window. Hence, we obtain sample-wise output during test.

**Window-wise:** We create the window with predefined window size and continue to slide the window with $50\%$ overlap. During the training, we will assign the most frequent activity in this window as the ground truth label of that window. In testing, the model produces one output label for each window. For test, no overlap is used. In the case where a window contains samples with a different label, we pad the window by repeating the last few samples and the next window starts from the differently labeled sample.

**Training and Hyperparameters:** For the proposed model, we set the number of self attention blocks to 2 for all of the datasets except USC-Had where 3 blocks are used. For construction of fixed size input for self attention as described in Section 3, $d$ was set to 128. Similar to [25], the number of units in position-wise feed forward layer was set to 4 times $d$. We used Adam optimizer with the default parameters discussed in [12] and learning rate $0.001$ for training of the models.

## 5.4 Evaluation Metric

We use macro average F1-score as metric to compare the performance of the proposed approach with other methods. In this regard, we calculate F1-score for each class according to (7) as follows:

$$\text{Macro F1-Score} = \frac{1}{|C|} * \sum_{i=1}^{C} \frac{2 * Precision_i * Recall_i}{Precision_i + Recall_i} \tag{7}$$

Here, $|C|$ in 7 indicates number of classes and F1-score for each class is given the same weight irrespective of their number of instances and $i = 1, ..., C$ is considered as the set of classes considered for experiment.

# 6 Results

In this section, we present the results of the experiments as described in the previous section. First, we discuss the performance on the benchmark test subjects for the datasets. Then, we describe the Leave One Subject Out Cross Validation (LOSO-CV) results for all subjects. We conclude this section with discussion of the results from window size variation experiments and the attention maps that we generate from the sensor attention layer.

**Performance on benchmark test subject:** Table 2 shows the performance comparison between the proposed model and the existing models for the benchmark test subjects described in Table 1.

**Table 2**: Macro F1-score for benchmark test subject

| | Architecture | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Proposed Model | | DeepConvLSTM | | ConvAE | | DeepConvLSTM + Attention | LSTM + Continuous Attention | AttnSense |
| Dataset | ◇ | □ | ◇ | □ | ◇ | □ | ◇ | ◇ | ◇ |
| PAMAP2 | 0.95 | 0.96 | 0.71 | 0.70 | 0.52 | 0.80 | 0.88 | 0.90 | 0.89 |
| Opportunity | 0.61 | 0.67 | 0.66 | 0.58 | 0.60 | 0.60 | 0.71 | - | - |
| USC-HAD | 0.50 | 0.55 | 0.42 | 0.38 | 0.42 | 0.46 | - | - | - |
| Skoda | 0.93 | 0.97 | 0.96 | 0.88 | 0.82 | 0.79 | 0.91 | 0.94 | 0.93 |

◇ Sample-wise □ Window-wise

**Table 3**: Average Macro F1-score for leave one subject out experiment

| | Architecture | | | | | |
|---|---|---|---|---|---|---|
| | Proposed Model | | DeepConvLSTM | | ConvAE | |
| Dataset | ◇ | □ | ◇ | □ | ◇ | □ |
| PAMAP2 | 0.92 | 0.96 | 0.61 | 0.52 | 0.47 | 0.48 |
| Opportunity | 0.39 | 0.42 | 0.44 | 0.41 | 0.41 | 0.42 |
| USC-HAD | 0.60 | 0.67 | 0.59 | 0.50 | 0.58 | 0.63 |

◇ Sample-wise □ Window-wise

With regards to the sample-wise classification scores, our proposed model achieves significant improvement over DeepConvLSTM and ConvAE for PAMAP2 and USC-HAD. However, our model obtains slightly lower sample-wise score for Opportunity and Skoda with DeepConvLSTM. Specifically, F1 macro has been decreased to 0.61 from 0.66 (DeepConvLSTM) for Opportunity dataset. Since the proposed model is fundamentally designed for window based output, we notice the significant improvement while we perform window-based tests for Opportunity.

However, we can observe more obvious and significant improvement in terms of the window-wise scores for the proposed model. In particular, the window-wise macro F1-score has been improved to 0.67 from 0.58 (DeepConvLSTM) and 0.60 (ConvAE) for the Opportunity dataset. Thus, it can be noted that our model works more accurately (0.67) on the Opportunity (datasets containing complex activities) compared to other methods. In terms of the window-based scores, our model also outperforms other models. Therefore, we can conclude that the proposed model can better capture the spatio-temporal characteristics of sensor-data more effectively than the DeepConvLSTM and ConvAE.

As discussed in Section 4, USC-HAD is a particularly challenging dataset due to the sensor setup. However, our model performs better (0.50 sample-wise and 0.55 window-wise) than the other models (DeepCpnvLSTM: 0.42 & 0.38; ConvAE: 0.42 & 0.46).

It is evident from the data in Table 2 that the proposed model also performs better than other attention based models: DeepConvLSTM + Attention, LSTM + Continuous Attention [27] and AttnSense [15] for PAMAP2 benchmark test set. We only compare the sample-wise results as the aforementioned models published sample-wise ones only. The macro F1-score for our model for PAMAP2 is higher than the corresponding scores for the other attention-based models e.g DeepConvLSTM + Attention, LSTM + Continuous Attention, and AttnSense. For Skoda, our model also outperformed (0.93) DeepConvLSTM + Attention (0.91) and equally performed with AttnSense(0.93). However, our model consistently outperformed the other attention-based models in terms of window-wise test scores on all the datasets considered except Opportunity.

**Performance on LOSO-CV:** In order to demonstrate the robustness of the proposed model in terms of sensitivity to specific test subjects, we conduct LOSO-CV experiments for each dataset containing activities of multiple subjects. In this regard, we hold the data from one of the subjects out during training and use that data for evaluating the model. This process is repeated for each subject for the particular dataset and the average score is reported. Note that, Skoda contains activities performed by only one subject and is excluded from these experiments.

As can be seen from Table 3, the macro F1-scores for the proposed method are consistently higher for both sample-wise and window-wise tests compared to the corresponding scores for DeepConvLSTM and ConvAE for LOSO-CV experiments. Thus, the results indicate that the proposed method is capable of modeling the inter-subject variability better. In other words, our model has more generalization capability than the others.

Specifically, LOSO-CV experiments with the PAMAP2 dataset shows that the proposed model significantly outperforms the other models under comparison for subject 102 (female subject). In this regard, the proposed model obtains F1-scores 0.93 (sample-wise) and 0.98 (window-wise) respectively. On the other hand, DeepConvLSTM achieves F1-scores of 0.47 (sample-wise) and 0.33 (window-wise) in the LOSO-CV experiment involving this subject. For ConvAE, the corresponding score is 0.35 in both cases.

Moreover, for subject 108 (male, left-handed), our model achieves F1 scores of 0.79 (sample-wise) and 0.88 (window-wise) whereas DeepConvLSTM gets 0.27 and 0.28, respectively. ConvAE performs slightly better than DeepConvLSTM, the scores for ConvAE are 0.40 (sample-wise) and 0.47 (window-wise).

**Performance of proposed model on window sizes:** As different activities have different repetitive periods, we conducted experiments to analyze the impact of window-size variations on the proposed model's performance. In this regard, we train different models while varying the window-size and use the benchmark test subjects defined in Table 1 as the test set. Figure 3 demonstrates the change in performance for different window-size and it can be concluded from the figure that the proposed model is less sensitive to changes in window-size than the other models in terms of performance. It is evident that datasets involving complex activities require relatively longer time-span for sliding window for capturing correct activity label.

**Feature Map for Sensor Modality Attention:** Sensor modality attention layer described in Section 3.1 has been utilized to deter-

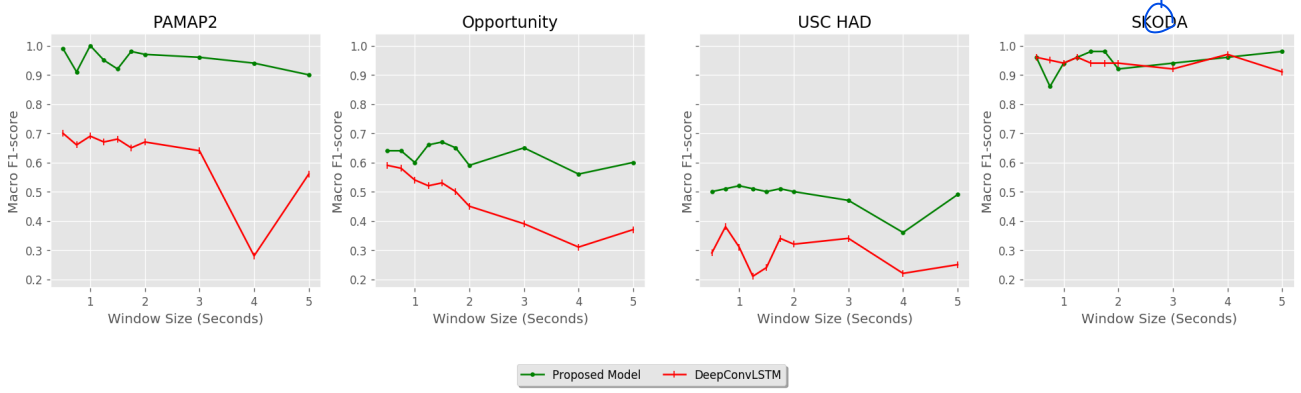
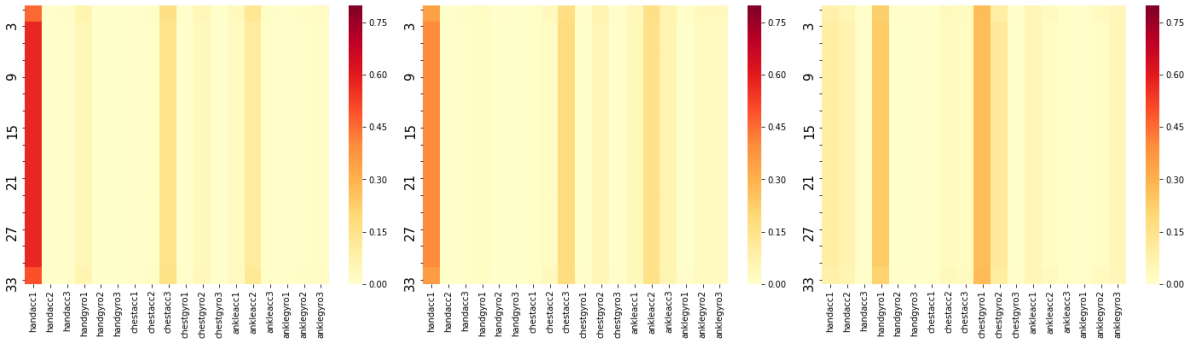
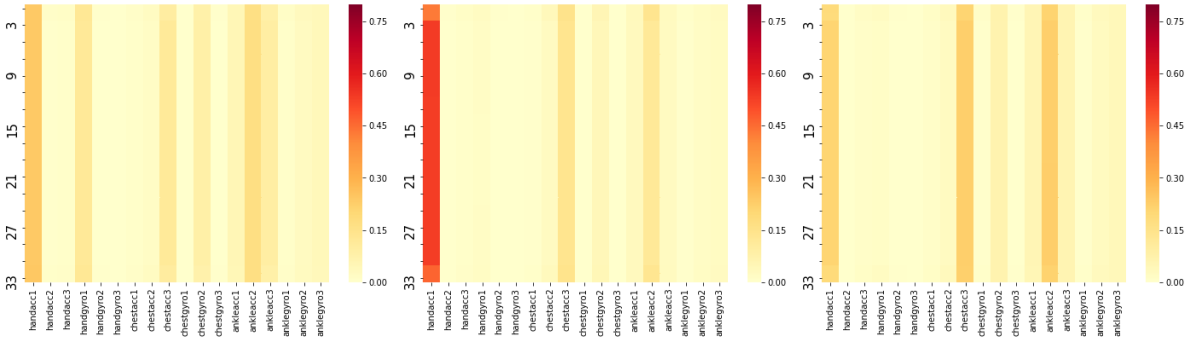



**Figure 3**: Performance measure against different window sizes



(a) Ironing (b) Ascending stairs (c) Lying



(d) Cycling (e) Vacuum cleaning (f) Walking

**Figure 4**: Attention weights on different sensor modality based on predicted class label in PAMAP2 dataset (e.g. Ironing involves higher attention weight for hand accelerometer, moderate attention for chest accelerometer and relatively low weights for other sensor placements)

mine the impact of sensors' placements on the classification task. In Figure 4, feature attention maps incorporate average attention on specific sensor modality listed in the x - axis over all activity window. The vertical axis indicates timestep within a specific window. If we consider the feature maps visualizing attention weights on sensor modality, it can be derived that while ironing, sensors placed at hand get greater attention weights which is visualized in Figure 4a. On the other hand, during ascending stairs, hand accelerometer and ankle accelerometer obtain relatively higher attention weight in feature segmentation which is evident in Figure 4b. Moreover, the attention map in Figure 4c exhibits that gyroscope placed at chest obtains higher attention weight than other sensor placements. The activity cycling involves simultaneous movement of different body parts which is captured through sensor modality attention and evident in the attention weight distribution in Figure 4d. Vacuum cleaning activity in 4e indicates that hand is the dominant body part in detecting this particular activity. The attention map illustrates similar weight distribution as ascending stairs in the case of walking.

Figure 4 demonstrates the higher emphasis on particular sensor modality in predicting class label. Here, the proposed model automatically distributes attention weight on heterogeneous sensor modalities and this weight is intuitively explainable with respect to the predicted class label.

## 7 Conclusion

In this paper we propose a self-attention-based deep learning architecture for Human Activity Recognition (HAR). The model is

adapted from the transformer architecture proposed for machine translation. The proposed model foregoes recurrent layers and utilizes attention mechanisms to generate feature representation used for classification. We perform experiments on leave one subject out cross validation on four benchmark datasets - PAMAP2, Opportunity, Skoda, and USC-HAD. We perform both sample-wise and window-wise classification. Compared with existing state-of-the art methods, we show that our proposed attention based model outperforms existing models in case of the benchmark test subject for all datasets except Opportunity for sample wise classification. In case of window-wise classification our model outperforms Deep Convolutional LSTM and Convolutional Autoencoder models. One limitation of our experiments is that we did not perform window-wise classification on the newly published models. In future, we intend to extend our model with a decoder network and perform more extensive experiments to compare with all existing models.

## Acknowledgements

This work is supported by grants from ICT Division, Government of Bangladesh and Independent University, Bangladesh.

## REFERENCES

[1] M. Bachlin, M. Plotnik, D. Roggen, I. Maidan, J. M. Hausdorff, N. Giladi, and G. Troster, 'Wearable assistant for parkinsons disease patients with the freezing of gait symptom', *IEEE Transactions on Information Technology in Biomedicine*, **14**(2), 436–446, (March 2010).

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, 'Neural machine translation by jointly learning to align and translate', *CoRR*, **abs/1409.0473**, (2015).

[3] Ling Bao and Stephen S Intille, 'Activity recognition from user-annotated acceleration data', in *International conference on pervasive computing*, pp. 1–17. Springer, (2004).

[4] Davide Figo, Pedro C. Diniz, Diogo R. Ferreira, and João M. Cardoso, 'Preprocessing techniques for context recognition from accelerometer data', *Personal Ubiquitous Comput.*, **14**(7), 645–662, (October 2010).

[5] Sojeong Ha, Jeong-Min Yun, and Seungjin Choi, 'Multi-modal convolutional neural networks for activity recognition', in *2015 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 3017–3022. IEEE, (2015).

[6] Nils Y. Hammerla, Shane Halloran, and Thomas Plötz, 'Deep, convolutional, and recurrent models for human activity recognition using wearables', in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pp. 1533–1540. AAAI Press, (2016).

[7] Nils Y Hammerla, Reuben Kirkham, Peter Andras, and Thomas Ploetz, 'On preserving statistical characteristics of accelerometry data using their empirical cumulative distribution', in *Proceedings of the International Symposium on Wearable Computers*, pp. 65–68. ACM, (2013).

[8] M. N. Haque, M. Tanjid Hasan Tonmoy, S. Mahmud, A. A. Ali, M. Asif Hossain Khan, and M. Shoyaib, 'Gru-based attention mechanism for human activity recognition', in *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*, pp. 1–6, (May 2019).

[9] Md Nazmul Haque, Mahir Mahbub, Md Hasan Tarek, Lutfun Nahar Lota, and Amin Ahsan Ali, 'Nurse care activity recognition: A gru-based approach with attention mechanism', in *Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers*, pp. 719–723. ACM, (2019).

[10] Harish Haresamudram, David V. Anderson, and Thomas Plötz, 'On the role of features in human activity recognition', in *Proceedings of the 23rd International Symposium on Wearable Computers*, ISWC '19, pp. 78–88, New York, NY, USA, (2019). ACM.

[11] Tâm Huynh and Bernt Schiele, 'Analyzing features for activity recognition', in *Proceedings of the 2005 Joint Conference on Smart Objects and Ambient Intelligence: Innovative Context-aware Services: Usages and Technologies*, sOc-EUSAI '05, New York, NY, USA. ACM.

[12] Diederik P. Kingma and Jimmy Ba, 'Adam: A method for stochastic optimization', *CoRR*, **abs/1412.6980**, (2015).

[13] Hyeokhyen Kwon, Gregory D Abowd, and Thomas Plötz, 'Adding structural characteristics to distribution-based accelerometer representations for activity recognition using wearables', in *Proceedings of the 2018 ACM International Symposium on Wearable Computers*. ACM.

[14] Zhouhan Lin, Minwei Feng, Cícero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio, 'A structured self-attentive sentence embedding', *CoRR*, **abs/1703.03130**, (2017).

[15] HaoJie Ma, Wenzhong Li, Xiao Zhang, Songcheng Gao, and Sanglu Lu, 'Attnsense: Multi-level attention mechanism for multimodal human activity recognition', pp. 3109–3115, (08 2019).

[16] Vishvak S Murahari and Thomas Plötz, 'On attention models for human activity recognition', in *Proceedings of the 2018 ACM International Symposium on Wearable Computers*, pp. 100–103. ACM, (2018).

[17] Francisco Ordez and Daniel Roggen, 'Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition', *Sensors*, **16**, 115, (01 2016).

[18] Madhuri Panwar, Dwaipayan Biswas, Harsh Bajaj, Michael Jobges, Ruth Turk, Koushik Maharatna, and Amit Acharyya, 'Rehab-net: Deep learning framework for arm movement classification using wearable sensors for stroke rehabilitation', *IEEE Transactions on Biomedical Engineering*, **PP**, 1–1, (02 2019).

[19] Thomas Plötz, Nils Y Hammerla, and Patrick L Olivier, 'Feature learning for activity recognition in ubiquitous computing', in *Twenty-Second International Joint Conference on Artificial Intelligence*, (2011).

[20] Bahareh Pourbabaee, Mehrsan Javan Roshtkhari, and Khashayar Khorasani, 'Deep convolutional neural networks and learning ecg features for screening paroxysmal atrial fibrillation patients', *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, **48**(12), (2017).

[21] Attila Reiss and Didier Stricker, 'Introducing a new benchmarked dataset for activity monitoring', in *Proceedings of the 16th Annual International Symposium on Wearable Computers (ISWC)*, pp. 108–109, Washington, DC, USA, (2012). IEEE Computer Society.

[22] D. Roggen, A. Calatroni, M. Rossi, T. Holleczek, K. Frster, G. Trster, P. Lukowicz, D. Bannach, G. Pirkl, A. Ferscha, J. Doppler, C. Holzmann, M. Kurz, G. Holl, R. Chavarriaga, H. Sagha, H. Bayati, M. Creatura, and J. d. R. Milln, 'Collecting complex activity datasets in highly rich networked sensor environments', in *Seventh International Conference on Networked Sensing Systems (INSS)*, pp. 233–240, (2010).

[23] Thomas Stiefmeier, Daniel Roggen, G. Troster, Georg Ogris, and Paul Lukowicz, 'Wearable activity tracking in car manufacturing', *Pervasive Computing, IEEE*, **7**, 42–50, (05 2008).

[24] Ilya Sutskever, Oriol Vinyals, and Quoc V Le, 'Sequence to sequence learning with neural networks', in *Advances in neural information processing systems*, pp. 3104–3112, (2014).

[25] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin, 'Attention is all you need', in *Advances in Neural Information Processing Systems 30*, 5998–6008, Curran Associates, Inc., (2017).

[26] Shuochao Yao, Shaohan Hu, Yiran Zhao, Aston Zhang, and Tarek Abdelzaher, 'Deepsense: A unified deep learning framework for time-series mobile sensing data processing', in *Proceedings of the 26th International Conference on World Wide Web*, WWW '17, pp. 351–360, Republic and Canton of Geneva, Switzerland, (2017). International World Wide Web Conferences Steering Committee.

[27] Ming Zeng, Haoxiang Gao, Tong Yu, Ole J. Mengshoel, Helge Langseth, Ian Lane, and Xiaobing Liu, 'Understanding and improving recurrent networks for human activity recognition by continuous attention', in *Proceedings of the 2018 ACM International Symposium on Wearable Computers*, ISWC '18, pp. 56–63, NY, USA. ACM.

[28] Mi Zhang and Alexander Sawchuk, 'Usc-had: a daily activity dataset for ubiquitous activity recognition using wearable sensors', pp. 1036–1043, (09 2012).

[29] Songyang Zhang, Yang Yang, Jun Xiao, Xiaoming Liu, Yi Yang, Di Xie, and Yueting Zhuang, 'Fusing geometric features for skeleton-based action recognition using multilayer lstm networks', *IEEE Transactions on Multimedia*, **20**(9), 2330–2343, (2018).

[30] Zengwei Zheng, Lifei Shi, Chi Wang, Lin Sun, and Gang Pan, *LSTM with Uniqueness Attention for Human Activity Recognition*, 498–509, 09 2019.