GLULA:Linear Attention Based Model for Efficient Human Activity Recognition from Wearable Sensors and Skeleton Data

by

Aldiyar Bolatov

Submitted to the Department of Computer Science or Data Science
in partial fulfillment of the requirements for the degree of

Master of Science in Computer Science or Data Science

at the

NAZARBAYEV UNIVERSITY

July 2023

Author .....
Department of Computer Science or Data Science
26.07.2023

Certified by .....
Adnan Yazici
Full Professor
Thesis Supervisor

Accepted by .....
Vassilios D. Tourassis
Dean, School of Engineering and Digital Sciences

GLULA:Linear Attention Based Model for Efficient Human Activity Recognition from Wearable Sensors and Skeleton Data

by

Aldiyar Bolatov

Submitted to the Department of Computer Science or Data Science
on 26.07.2023, in partial fulfillment of the
requirements for the degree of
Master of Science in Computer Science or Data Science

Abstract

Sensors' data is used in monitoring patient activity during rehabilitation and also can be extended to controlling rehabilitation devices based on the activity of the person. Both wearable sensors and extracted skeleton data from the video can be used for that. As there, exist similarities, a unified solution can be presented, which also focuses on effectively capturing the spatiotemporal dependencies in the data collected by these sensors and efficiently classifying human activities. With the increasing complexity and size of models, there is a growing emphasis on optimizing their efficiency in terms of memory usage and inference time for real-time usage and mobile computers. There is an opportunity to develop a novel unified framework that incorporates recent advancements to enhance speed and memory efficiency, specifically tailored for Human Activity Recognition (HAR) tasks. In line with this approach, we present GLULA, a unique architecture for human activity recognition. GLULA combines gated convolutional networks, branched convolutions, and linear self-attention to achieve efficient and powerful solutions. Extensive experiments showed its effectiveness both in wearable sensors' data and skeleton-based sets. Tests were conducted on five benchmark IMU datasets: PAMAP2, SKODA, OPPORTUNITY, DAPHNET, and USC-HAD. Our findings demonstrate that GLULA outperforms recent models in the literature on the latter four datasets but also exhibits the lowest parameter count among state-of-the-art models. In HAR for the human skeleton domain, examinations were done on the NTU RGB+D dataset. While getting comparable results with recent work in this field, it managed to be smaller and significantly faster.

Thesis Supervisor: Adnan Yazici
Title: Full Professor

Acknowledgments

To begin with, I want to acknowledge and express my appreciation to my advisor, Professor Adnan Yazici. His guidance was pivotal from the outset and remained consistent throughout the entirety of the research journey. I also want to express my sincere thanks to my co-worker, Dias Ainyshev. Our collaborative discussions bore many research concepts that became instrumental parts of the thesis. Lastly, a special note of thanks goes to Professor Adnan Yazici and the Provost's Office for supplying us with an appropriate environment and necessary equipment to conduct our research.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The increasing prevalence of sensor devices due to their affordability and advancement in motion analysis technologies have sparked an interest in the utilization of those aforementioned systems. One of the main possible applications is human activity recognition (HAR). While activity is a broad term, it can encapsulate diverse areas such as fitness monitoring and drug control systems [14], as well as stress and affect detection [28]. Furthermore, it can find even more applications in health-related topics such as monitoring patient activity during rehabilitation and also can be extended to controlling rehabilitation devices based on the activity of the person [2]. Each of these applications demands a robust, accurate, and real-time activity classification mechanism to ensure seamless integration and functionality.

Human activity recognition on sensor data can be based on various data types and systems such as cameras, wearable sensors, and depth measurement equipment. For example, body-worn devices, including smartphones, watches, and other wearable technologies, are equipped with accelerometers, gyroscopes, and other sensors that can capture motion signals. However, body-worn inertial sensors (IMUs) have emerged as an effective and ubiquitous tool for capturing complex human motions, and can be found in smartwatches and rehabilitation trackers [3]. These sensors, which include accelerometers and gyroscopes, are typically mounted on various joints of the human body, capturing movement patterns that can characterize a range of human activities. Inertial sensors are beneficial in HAR, since their ability to quantify the motion

of body parts in two to three-dimensional space, capturing the dynamic of human movement [3]. Thus, as one of the most common human body measurement devices is body-worn inertial sensors, it was chosen as one of the focus for this research.

Regarding output from camera sensors, another common sensor device, it is possible to perform action recognition directly on RGB frames or process data in such a way as to get more meaningful input for HAR systems. One of the most common ways to prepare data for action recognition is to extract human body joint data in the form of skeletons from the frames [6]. In this work, extracted body joint data from camera output was the focus for several reasons. First of all, while it haves drawbacks such as additional frame extraction processing, it has the advantage of higher controllability since it is feasible to manage inter-step between video data as an input and action prediction. The most important reasons however are: skeleton data have an advantage over RGB frames in terms of better robustness to image noises such as lighting, background, and perspective [6]; the skeleton data is similar in nature to data from body-worn sensors. The correlation in data structure exists since in both cases specific joints (via placed trackers or via extracted skeleton data) are tracked at a certain temporal frame, while the latter contains information on changes in position, the former posses information on exact position in space. Thus, there is a potential to unify a solution for human activity recognition on the two most prevalent data types used for this task.

Since both wearable sensors output data and extracted human joints in the form of the skeleton have a similar structure and lie in the temporal dimension, they had analogous development in human activity recognition research. As the deep learning research progressed, currently, in both domains, state-of-the-art results are based on the transformer architecture, which was introduced in [38].

In HAR systems for body-worn inertial sensors' data, the signal time series is divided into equal-length subsequences using the sliding window technique. These subsequences are then classified into activities using various algorithms, ranging from traditional machine learning approaches like Support Vector Machines and Random Forests to advanced neural networks such as Recurrent Neural Networks (RNN), Con-

volutional Neural Networks (CNN), or hybrid models. Notably, deep neural network models have demonstrated superior performance in activity classification compared to conventional machine learning algorithms in HAR [19]. The two most common solutions were CNNs and RNNs, where both had their advantages and disadvantages [19]. But overall, the hybrid solution came on top since it was based on Long Short-Term Memory (LSTM) networks, which solved the vanishing gradient problem of previous generation recurrent units [12] and utilized CNNs extraction ability, while avoiding the CNN problem of inability to encode spatial information by incorporating new representation in a temporal domain for LSTMs [47]. Although, hybrid solutions were still based on a recurrent mechanism. That meant a lack of parallelization of the input on the temporal dimension, which makes them slower than convolutional-based solutions that allow parallelization. With advances in natural language processing (NLP), Vaswani et al. presented a new fully-attention-based Transformer architecture [38]. The main part of the transformer is a non-recurrent multi-head self-attention mechanism, which makes it parallelizable in computing the importance of each point. To add temporal awareness to the network, they add positional encoding, which reduces the effect of the same problem existing in CNN-based solutions. Inspired by the transformer, Mahmud et al. utilized a self-attention-based neural network (almost identical to the original architecture) to solve HAR tasks for body-worn inertial sensors' data, which resulted in improved performance over state-of-the-art models [20].

The development of HAR solutions utilizing body skeleton data extracted from video sequences has made similar strides. Initially, vision-based HAR systems focused on interpreting human action by analyzing 2D skeletal data-frames using classical machine learning methods. The approach involves extracting the skeleton from the video, followed by spatiotemporal handcrafted feature extraction, and then feeding it to supervised machine learning for activity recognition [7]. Subsequently, advancements were made with the introduction of deep learning solutions such as CNNs and RNNs. As in HAR systems for body-worn inertial sensors' data, each had similar drawbacks and benefits. In due course, hybrid solutions were introduced. For example,

Convolutional Long Short-Term Memory models combined the capabilities of Convolutional Neural Networks and Long Short-Term Memory networks. This approach efficiently extracted spatiotemporal features for activity recognition, outperforming both LSTM and CNN models when used individually [45]. However, the transformer-based solution again came on top, they not only have the best performance but also allow parallelization for further speed optimization. One of the recent models that achieved state-of-the-art (SOTA) results utilize a hypergraph representation of the skeleton data in the Transformer [50]. To note, it was done on three-dimensional skeleton data extracted from video using the Kinect system.

As it was mentioned, due to the correlation between the two data representations of the activity and similar development in classification systems, it is possible to build a unified solution. In addition to achieving high accuracy in activity recognition, efficient resource utilization is essential in the field of HAR due to increasing computational demands. The reason is that there is a growing emphasis on optimizing models' efficiency in terms of memory usage and inference time for real-time usage and mobile computers used in health-related monitoring systems. While transformers provide parallelization due to non-recurrence, which is important in improving inference speed, they are computationally intensive having quadratic space and time complexity [13].

Taking everything into consideration, in this work, the novel proposed approach is an attention-based non-recurrent architecture that incorporates gated convolutional networks (GCN) [4], a branching convolution structure, and linear attention. In this model. first local information is extracted and added to the data and then fed to the global layers. This concept correlates with HAR tasks, where locality is present, but broad context is also essential. At the same time, the network does not reduce the dimension of the data, so it expands the information and not just extract it, which is advantageous for global parts. The GCN utilizes Gated Linear Units (GLU) as gate mechanisms, enabling convolutions' local learning and spatial awareness with no recurrent connections. As the global learning step, linear attention was chosen instead of the conventional softmax self-attention, to further enhance the model's

efficiency. As the name suggests, a linearized solution has linear time and space complexity with respect to the length of the sequence, instead of quadratic [13]. Moreover, it showed comparative results to full self-attention models in language-related task classification. This structure of the model helps in learning local and global spatiotemporal relationships effectively.

Due to the similar structure to a regular transformer, the proposed model can utilize both human skeleton data and body-worn inertial sensors' data in raw format. The only change needed to be done is to the dimensionality of the network depending on the samples. The attention-based linear network proposed for HAR tasks is called GLULA in this study. As discussed below, this model shows the best or most comparative results when compared to different variants and state-of-the-art models while maintaining its advantages in size, space, and time complexities. Precisely it has improved performance in datasets collected from IMUs and comparative results on skeleton sets. The contributions of the paper are as follows:

- Introduction of the novel attention-based network architecture for human activity recognition (HAR) that offers several key contributions. Firstly, the model is parallelizable and achieves the lowest parameter count with a noticeable difference compared to other state-of-the-art solutions. Also, optimize space and time complexity through the utilization of linear attention. This architecture demonstrates superior performance on four body-worn inertial sensors' HAR datasets (SKODA, OPPORTUNITY, DAPHNET, USC-HAD) compared to recent models in the literature, and comparable results on the PAMAP2 dataset, validating the effectiveness of the proposed solution.

- Additionally, the conduct of a comprehensive comparison of various layers at different positions within the proposed network. Experiments illustrate that the network structure outperforms the different variants presented in this paper. When comparing the softmax self-attention unit [38] with linear attention and gated convolutional networks (GCN), it is observed that while softmax self-attention may have higher complexity, replacing linear attention or GCN with

softmax self-attention in different parts of the network either yields equivalent or inferior performance. Furthermore, the utilization of linear attention improves the model's speed compared to regular self-attention.

- As the optimal model and training configuration was found for activity recognition on IMUs data, GLULA also showed to be optimal for skeleton-based data among network variation. The experiments for skeleton-based data were done on the NTU RGB+D dataset, The former consists of videos of performed activities and extracted three-dimensional skeleton data by Kinect systems. In this work, for NTU, two coordinates out of three were used since most of the benchmarked learned methods [44] can only extract two dimensions from the video. Moreover, it does not create complex systems, which is avoided in this work to be consistent with the idea of raw data inputs. The proposed model on the NTU dataset in two-dimensional format showed relative results with recent models in the literature. Moreover, it showed comparative results in the health-related subset of the NTU dataset achieving 89% accuracy. Thus, it shows that both types of inputs can be correlated and the unified solution is possible, where the resulting model is effective in terms of size and computation complexity.

- While the experimental results of the network show moderately inferior scores to recent models focused only on skeleton data, as was underlined in the experiments' discussion, it is possible to enhance classification ability even further with more pre-training on different skeleton sets. So, it makes usage of unified solution GLULA in this type of domain a compelling option. But the most vital advantage over networks for skeleton-only data is much higher processing speed due to structural optimization and overall simplicity, such as taking raw inputs.

The remainder of the paper is structured as follows. Chapter 2 covers the background: sensors' data types for HAR and how they are collected, human activity classification, and recent developments in the field. In Chapter 3, an overview of related works in the field is provided. The problem formulation and the proposed approach are outlined in Chapter 4. Chapter 5 showcases the experimental results

and conducts comparisons with other methods. In Chapter 6, conclusions are drawn based on the thesis findings in thesis and discuss potential avenues for future research.

Chapter 2

Background

2.1 Sensors' output types

Human Activity Recognition involves identifying human activities using various sensor data types. IMUs output measures acceleration and orientation from body-worn sensors. Video data captures body movements and interactions in RGB format. Skeleton data represents joint positions and can be collected using Kinect-like systems or processed and extracted from video using machine learning techniques. As it was mentioned, this work focuses on skeleton data and IMUs sensors' data.

2.1.1 Body-Worn Sensors

Body-worn sensors are popular for human activity recognition tasks due to their non-intrusive nature and ability to capture human movements continuously. Most of these sensors come in the form of inertial measurement units and can be found in devices like smartwatches and smartphones, typically including accelerometers, gyroscopes, and sometimes additional sensors like heart rate monitors [3]. They provide valuable quantified information about the wearer's movements, orientation, and physical activity patterns in two to three-dimensional space [9]. Datasets of activities with body-worn sensors are collected by recruiting participants to wear IMUs on various joints of the human body while performing specific activities, recording data, and

labeling it for supervised machine learning algorithms. The placement of IMUs can vary from one dataset to another. For example, in the SKODA dataset [32], the subject was performing 10 manipulative gestures, while mounting 20 acceleration sensors as can be seen in the figure 2-1.

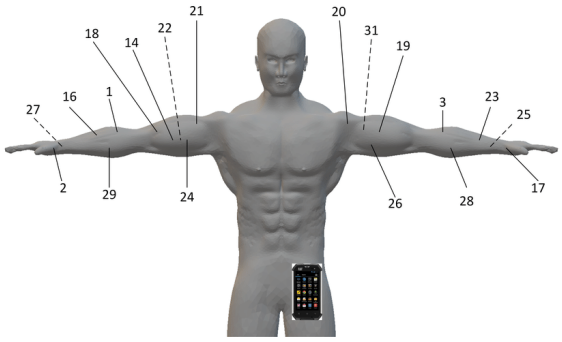

Figure 2-1: The placement of sensors on the subject in SKODA dataset [33]

2.1.2 Human Skeleton Data

Skeleton data, on the other hand, represents human poses or joint locations in a 2D to 3D space. Despite necessitating supplemental processing for frame extraction, which could be considered a drawback, skeletal data exhibits superiority in HAR over video RGB frames due to its enhanced robustness against noise. For example factors such as lighting conditions, background interference, and perspective variations demonstrate less impact on skeleton data in action recognition [6]. There are several ways how to gather skeletal information from frame sequences for action recognition.

The most common way is to collect human skeleton data for HAR using depth sensors, such as RGBD cameras (e.g., Microsoft Kinect), that can estimate the two or three-dimensional positions of human body joints, capturing the spatial configuration of the body during various movements. Then, joint information is extracted and can be represented as dots or connected skeletal graphs in 2D or 3D planes. During the gathering, participants are recorded performing the activities, and the depth sensor tracks and records the positions of their body joints, creating the skeleton data used for a certain particular activity. For example, the NTU RGB+D dataset was

collected using three Kinect V2 cameras with different horizontal imaging viewpoints in a controlled indoor environment. The dataset includes 56,880 video samples of 60 different human activities performed by 40 subjects. Authors were able to extract skeleton data in a joint position format using the data from Kinects [29].

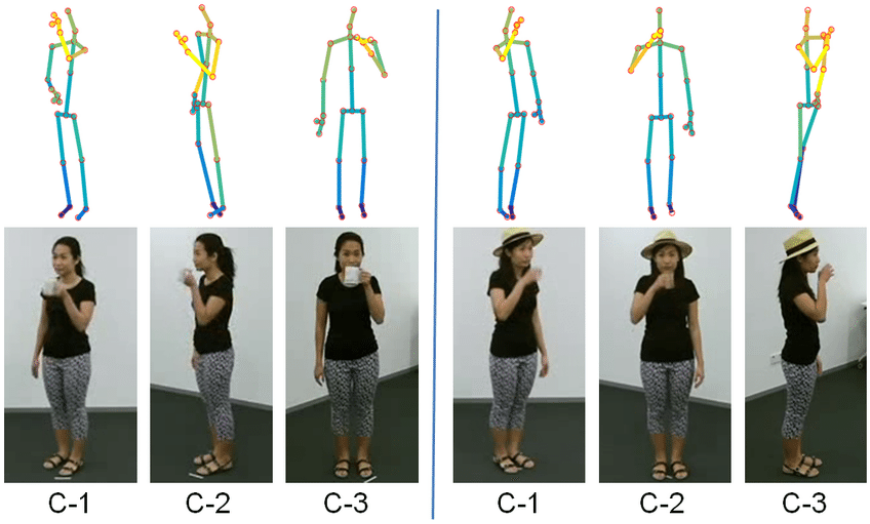

Figure 2-2: An example from NTU dataset; skeletons with connected joints are shown at the top [29]

Another way to extract skeleton data besides sensors is through pose estimation. However, it requires additional computations since it is complicating tasks and current algorithms that have adequate results are based on deep learning methods. Most of the models in pose estimation are pure CNNs [44], where convolution as an operation for inference was optimized substantially [15]. Although, vision transformers currently have SOTA results in image pose estimation with simple and flexible architecture [44].

2.2 Overview of HAR on RGB Videos

Human Activity Recognition on video involves the use of computer vision and deep learning techniques to automatically identify and classify human actions performed in video sequences. Traditional HAR methods often relied on additional sensors like depth estimator units. Nevertheless, recent advancements in deep learning have enabled the use of video classification models to predict actions directly from video

frames. One of the solutions was the introduction of 3D CNNs, where the first and the second dimension represented pixels in the frame, and the third represented point in time, or an exact frame [11]. Another method was to fuse the CNN model and RNN-type models such as LSTM, where CNN extracts features from the frame and then feeds it into the recurrent model to analyze it in the temporal domain before giving the prediction. Various deep learning models were tried in the fusion such as ResNet (pure CNN model) and Vision Transformer (ViT - self-attention based solution for images), which were utilized to recognize human activity in videos. These models have achieved remarkable accuracy levels; for instance, the ViT-based solution model managed to get SOTA accuracy on the HMDB51 dataset [41]. In this context, the ViT-LSTM hybrid model has demonstrated superior performance compared to the 3D ResNet and ResNet-LSTM in HAR tasks [34].

2.3 Human Activity Recognition

in this work, it is decided to focus on wearable sensors' output data and extracted human joints in the form of a two-dimensional skeleton. First of all, 2D was chosen over 3D since it is more practical due to most of the pose estimation models work in this space, which will be beneficial in further research. Another point is that 3D pose estimation involves multiple steps, such as human body tracking, joint localization and transformation to 3D [16]. This adds another complication to the process of human activity recognition and gets out of scope in this work.

Despite the differences in their visual representations and collection methods, both body-worn sensors and skeleton data can be categorized together due to their spatial nature. Both types of data capture human movements and activities, but they differ in exactness. Body-worn sensors capture changes in location, orientation, and motion patterns, while skeleton data provides exact joint locations, sometimes enabling a more detailed analysis of human poses and movements. Since both of the outputs sit in the temporal dimension, it can be assumed that, with knowing the priors, one data type can be transformed into another.

Thus, the hypothesis is that both of the data types can be solved by a unified solution. Consequently, in further research, solutions, where both wearable sensors output data and extracted human joints are merged as one input, can be examined. For example, the withstandability to losing part of the data like the subject is in occlusion in video, but sensor data is still present.

2.4 Recent Developments in the field

Human activity recognition on both IMUs sensor data and skeletal data representing human joints had a similar trend in the research area. Initially, traditional machine learning techniques were utilized. In recent years, substantial advances have been made facilitated by deep learning research. The introduction of Convolutional Neural Networks and Recurrent Neural Networks marked a significant shift in both types of data, providing the ability to recognize spatial and temporal features, respectively [47] [7]. Nevertheless, these architectures have certain limitations, like the vanishing gradient problem in RNNs and the inability to sequence context capture in CNNs.

Thus, the hybrid approach for both IMUs sensors and skeletal HAR came on top and it is attributed to its foundational structure, the Long Short-Term Memory networks. LSTMs were capable of addressing the issue of the vanishing gradient, a limitation notably present in preceding recurrent units. The compound solution was able to withstand the inherent flaw of CNNs, namely their incapability to encode spatial information, this drawback was mitigated by introducing CNNs' features in the temporal domain for LSTM modules [47] [45].

The introduction of transformer models, originally introduced by [38] for natural language processing, proved to be adaptable in human activity recognition tasks. With their self-attention mechanisms, transformers overcome the limitations of previous models, enabling effective recognition of long-term dependencies and intricate spatial-temporal relationships in human activity data. The Transformer model uses positional encoding to maintain a sense of order in the input data and consists of multiple identical layers. Each of these employs a self-attention mechanism to weigh

the relative importance of input tokens in a global context, and a feed-forward neural network to process this information further and store learned "knowledge" like lookup tables.

Moreover, it was found to be highly parallelizable compared to recurrent LSTMs. Currently, most of the SOTA results both in IMUs sensors and skeletal datasets are achieved by transformer-based models [20] [50].

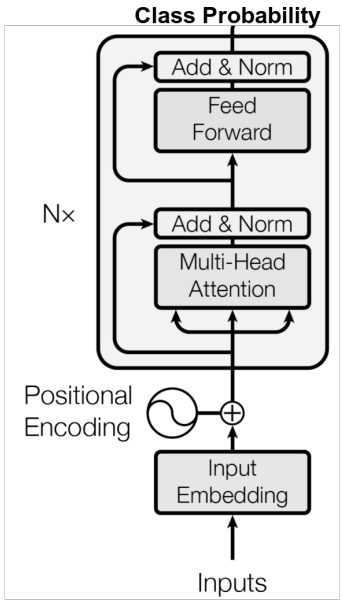

Figure 2-3: Regular Transformer architecture for classification [38]

Chapter 3

Related Works

3.1 Skeleton-based Action Recognition

Initially, Recurrent Neural Networks [5] were the dominant tool in the sphere of skeleton-based human action recognition, due to their innate ability to deal with sequential data. In parallel, the usage of Convolutional Neural Networks was explored as an alternative approach [18].

As it was mentioned, each of the solutions has its own drawbacks, which can be partially solved by the introduction of the hybrid solution: use CNNs as a feature extractor from the particular frame and then feed new representations to the LSTM [45]. Concurrently, the hybrid approach leverages the local feature extraction prowess of CNNs. Due to being a recurrent-based model, LSTMs work in the temporal domain and thus can extract useful information from a sequence of frame representations for the classification. For this reason, the hybrid solution reacher higher results than pure CNN and RNN networks [45].

Despite their popularity, these strategies overlooked the spatial interconnections among skeletal joints, which led to the progress of Graph Convolutional Networks (GCNs). By efficiently mapping these spatial configurations onto graphs, GCNs got comprehensive results [46]. At first, limitations in the form of GCN's reliance on a predetermined topology led to the introduction of a learnable topology used for action recognition. This made subsequent models that used this strategy more robust and

eventually boosted the performance [48].

As in different tasks, Transformer-based solutions were also employed in skeleton-based action recognition. However, the challenge in integrating these solutions is the need to navigate the additional temporal dimension. One strategy is to develop a dual-stream model, employing spatial and temporal Self-Attention for the task of modeling correlations both within and between frames [24]. Despite these attempts, the Transformer-based solutions were still not among SOTA methodologies. One of the reasons could be the strict adherence of these approaches to the traditional design of vanilla Transformers ignoring special characteristics of skeleton data. To mitigate it, in [50], authors proposed a self-attention mechanism on a hypergraph. This was integrated into the model named Hyperformer and helped it with incorporating intrinsic higher-order relations. They achieved state-of-the-art results while possessing the best efficiency among recent SOTA works.

3.2 Body-Worn Sensors Action Recognition

A significant number of research proposals on body-worn sensors' HAR tasks have been implemented and tested using classical machine learning algorithms and more recently advanced deep learning models. For example, the performance of classical machine learning algorithms has been analyzed on a benchmark PAMAP2 dataset in [25]. A. Reiss [25] implemented the decision tree, kNN, and Boosted classifiers, of which kNN performed best on the PAMAP2 dataset. Boosted classifiers performed very close to kNN.

The problem with classical machine learning algorithms is that they rely on features, which are handcrafted like variance, mean, and standard deviation. Thus, they should be selected from a pool limited by human knowledge and suitability for the task. The main advantage of deep learning approaches over classical machine learning is automatic feature selection, which has also been tried on HAR. Various architectures such as CNN, and LSTM have been used, and they show better results than classical machine learning algorithms [19]. As in skeleton-based action recognition,

hybrid solutions such as DeepSense [47] were utilized and showed better performance than CNNs and RNNs separately.

In [19], the authors introduced the multimodal model AttnSense, which has a similar structure to the DeepSense in [47] but with the addition of an attention mechanism. The AttnSense model combines distinct convolutional layers on the dimension of the sensors, the attention mechanism, and Gated Recurrent Units (GRU). This model made it possible to capture the spatiotemporal dependencies of the signals, and the attention mechanisms contributed to improving the performance score of the DeepSense [19].

However, AttnSense still relies on a recurrent mechanism, where input should be processed sequentially. Therefore, AttnSense inherits the problem of a lack of parallelization. To address this problem, Mahmud et al. [20] proposed the non-recurrent self-attention-based model. They utilized the self-attention mechanism in their work, and it was able to capture spatiotemporal dependencies of sensor data as before, but without any recurrent mechanism in their model.

As networks become more computationally intensive, memory usage has become the interest of HAR research. Tang et al. [35] proposed a memory-efficient approach to HAR tasks, which addresses the problem of memory usage. They presented the CNN model with redesigned Lego filters. The model managed to preserve the comparable performance with considerably fewer parameters. Their work was the first that proposed a lightweight CNN for sensors-based human activity recognition.

3.3 Transformer-based architectures

Following the ideas above, non-recurrence is a desired property of the model because it allows parallelization. Furthermore, the attention mechanism boosts the networks by giving them the ability to learn the importance of each input dimension.

Another point is that transformer and RNN perform better in HAR tasks than conventional approaches, so research in the NLP domain may lead to better solutions that do not use standard architectural structures. One particularly timely net-

work was Evolved Transformer (EV) [31]. To find it, the authors of [31] used an evolution-based neural architecture search, and the network discovered demonstrated a noticeable improvement in NLP tasks.

The main interest point for HAR from the EV model was its structure, where the first local information is extracted and added to the data and then fed to the global layers. During the inference, model parts responsible for local representation expand the information and do not just extracts it because it preserve the dimensionality of the data. This can be advantageous for global parts and therefore for the classification. Since it is possible, authors discard feed-forward (FF) layers from EV saving memory by decreasing the total number of parameters.

One of the drawbacks of transformer-based solutions is quadratic complexity. To solve this, linear attention in [13] was introduced. By utilizing approximation to self-attention, authors were able to get linear time and space complexity with respect to the length of the sequence, while full softmax self-attention gives quadratic which affects the whole transformer. If linear attention is utilized properly in HAR tasks, it can help with the rational use of the resources available in mobile and embedded systems, where it is critical due to the limitation of their capacities.

Chapter 4

Methodology

4.1 Problem Definition

4.1.1 Problem Definition for Body-Worn Sensors' Data

To provide context for the proposed architecture, it is essential to outline the problem at hand. Most IMU-based HAR datasets exhibit a consistent structure, characterized by C columns representing different sensor outputs. For instance, three columns may represent 3D acceleration data captured by an inertial measurement unit. Moreover, the IMU often includes additional sensors such as a magnetometer and an accelerometer. In IMU HAR datasets, these sensor outputs are organized into R rows, with each row representing an instance of sensor readings over time. Consequently, matrix $\mathbf{T}$ is obtained with dimensions R x C, encapsulating the time-series data from the sensors.

Then, $\mathbf{T} = \begin{bmatrix} T_1, & ..., & T_c, & ..., & T_C \end{bmatrix}$, where $T_c = \begin{bmatrix} T_{c_1}, & ..., & T_{c_r}, & ..., & T_{c_R} \end{bmatrix}^T$.

In this scenario, the atomic value of the sensor output at position (column) $\mathbf{c}$ and time instance (row) $\mathbf{r}$ is denoted as T_{c_r}. The objective of the human activity recognition (HAR) task is to classify a label based on the given matrix $\mathbf{T}$. The label can represent either a specific action or a sequence of actions.

4.1.2 Problem Definition for Human Skeleton Data

Human skeleton-based activity recognition datasets for the most part have a common structure among themselves as well. At some time point $\mathbf{r}$, the structure can be characterized by $\mathbf{J}$ joints and each joint has $\mathbf{D}$ dimensions. However, it is possible to change the representation to one-dimensionality, where there would be $C = \mathbf{J} * \mathbf{D}$ columns. Now, the first $\mathbf{D}$ columns would represent the first joint. Also, frames from one video in skeleton datasets are organized into R rows, with each row representing an instance of extracted joints information over time. Consequently, matrix $\mathbf{T}$ can be constructed with dimensions R x C, which is the same as in IMU-based HAR datasets. By representing them similarly, the objective for both data types can be unified, where the task is to assign an activity label to the given matrix $\mathbf{T}$.

4.2 Proposed Approach

The objective of this study is to develop an efficient non-recurrent model for both skeleton-based and IMU-based human activity recognition (HAR) that achieves effective parallelization, considers the importance of each sensor output, and ensures high performance. Additionally, it is aim to address the complexities associated with memory usage and inference time.

While Transformer networks have shown superior performance in classification tasks, they suffer from high time and space complexity. The Evolved Transformer (EV) model, discovered through evolutionary search, addresses some of these issues by reducing parameters and avoiding feed-forward constructions. However, EV models were primarily designed for conversational NLP or translation tasks and still rely on softmax self-attention, which leads to high space and time complexity during inference.

To create a more efficient and suitable network for (unified) HAR, inspiration from EV models is drawn and significant modifications are introduced. The approach incorporates the concepts of local-global aggregation and addresses the problem of quadratic space and time complexity by utilizing linearized attention.

The novel HAR model begins by prepending learnable tokens and employs a gated convolutional network with a branching structure of wide convolutional layers. It also incorporates a linear attention network, which has linear complexity, to extract features for the classification layer. The classification layer consists of two fully-connected (FC) layers that utilize the processed learnable class tokens, sized to one timestep.

However, like most transformer-based models, the approach requires a substantial amount of training data. Since many HAR datasets are limited in size, it is possible to employ manifold mixup regularization and other training techniques to enhance the network's accuracy and generalizability in data-deficient settings.

The overall structure of the network is illustrated in Fig. 4-1, with each component labeled accordingly. In the model, the input data is first segmented using a sliding window and then normalized. The normalization technique varies for each dataset. Next, each timestep of the normalized input data is mapped to a constant dimensionality R^E through a trainable projection or embedding. Learnable class token is appended to the start of the input data, allowing it to extract relevant information from all timesteps and channels through the self-attention block. The linearized version of softmax self-attention achieves effective results and learns to abstract and attend to the information, making the prepended learnable vector a valuable feature for classification.

To incorporate positional information and enhance the model's performance on time-dependent tasks, axial positional embedding is utilized, which is learnable and added through augmentation. This approach factorizes the encoding matrix into two matrices, reducing the number of parameters and optimizing memory usage.

Throughout the model, normalization layers are applied before each block to ensure distribution stability. The main block consists of either the gated convolutional network or the attention network, both preceded by the normalization layer. In experiments, the gated convolutional network demonstrates better generalization performance compared to softmax self-attention and linear attention. Additionally, skip connections are used to facilitate gradient flow, partially addressing the vanishing

gradient problem and promoting faster convergence.

Next, the data is passed through two separate branches of convolutions, as shown in (6) in Fig.4-1, inspired by similar structures found in the Evolved Transformer. Various activation functions, such as ReLU and GELU, were evaluated, but the Mish activation function demonstrated the best performance in transformers, as depicted in (7) in Fig.4-1. Following the branching structure of EV models, a depthwise separable convolution is applied to extract spatiotemporal local features, as indicated by (8) in Fig. 4-1.

After this module, a skip connection is performed, and the data is then passed through an additional block, denoted as (9) in Fig. 4-1, which significantly enhances the results by providing a more complex representation of temporal data. This block can be implemented using attention networks or GCNs to generate the final output data. In this study, the linear attention network outperformed GCN and softmax self-attention in the additional block. The linear attention network offers linear complexity with respect to the input length, while the complexity of softmax self-attention is quadratic.

Moreover, both softmax self-attention and linear attention networks have the same number of parameters, with GCN utilizing slightly fewer. However, during inference or training, the softmax self-attention network incurs a higher memory footprint and computational usage compared to the linear attention network.

The assumption is that the gated convolutional network, serving as the main block, and the branched convolutions at the beginning of the model help capture local features and introduce new locally found information to the data. Subsequently, the linear attention network, as an additional block, extracts global features from the locally transformed data. Self-attention considers all data simultaneously, which may explain the superior performance of GCN in the main block and the utilization of attention in the additional block.

Finally, a learnable token is extracted and processed, as depicted in (10) in Fig.4-1, at the beginning of the processed input matrix. This token is then fed into the classification layer, shown as (11) in Fig.4-1. The classification layer includes the

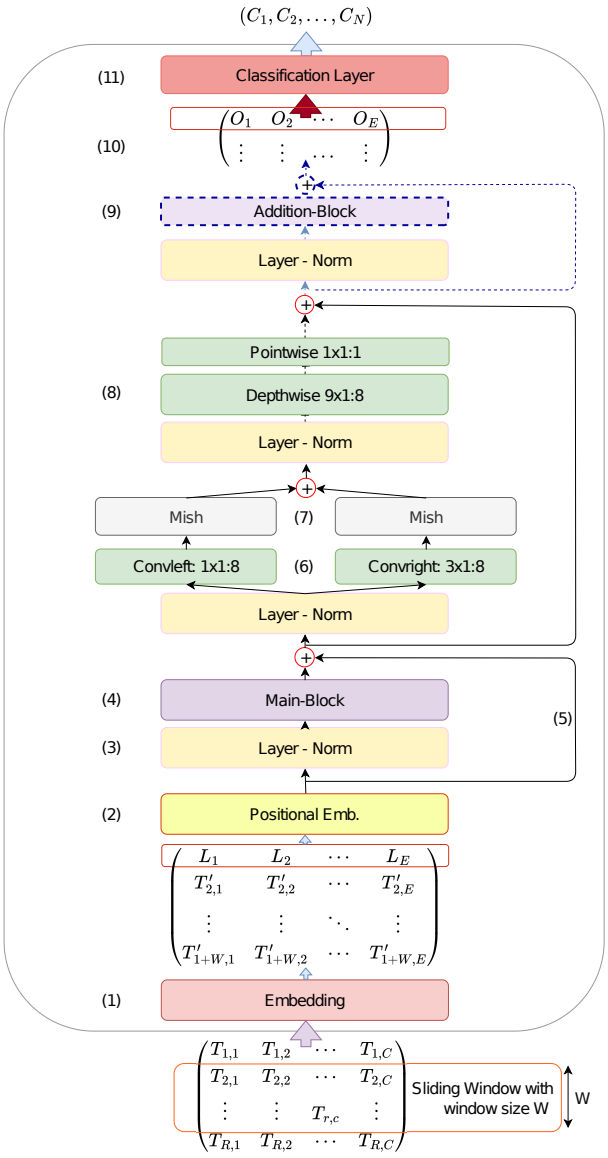

Figure 4-1: The graphical representation of the proposed model's structure. Data preprocessing and each layer are shown and numbered following the model description given in the methodology

Mish activation function and, considering concerns for memory usage, the learnable token is of size one timestep or embedding dimension. The two fully connected layers of the classification layer scale accordingly to the token's size, resulting in minimal additional parameter overhead.

Throughout the study, several models were tested, with a focus on three main variants: GLU-HAR, GLUSA-HAR, and GLULA-HAR. The "GLU" component represents the gated linear unit (or GCN), "SA" refers to self-attention, and "LA" denotes linear attention. The GLU-HAR model incorporates GCN as the additional block, while GLUSA-HAR utilizes softmax self-attention, and GLULA-HAR employs the linear attention network. In some tables, the HAR appendage is omitted for brevity.

In summary, the proposed architecture enhances the efficiency and performance of HAR models by taking into account the unique characteristics of sensor data. Through the utilization of linearized attention, axial positional embedding, and manifold mixup regularization, this work address challenges related to space complexity, time complexity, positional information, and limited training data, thereby contributing to the advancement of HAR techniques. As discussed later, GLULA-HAR achieved the highest performance among the three variants, outperforming other models and demonstrating comparable or superior results compared to state-of-the-art approaches. Further details on these models and their outcomes can be found in 5 and 5.5.

4.3 Gated Convolutional Network

In line with the aforementioned information, it is worth noting that RNN suffers from a lack of parallelization in input processing, resulting in slower training and inference times. On the other hand, CNN networks can perform computations simultaneously, making them faster than RNN-based solutions. To create more efficient language models, Dauphin et al. (2017) introduced a gated convolutional network (GCN) that utilizes parallelizable causal convolutions.

Let's begin by clarifying what causal convolutions entail. Causal convolutions

are similar to regular convolutions, but the input is left-padded with zeros by $\mathbf{k}-\mathbf{1}$, where $\mathbf{k}$ represents the kernel size of the causal convolution block. This approach ensures that the GCN only considers previous and current timesteps, avoiding any influence from future inputs.

The input $\mathbf{X}$ would be fed into two different causal convolutional blocks with filters $\mathbf{W}$ and $\mathbf{V}$ respectively, where $\mathbf{W}, \mathbf{V} \in \mathbf{R}^{k \times C \times C}$ and C is the dimension size of sensors' signal. Then, two separate outputs will be put into the gated block, which uses a mechanism of gating linear unit [4]. This mechanism puts one of the outputs through the activation function and then gates the other by element-wise multiplication. See formula 4.1, where $\mathbf{b}$ and $\mathbf{c}$ are learnable bias parameters; ϕ is an activation function and f_{gcn} is a function interpretation of a simple GCN.

$$f_{gcn}(\mathbf{X}) = (\mathbf{X} * \mathbf{W} + \mathbf{b}) \otimes \phi(\mathbf{X} * \mathbf{V} + \mathbf{c}) \tag{4.1}$$

In concept, the gating mechanism can perform the selection of valuable features by $\mathbf{V}$ that control what information from other output (which was convolved by $\mathbf{W}$) will be passed to the subsequent layers. By this, GCN learns to move only relevant information and gain non-linearity. Furthermore, as reported by the original paper [4], the residual skip was added to the GCN for reducing the vanishing gradient problem, and GCN was with bottleneck structure within a layer for reduction of computational cost. For the activation function, ϕ, the Mish function was used as it showed optimal results in a variety of tasks [21].

4.4 Self-Attention

The self-attention mechanism can be viewed as a means of computing the significance or importance of each timestep in a sequence. In this scenario, the self-attention function establishes relationships between each timestep and all other timesteps within the input. By assessing the similarities and correlations between timesteps, self-attention calculates new values for each timestep. These values reconstruct the timesteps and

incorporate information from other timesteps based on their relevance.

As detailed by Vaswani et al. (2017), the query, key, and value (Q, K, V) can be formed by linearly transforming the source sequence using three distinct learned weight matrices W^q, W^k, and W^v. The query can be envisioned as the information being sought, while the key represents the data that is pertinent to this information. The value, on the other hand, is a learned representation of the content within the input.

In this case, architecture represent the query and the key as separate transformed timesteps, which are subsequently compared. This comparison is accomplished through a scaled dot-product, following the approach outlined in [38]. The output of this comparison represents the similarities between the query and key, or in other words, the attention scores among different timesteps. Next, the output is normalized using softmax and multiplied by the value vector, resulting in values that encode the relative importance of each timestep and incorporate relevant information from other timesteps.

$$f_{at}^{(h_i)}(Q, K, V) = V' = softmax(\frac{Q, K^T}{\sqrt{d_k}})\, V \tag{4.2}$$

where

$$\frac{Q, K^T}{\sqrt{d_k}} \tag{4.3}$$

is the compatibility function in the form of a scaled dot-product, d_k is the dimension of the key used for scaling the function to improve numerical stability, h_i represents head i. After computing attention on Q, K, V, an output matrix is constructed, which can then further be utilized as an attention head. We can compute more heads and by this, we can make use of multi-head attention. Distinct heads can capture different unique features by having individual parameters. Then, to get the final result, the output from different heads is concatenated, and by applying learned linear projection $\mathbf{W_{out}}$, the concatenated outputs are transformed into the original dimension.

$$f_{output} = \mathbf{W_{out}} \times concat(f_{at}^{(h_1)}, ..., f_{at}^{(h_i)}, ..., f_{at}^{(h_I)}) \tag{4.4}$$

4.5 Linear Attention

In the softmax self-attention mechanism, once the query, key, and value (Q, K, V) are obtained through linear transformations, all three are input into Equation 4.2. In this research work, the dimensions of Q, K, V are $R^{N\times D}$, where N represents the sequence length of the input, and D denotes its dimensionality. Examining Equation 4.2, we can observe that softmax attention scales quadratically with respect to N, resulting in a computational complexity of $O(N^2D)$ [13]. This quadratic scaling is also applicable to memory consumption since the full attention matrix $N \times N$ needs to be stored for gradient computation.

To alleviate the time and space complexity, it is essential to consider softmax self-attention as a more generalized form of self-attention, where the similarity function is an exponentiated dot product between the query and key (Q, K). By introducing a unique value vector V', the generalized self-attention can be expressed as follows:

$$V_i^{'} = \frac{\sum_{j=1}^{N} sim(Q_i, K_j)V_j}{\sum_{j=1}^{N} sim(Q_i, K_j)} \tag{4.5}$$

where, in the softmax self-attention case, the similarity function $sim(Q, K) = exp(\frac{Q^T K}{\sqrt{d_k}})$ as it was stated above. To note, in the self-attention different similarity function can be used such as polynomial attention [37].

For the equation 4.5 to be an attention function, a constraint must be followed for a $sim(\cdot)$ function: to be a non-negative function [13]. This actually includes all kernels of type $ker(q, k) : R^{2\times D} \rightarrow R_+$. Then given such a kernel $ker(q, k)$ with a feature mapping ϕ, we can define $sim(\cdot)$ the function as the corresponding kernel $ker(q, k) = \phi(q)^T\phi(k)$. Then the whole rewritten equation 4.5 goes as follows,

$$V_i^{'} = \frac{\sum_{j=1}^{N} \phi(Q_i)^T\phi(K_j)V_j}{\sum_{j=1}^{N} \phi(Q_i)^T\phi(K_j)} \tag{4.6}$$

then, using the associative property of the matrix product, the equation goes further:

$$V_i^{'} = \frac{\phi(Q_i)^T \sum_{j=1}^{N} \phi(K_j)V_j^T}{\phi(Q_i)^T \sum_{j=1}^{N} \phi(K_j)}. \tag{4.7}$$

Equation 4.7 is called linear attention. It has linear complexity in time and memory with respect to the sequence length N, due to $\sum_{j=1}^{N} \phi(K_j)V_j^T$ and $\sum_{j=1}^{N} \phi(K_j)$ can be computed once for each query [13].

For linear attention, feature maps of a certain dimensionality K are first computed, which in turn give us the complexity $O(NKD)$ for linear attention in terms of mathematical operations. However, Katharopoulos used an exponential linear unit as the feature map [13]. The full equation looks like this: $\phi(x) = elu(x) + 1$. In linear attention, this feature map resulted in a complexity of $O(ND^2)$ in terms of mathematical operations. It is very efficient if N is considerably greater than D. As the authors have shown in their work, linear attention achieved results comparable to regular softmax self-attention while being faster on the task where the length of the sequence is higher than the dimensionality of the data [13].

4.6 Training Techniques

The development of Manifold Mixup regularization was driven by the need to address the issue of overconfident predictions made by neural networks trained on hard labels [39]. Overconfidence can be problematic as it may result in incorrect classifications when evaluated on slightly different samples, which can include outliers, noise, or distribution shifts.

To mitigate these effects, manifold mixup regularizers aim to encourage deep learning models to produce less confident predictions during training by leveraging the interpolation of hidden features as an additional training signal.

Algorithm 1 Manifold Mixup regularizer [39]

Input: Deep neural network f with set of layers S and parameters θ, constant α, input minibatches.

1: Random layer k from S is selected.

2: Minibatches (x, y) and (x', y') is fed through the layers of the network f until layer k. The resulting hidden representations are $(g_k(x), y)$ and $(g_k(x'), y')$. Minibatches could be two distinct or the same reshuffled batch.

3: Input Mixup Mix_λ is performed on intermediate hidden representations (($g_k(x)$, ($g_k(x')$) and one-hot labels (y, y'). The result is the mixed minibatch: $\bar{g_k}, \bar{y} := Mix_\lambda(g_k(x), g_k(x')), Mix_\lambda(y, y')$, where $Mix_\lambda(q, u) = \lambda \cdot q + (1 - \lambda) \cdot u$ and $\lambda \sim Beta(\alpha, \alpha)$.

4: The forward pass is continued in the model from where the networj was stopped at the layer k until the output using newly acquired mixed minibatch.

5: The output and mixed labels $\bar{y}$ are fed into the loss function, and the calculated value is used to update all of the parameters θ in f. During updating, backpropagation goes through the whole computational graph.

Output: Neural network f with updated θ. =0

In order to represent this regularization technique, we first define a deep neural network as $f(x) = f_k(g_k(x))$, where g_k represents a component of the network that maps the input x to the hidden representation at a specific layer k. The function f_k encompasses the remaining parts of the model that lead to the output $f(x)$ based on the extracted features $g_k(x)$. To incorporate Manifold Mixup as a regularization approach for training such a network, five steps need to followed.

By employing Manifold Mixup during training, the resulting network exhibits smoother decision boundaries across various levels of representation, as noted in Verma et al.'s work [39]. Moreover, the network learns flattened class representations with reduced variance directions. These effects ultimately enhance the model's generalization capabilities, leading to improved performance not only on test data but also in the face of adversarial attacks [39].

In this work, Manifold Mixup also considered as a valuable data augmentation technique, particularly in scenarios where datasets are limited, such as in the case of HAR datasets. The rationale behind utilizing Manifold Mixup is that it generates new mixed samples at each step, which differ not only due to changes in mini-batches but also as a result of shuffling and mixing at various layers. Hence, this regularization method was applied and evaluated in the work.

Another challenge associated with limited datasets is the issue of overfitting and divergence. Overfitting arises from training on a small sample space, but it was partially addressed by incorporating Manifold Mixup. However, to further mitigate divergence between re-initialized networks, scheduling techniques can be employed. These techniques promote more stable training and weight updates. In this study, the one-cycle policy proposed by Smith et al. was utilized [30]. This policy involves gradually increasing the learning rate to a maximum value and then annealing it close to zero. The result is that it helps the model navigate steep points of the loss landscape and settle into flatter minima, enhancing stability during training.

4.6.1 Additional Pre-Training for Skeleton-based HAR

Given the availability of additional skeleton datasets and the nature of this data, it is relatively easy to transform representations from one dataset to another. The primary difference lies in the number of joints, but the datasets retain identical spatial coordinates. This conversion is significantly more challenging with body-worn sensor data due to the high variability in sensor specifics, wearing locations, and sensitivity. Therefore, it is feasible to pre-train only skeleton-based activity recognition models on a larger dataset. Several approaches can be employed to do so with the proposed network, but the decision was made to test learning hidden representations from a teacher model.

Although contrastive learning is a possibility, it demands substantial computational resources and may lead to inconsistent results due to its high sensitivity. This approach could constitute a separate research project in itself [40]. Distillation learning [8], based on direct classification probabilities from the teacher model, is also plau-

sible. However, there may be instances where the output from the teacher model's classification head breaks with unseen data, making distillation more difficult. Consequently, distillation on hidden representations was selected as the target.

As previously mentioned, this study employs 2D joint data for training the model. This decision is based on the opportunity to establish a direct connection with powerful pose estimation models, which yield results in 2D coordinates. Nevertheless, it is entirely plausible to use a teacher model that employs 3D input, potentially enabling the transfer of knowledge about the skeleton in a higher dimension. The mean absolute error was utilized as a loss function for the distillation.

For the teacher model, the Hyperformer network was employed, as its feature collection closely aligns with the outputs of the proposed architecture before classification [50]. This teacher model is further described in 3.1, where it achieved state-of-the-art results in 3D skeleton-based action classification.

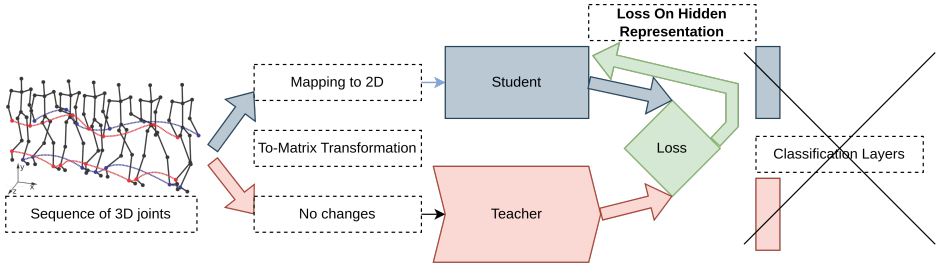

Figure 4-2: Distillation learning on hidden representations

Chapter 5

Experiments and Results

5.1 Setup and Evaluation

The proposed solution was implemented using the PyTorch library and trained and tested on a cloud-based GPU. The network was randomly initialized, and a batch size of 64 was utilized. To ensure robustness, each experiment for body-worn sensors' data was repeated five times with different seeds, and the averaged values from these experiments were used in the tables for analysis.

For human skeleton data, experiments were conducted only once. This decision was made since seed randomness did not significantly influence the results of the experiment due to the size of the skeleton dataset and the resulting stability. Moreover, as the training sessions were lengthy, it would have been computationally costly to perform five iterations of the experiment.

For evaluating and comparing the models' performance on IMUs data, the weighted F1-score as the measurement metric was employed. The weighted F1-score takes into account the label imbalance of HAR datasets and is independent of the class distribution [35]. The difference with the macro F1-score is that weighted F1-score assigns a weight to each class, which corresponds to the class's sample proportion in the total dataset:

$$F_w = \sum_{c=1}^{C} \frac{N_c}{N_{total}} \frac{2 * Precision_c * Recall_c}{Precision_c + Recall_c} \tag{5.1}$$

To evaluate skeleton-based activity recognition, the accuracy (acc) score was employed, as has been done in various studies [50]. This approach was appropriate given that the dataset used was perfectly balanced.

$$\text{Accuracy} = \frac{N_{CorrectPredictions}}{N_{Total}} \tag{5.2}$$

5.2 Datasets

To evaluate the proposed model on body-worn sensors' data, its variations, and different training techniques, experiments utilized five IMUs' HAR datasets. They are first in the list. For evaluation of activity classification on skeleton-based data, The NTU-RGB+D dataset was used.

The first dataset used for benchmarking is PAMAP2 [25]. PAMAP2 consists of sensor data from three IMUs placed on the chest, the dominant leg's ankle, and the wrist of the dominant arm. Each IMU includes an accelerometer, gyroscope, magnetometer, and temperature sensor. Heart rate measurements were also recorded separately. The sensors had a sampling frequency of 100 Hz, except for the heart rate which was sampled at 9 Hz. PAMAP2 was collected from nine participants and contains twelve different activities, with an additional six activities that were not used. Following the principle of leaving-one-subject-out (LOSO) as done in previous works [36], the data from participant number 106 is benchmark test set, while the rest of the dataset was used for training.

The second dataset is SKODA [32], which focuses on describing activities of workers in a car manufacturing environment. The dataset includes several accelerometers worn by a worker, with a sampling frequency of 98 Hz. Data was recorded from a single subject for all cases. The dataset comprises ten different activities performed during the manufacturing process, along with a null division representing no activity. In total, there are eleven classes. For training, 90% of each class was used, while the remaining 10% was reserved for testing.

The third dataset is OPPORTUNITY [26], which contains data from body-worn

and ambient sensors, with each timestep annotated with a specific activity. Activities are annotated at three levels: high-level, mid-level, or gestures, and low-level or modes of locomotion. For the experiment, focus was only on the mid-level activities, while other activities were labeled as null. This resulted in a total of 18 different activities, with a significant class imbalance where around 75% of the dataset consists of the null class. The dataset includes one drill session and five daily activity (ADL) sessions performed by the subjects. The sensors were sampled at a frequency of 30 Hz. Following previous research, ADL4 and ADL5 from subjects 2 and 3 are for testing, while the rest of the dataset was used for training and validation.

The fourth dataset is USC-HAD [49], which provides sensor data from body-worn gyroscopes and accelerometers. Each sensor provides 3-axis readings, resulting in a total of six dimensions for each instance of data. The sampling rate is set at 100 Hz. The dataset consists of an equal number of male and female participants, with each subject performing 12 different activities. USC-HAD is a challenging dataset due to the diversity of activities, sensor placement, and low dimensionality, which limits the available activity information. However, the dataset size is larger compared to the other three datasets, and it is also balanced unlike OPPORTUNITY, which is heavily skewed towards the null class. Following the LOSO principle, two subjects (13 and 14) were separated for testing.

The last dataset for analyzing body-worn sensors' data is DAPHNET [1], which was collected to evaluate the ability of different machine learning methods to learn and recognize gait freeze events. This dataset has the potential for developing an assistant for Parkinson's disease patients. The data includes readings from three wearable acceleration sensors placed on the hips and legs, resulting in a total of nine channels per sample. Each sample is annotated as either a freeze or not. DAPHNET shares similar challenges with USC-HAD in terms of sensor placement and low dimensionality, which limits the available information for activity prediction. However, the dataset is imbalanced towards the no-freeze class and has only two classes to recognize. The sensors were sampled at a frequency of 64 Hz. Following previous works, subject 2 is a the benchmark test set [36].

The only dataset that was used to analyze skeleton-based data is NTU RGB+D dataset [29]. Despite using only one skeleton dataset, it first of all has different configurations, variability, and balance in terms of samples per action and it is substantial in terms of size. Thus, this dataset alone can give comprehensive insights into how a unified model would perform on skeleton data. NTU RGB+D dataset was gathered using three Kinect V2 cameras with different imaging viewpoints in a controlled environment. The dataset consists of 56880 videos of 60 different human activities performed by 40 subjects, with each video having 30 frames per second. The authors were able to extract skeleton data in a joint position format using the data from Kinects. Therefore, the dataset provides 3D skeletons, their 2D projections, and labels. Each skeleton is composed of 25 joints in 2D or 3D format. Following the principle of LOSO, cross-subject evaluation was performed, where 40 subjects were split into equal training and testing groups.

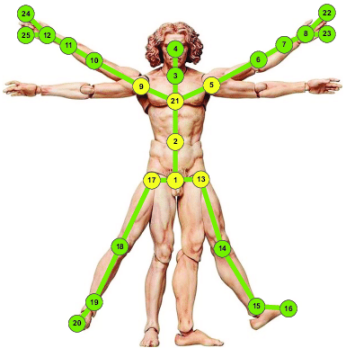

Figure 5-1: Joint Placement in the NTU dataset

A summary of each dataset, including key information and an outline, can be found in Table 5.1.

5.3 Data Preproccesing

All the presented datasets contain NaN values, which occurred during the recording process. These NaN values indicate instances where sensors were not functioning properly due to various reasons such as loss of connection, internal errors, or inability to maintain the sampling frequency.

Table 5.1: Information about presented datasets' structure

Dataset	Number of Activities	Number of Subject(s)	Testing Subject(s)
PAMAP2	12	9	106
SKODA	11	1	10% of each class
OPPORT.	18	4	2,3(Run 4,5)
USC-HAD	12	14	13, 14
DAPHNET	2	10	2
NTU RGB+D	60	40	half of all

NaN values can have a significant impact on the computations of the model and can potentially corrupt the results. Therefore, as a preprocessing step, the data were linearly interpolated to replace the NaN values with estimated values based on neighboring samples.

Furthermore, it was observed empirically that normalized data tends to exhibit more stable behavior during training and can accelerate the convergence of the model. To achieve this, all the datasets except NTU RGB+D (due to being centered already) were subjected to Z-score normalization, which standardizes the data distribution.

Next, the data were segmented using a sliding window technique with a 50% overlap. The window size for PAMAP2, DAPHNET, and OPPORTUNITY was set to the standard value of 5 seconds in real-time, as used in previous works [36]. For the SKODA dataset, the window size was set to 2.5 seconds. In the case of USC-HAD, the window size was set to 1 second, which is the standard time span utilized in related studies [20], [10]. NTU RGB+D has a window size of 64 frames. Because the dataset was already divided, only the central window part of the video sample is taken without sliding.

After segmentation, each resulting sequence was fed into the network for classification, where the model performed predictions on each segment.

5.4 Hyperparameters and Training

Firstly, various attention networks were tested as the main block in the model, but they did not yield any improvement in performance. Moreover, they consumed more memory and time compared to Graph Convolutional Networks (GCNs).

To evaluate the performance of different network configurations in the additional block layer, three types of models were constructed as described in 4. The first model, called GLU-HAR, utilized GCNs as both the main block and the additional layer. The second model, GLUSA-HAR, used a GCN as the main block and a self-attention block as the additional layer. The GLULA-HAR model shared the same structure as GLUSA-HAR but employed a linear attention network as the additional layer.

Two other models, namely GLUDynamic-HAR and GLUlightweight-HAR, were not included in the resulting tables and were not evaluated with other parameters. These models followed a similar structure to GLUSA-HAR, but differed in the additional layer, utilizing dynamic layers and lightweight convolutional networks, respectively [43]. However, the results obtained from these models were significantly inferior to the scores achieved by GLUSA-HAR, and thus they were not further tested.

In the model, the input data is embedded into a new dimensionality from the original number of channels. After resizing, it is important to ensure that the attention layer utilizes different heads to capture distinct features. To achieve this, the embedding dimension E was set to be a power of two. Specifically, we can compute it: $E = 2^{\lceil \log_2(C) \rceil}$, where C represents the number of channels. The count of model parameters differs proportionally to E for each dataset.

In the case of USC-HAD, where the channel set is limited, the model size is smaller. To enhance its learning capabilities, the embedding dimension was doubled. Despite this modification, the number of parameters for USC-HAD remained comparatively low. Moreover, in the case of NTU RGB+D, doubling was also needed in order to get close dimensionality with the Hyperformer model [50]. The requirement exists to be able to evaluate distillation learning, where Hyperformer would have a teacher role. More details can be found in 5.5.2.

A comparison between the Adam and AdaBelief optimizers was conducted in 5.5.1. Both optimizers performed well, but AdaBelief showed better performance. AdaBelief is a combination of Adam and stochastic gradient descent (SGD), providing flexibility to adapt to different problem scenarios. Additionally, AdaBelief takes into account the gradient and the curvature of the loss function. Based on the test results and the aforementioned reasons, all experiments utilized the AdaBelief optimizer during training.

As mentioned earlier, HAR datasets can be imbalanced, as in the case of OPPORTUNITY, and have a limited sample size, which is crucial for self-attention-based models. However, the opposite holds for USC-HAD, where there is an equal number of examples per class and a large number of samples. Therefore, Manifold Mixup and input balancing techniques were employed for PAMAP2, SKODA, DAPHNET, and OPPORTUNITY. While manifold mixup can bring benefits to the NTU RGB+D dataset, it was also discarded as in USC-HAD. There are two reasons: the dataset is perfectly balanced; a lot of changes to the algorithm should be done for it to work properly with distillation pre-training performed on the NTU RGB+D extended dataset.

In GLUSA-HAR and GLULA-HAR, two attention heads were utilized for PAMAP2, SKODA, OPPORTUNITY; four for NTU RGB+D. However, for USC-HAD and DAPHNET, which have a small number of channels and consequently a small embedding size, only one attention head was used. Nevertheless, using two attention heads for these datasets would not noticeably degrade performance. Increasing the embedding size to make the model wider in all dataset cases, as mentioned before, did not lead to any significant changes.

5.5 Results on Evaluation of proposed methods

Initially, training techniques were assessed on the proposed models, GLULA and GLUSA. Subsequently, three variations of the model were trained using the suggested training methods and compared across HAR datasets. Additionally, the impact of

the network structure on inference time was examined.

5.5.1 Evaluation of Training Methods

Before evaluating the proposed model and its variations with the suggested training techniques, it is important to test the performance of each training method individually.

For this purpose, the PAMAP2 was selected for representing IMU datasets as it provides a good balance among the different body-worn sensors' datasets. PAMAP2 is moderately imbalanced, but not as severely as OPPORTUNITY, and it has a sufficient sample size for training, although smaller than USC-HAD. This makes PAMAP2 a suitable benchmark for comparing the performance of the model under different training conditions and assessing how well the suggested methods meet the proposed expectations.

Another benchmark set is the NTU RGB+D skeleton dataset due to its perfect balance in terms of samples per class distribution and also because this set is of a different origin. However, there would be also a difference in evaluation: distillation training is tested only on NTU RGB+D and mixup do not work with it without significant modification.

Both GLULA and GLUSA models were used for testing the training techniques, while GLU-HAR was not considered due to its overall poor performance, as shown in 5.5.2. After the first two combinations, GLUSA was no longer used afterward on NTU RGB+D due to its notable lag behind GLULA. Another point to mention is that there is no STD calculation for NTU RGB+D due to only one run.

The distillation was performed using the extension of the NTU RGB+D dataset called NTU RGB+D 120 [17] and the Hyperformer model in the role of a teacher [50]. Then, the model was fine-tuned on NTU RGB+D classification.

As depicted in Table 5.2, manifold mixup (MM) significantly improved the performance of GLUSA. With GLULA, although the F1-weighted score remained nearly the same, the macro-score increased from 80.65 to 85.13. This indicates that MM helped the model better capture class differences, while the weighted score showed minimal

Table 5.2: F1-weighted scores with STD on PAMAP2, Accuracy score (ACC) on NTU RGB+D using training methods on GLULA and GLUSA models

Methods	Models	F1w	STD	ACC
Adam	GLULA-HAR	86.69	3.18	77.8
	GLUSA-HAR	87.15	3.91	77.3
Adam + MM	GLULA-HAR	86.04	3.48	78.1
	GLUSA-HAR	88.54	2.52	77.5
Adam + MM + Scheduling	GLULA-HAR	88.08	2.89	79.1
	GLUSA-HAR	88.12	2.38	-
AdaBelief + MM + Scheduling	GLULA-HAR	90.09	1.14	79.8
	GLUSA-HAR	89.89	2.16	-
AdaBelief + Distill. + Sch.	GLULA-HAR	-	-	82.9

change. It is worth noting that mixups can be seen not only as a regularization technique but also as an augmentation method. Such an approach can be particularly beneficial in scenarios with limited data or imbalanced class distributions. Therefore, it can be effectively employed to address the specific requirements of HAR tasks, as demonstrated in the results.

Furthermore, as observed in Table 5.2, scheduling techniques contributed to more stable training and reduced divergence across multiple observations (five in our case). This is evident from the lower standard deviation (STD) observed across the results of the experiments. With more stable training, the average score of GLULA improved to 88%, as it no longer had a lower offset than before.

When evaluating the results of using the AdaBelief optimizer, it is evident that the accuracy scores for both GLULA and GLUSA models improved significantly. The training process of the models benefited from AdaBelief's capability to consider both the curvature of the loss function and its gradient. Additionally, another advantage of incorporating the AdaBelief optimizer is its tuning capacity, which allows for further optimization. In the other datasets, the inclusion of AdaBelief resulted in nearly identical results for both GLUSA and GLULA models.

Based on the results, the accuracy score on the NTU RGB+D dataset for different training methods showed an increasing trend on GLULA. GLUSA was discarded at

the start due to notably lower performance.

Introducing manifold mixup to Adam slightly improves the accuracy, with GLULA-HAR being at 78.1%. Adding Scheduling to the mix further increases the accuracy for GLULA-HAR to 79.1%. Both scheduling and manifold mixups made training a lot more stable and fewer outliers affected the training process. A change of optimizer to AdaBelief with MM and Scheduling improves the accuracy score significantly for GLULA-HAR to 79.8%. But the biggest change was made by distillation training, it boosted the performance up to 82.9 %. My assumption is that GLULA was able to extract useful information from the 3D model for further classification.

5.5.2 Evaluation of Proposed Models on Body-Worn Sensors' Datasets

First, it can be observed from Table 5.3 that GLULA and GLUSA models have the same parameter count for each dataset. This similarity arises due to both linear attention and self-attention layers having an equal number of learning parameters. However, they differ in terms of time and space complexities during inference. Linear attention approximates the computations of softmax self-attention, resulting in linear complexity with respect to the input length but without reducing the parameter count. On the other hand, GLU-HAR has a slightly larger parameter count.

Overall, despite the similar structure, the networks' sizes are nearly identical, but they exhibit significant differences in their complexities. GLULA-HAR is faster than GLUSA-HAR, as reflected in the speed comparison shown in Table 5.4. However, the most crucial factor is performance, as indicated by the $F1 - weighted$ scores. As seen in Table 5.3, GLULA-HAR outperforms GLUSA-HAR by a small margin on the PAMAP2 dataset. Conversely, GLU-HAR performs substantially worse than the other two models, with a score of 64.05%.

Table 5.3 demonstrates that GLUSA-HAR and GLULA-HAR have almost the same score on the SKODA dataset. The results exhibit a slight fluctuation of 0.13%, with GLUSA-HAR having a slight advantage. Once again, GLU-HAR achieves the

lowest score. In both the PAMAP2 and SKODA datasets, the embedding size is the same due to the number of channels, resulting in approximately 50K parameters. The only minor difference arises from the classification and embedding layers.

For the OPPORTUNITY dataset, the segments have the same number of timesteps as PAMAP2, but the number of sensor dimensions is tripled, and the embedding size is doubled. Consequently, the models are larger in size. Similar to SKODA, GLUSA-HAR, and GLULA-HAR show minor differences in performance, with GLULA-HAR leading by 0.43% with a score of 95.93

The USC-HAD dataset is the most complex, with the lowest number of channels and a small-scale embedding dimension. Even after doubling the embedding dimension, the number remains small, around 4.0K. In this case, GLU-HAR once again performs the worst. GLULA-HAR outperforms the softmax self-attention-based model by 5.12%, achieving a $F1 - weighted$ score of 59.38%. Attempts to improve the model's performance on USC-HAD by adding more learnable tokens or increasing the embedding dimension resulted in a size increase but degraded network performance due to overfitting. Therefore, the number of parameters for USC-HAD is optimal, and the embedding cannot extract more valuable information even with a low count.

In the DAPHNET dataset, the embedding dimension is the same as in USC-HAD, which is 16. This leads to an almost identical number of parameters between the DAPHNET and USC-HAD cases, around 4.0K. However, what makes the DAPHNET task easier compared to USC-HAD is that there are only two activity classes: gait freeze and not. Similar to USC-HAD, GLU-HAR performs the worst, while GLULA-HAR outperforms the self-attention-based model by a significant margin, with a score of 94.11% for the linear-attention-based model compared to 92.38% for GLUSA-HAR on the benchmark test set.

Although softmax self-attention is a powerful mechanism, it may lead to overfitting in constrained environments. In contrast, linear attention, despite being an approximation, often provides a more general solution. This is evident in the significantly higher weighted F1 scores achieved by GLULA-HAR in the USC-HAD and DAPH-

Table 5.3: Results obtained on different Body-Worn Sensors' datasets using the proposed GLULA and its variations

Dataset	**Models**	**F1w**	**Num. of Param.**
PAMAP2	GLULA-HAR	90.09	50.2 K
	GLUSA-HAR	89.89	50.2 K
	GLU-HAR	64.05	54.4 K
SKODA	GLULA-HAR	97.63	51.4 K
	GLUSA-HAR	97.76	51.4 K
	GLU-HAR	88.77	55.6 K
OPPORT.	GLULA-HAR	95.93	196 K
	GLUSA-HAR	95.50	196 K
	GLU-HAR	87.50	213 K
USC-HAD	GLULA-HAR	59.38	4.0 K
	GLUSA-HAR	54.26	4.0 K
	GLU-HAR	38.31	4.3 K
DAPHNET	GLULA-HAR	94.11	3.8 K
	GLUSA-HAR	92.38	3.8 K
	GLU-HAR	80.69	4.1 K

Table 5.4: Speed comparison using the average forward pass time of the model with its variations on different datasets

Model	PAMAP2	SKODA	OPPORT.	USC-HAD	DAPH.
GLU	34.3	17.7	18.6	6.82	20.38
GLULA	35.2	18.1	18.8	6.84	20.43
GLUSA	42.8	19.5	19.4	7.53	22.1

NET datasets, and comparable results to the GLUSA model on other benchmarks. Therefore, in the context of HAR, linear attention proves to be more effective.

Additionally, in theory, GLULA-HAR should be faster during both forward and backward propagation compared to GLUSA-HAR. To verify this, the speed examination of the models is done, as presented in Table 5.4.

We conducted a comparison of the inference times for each network on different datasets, using a cloud-based system. It's important to note that the time values provided may vary across different machines. The measurements are given in milliseconds, with lower values indicating better performance.

As shown in Table 5.4, GLULA-HAR consistently outperforms GLUSA-HAR in terms of speed across all datasets. This advantage is more pronounced when the ratio between input dimensionality and input sequence length is lower. Notably, this effect is particularly evident in the PAMAP2, DAPHNET, and USC-HAD datasets, where the ratio is the lowest. Although GLU-HAR was the fastest among the models, its poor performance and higher number of parameters outweigh this advantage. Therefore, this variation of the model is considered the least favorable.

While the inference times of the models were relatively close in the cloud-based GPU setup if we were to reduce the allocated memory and bandwidth or increase the batch size, the speed difference between GLULA-HAR and GLUSA-HAR would become more pronounced. This indicates that in computationally limited environments, such as embedded systems, GLULA-HAR would be even faster than GLUSA-HAR.

Considering both the speed results and the performance scores from Table 5.3, we can conclude that the GLULA-HAR model is the preferred choice among the different variations of the proposed solution. Furthermore, it offers the advantage of linear complexity compared to the quadratic complexity of GLUSA-HAR.

5.6 Comparative analysis of the proposed model on Skeleton Data

As observed in section 5.5.2, the GLULA model exhibited superior speed compared to other variations. This was a foreseeable outcome, credited to the linear attention inherent in its architectural design. Therefore, the model was not retested on the NTU RGB+D dataset, as it follows similar dimensionality to IMU-based datasets.

Section 5.5.1 also illustrated the advantages of learning distillation on teachers' hidden variables. Even when the teacher was a 3D skeleton-based model [50], it was beneficial. The assumption is that with a larger pre-training dataset and a more powerful teacher, this advantage could be further amplified. However, due to computational limitations, this proposition was not tested.

Table 5.5: Results obtained on NTU RGB+D dataset using the proposed GLULA and Hyperformer

Data	Models	Accuracy %	Num. of Param.	Av. Time s.
NTU-RGB	GLULA-HAR	82.9	2.1 M	8.6e-3
	Hyperformer [50]	90.1	2.7 M	1.02

To show the validity of the results we can compare the GLULA model with a recent transformer model, the Hypergraph Transformer [50], which has demonstrated close or SOTA results on the NTU RGB+D dataset. It was also used as a teacher in the distillation training. Despite its focus on 3D joint information, the comparison was feasible since the data sources for both 2D and 3D data are the same.

As Table 5.5 illustrates, the Hypergraph Transformer model outperforms GLULA by a considerable margin in terms of accuracy. However, the GLULA model, which used a 2D representation, held an edge in terms of processing time and model simplicity, having 28% fewer parameters. For a batch size of 64, where each element contained skeleton data frames, GLULA processed one batch in 8.6e-3 seconds on average, which is significantly faster than the Hypergraph Transformer's 1.02 seconds.

To continue a more comprehensive comparison, a health-related subset of the NTU RGB dataset was selected. This subset contains nine classes: sneeze/cough, staggering, falling down, headache, chest pain, back pain, neck pain, nausea/vomiting, and self-fanning. There were two options for GLULA to work with this subset: directly predict and calculate test accuracy or fine-tune the training set and then perform the testing. Given that the models could predict actions not present in the dataset, the F1-weighted metric was added to the evaluation.

As shown in Table 5.6, although GLULA-HAR still trails the Hypergraph Transformer on the NTU HEALTH subset, the model narrow downed the gap, especially in terms of the F1-weighted score. Additional fine-tuning further decreased the difference, indicating the potential of the GLULA model in achieving competitive performance.

Overall, it is evident that there is potential for further learning and bridging the

Table 5.6: Results obtained on NTU RGB+D health subset using the proposed GLULA and Hyperformer

Data	Models	Accuracy %	F1w %
NTU-HEALTH	GLULA-HAR	81.6	86.3
	Hyperformer [50]	90.3	92.5
	GLULA-HAR + tuning	89.3	89.4

gap between specialized skeleton-based models and unified solutions. These solutions can take either IMU measurements or a skeleton as input and can be effectively trained. Additionally, GLULA, due to its reliance on raw data and optimization via linear attention, demonstrated a significant speed-up compared to transformer-based solutions for skeletal data, which also perform graph manipulation before using it as input for the model.

Furthermore, the benefits of learning distillation on the teachers' hidden variables, which have been empirically established, are emphasized. Despite certain computational limitations, preliminary investigations indicate the potential for improved performance with a larger pre-training dataset and a more potent teacher model.

5.7 Comparative analysis of the proposed model on Body-Worn Sensors' Datasets

In 5.7.1, we provide a brief description of the networks employed in other HAR papers. These networks are then compared with the proposed model in 5.7.2, utilizing benchmark test sets.

5.7.1 Compared Algorithms

We conducted a comparison between the proposed models, GLULA-HAR and GLUSA-HAR, and several other existing methods:

Lego-CNN [35]: A lightweight convolutional neural network that employs memory-efficient lego-filters.

Self-Att [20]: A self-attention-based model that utilizes SA blocks along with sensor Modality Attention and global Temporal Attention. The evaluation of this model differs from others as it uses the $F1 - macro$ score instead of $F1 - weighted$. We followed the structure of the model and estimated the number of parameters. However, we couldn't reproduce the results on the PAMAP2 dataset due to a minor code oversight, where the label class was included as input during inference instead of being predicted. Therefore, we relied on the results presented in the papers [20] and [36].

DeepConvLSTM [23] (also known as DCL): A classical approach for HAR tasks that combines convolutional layers with recurrent units. The exact number of parameters may vary depending on different implementations. This model can be trained effectively with limited data, unlike self-attention-based models and transformers. Hence, we referred to the results presented in [19] and [36].

Attsense [19]: A multimodal model that incorporates attention-fusion subnets. It combines convolutional layers, Gated Recurrent Units, and attention mechanisms. This model extends the DeepConvLSTM and DeepSense models, and the number of parameters is similar to DeepConvLSTM. However, due to its reliance on recurrent mechanisms, it is less computationally efficient in terms of inference and training speeds.

R-CNN [22]: A convolutional neural network that has demonstrated high performance on various datasets. Although the original work did not cover all the datasets we used, we employed the official implementation and extracted the number of parameters.

HSA [36]: This model addresses the challenge of recognizing unseen activities through a hierarchical self-attention-based approach. It combines data from different sensor placements over time and incorporates a decoder that uses feature representations from the self-attention encoder for detecting unseen activities in open-set recognition. However, the encoder-decoder structure of hierarchical self-attention adds complexity. Similar to the Self-Att model, there was a misstep in the PAMAP2 dataset implementation where labels were included as part of the input during in-

Table 5.7: Size and scores (F1-weighted/F1-macro) comparison of the proposed model with listed methods on benchmark Body-Worn Sensors' datasets

	PAMAP2	SKODA	OPPORT.	USC-HAD	DAPH.	Num. of Params
GLULA	90.1/90.3	**97.6/96.3**	**95.9/78.0**	**59.4/50.9**	**94.1/79.7**	**50/51/196/4/4** K
GLUSA	89.9/88.0	**97.8/96.6**	95.5/72.2	54.3/46.5	92.4/72.8	*same as above*
Lego-CNN[35]	91.4/	-	85.5/	-	-	2.86M/-/610k/-/-
Self-Att[20]	/**95.0**	/93.0	/61.0	/50.0	82.0/	~590K
DCL[23]	74.8/	91.2/	67.0/	38.0/	84.0/	~331K
Attsense[19]	89.3/	93.1/	-	-	-	~331K
HSA[36]	**99.0**/	95.0/	68.0/	55.0/	85.0/	0.51/2.1/0.91/2.3/0.59 M
iSPLI[27]	89.0/	-	88.0/	-	**94.0**/	1.34M/-/1.35M/-/1.33M
R-CNN[22]	93.7/	-	92.1/	-	-	26M/-/30M/-/-

ference. We relied on the results from the paper [36], but this oversight should be addressed in future works.

iSPLI [27]: A deep learning model inspired by the Inception-ResNet structure introduced by Google. It emphasizes high predictive accuracy while utilizing fewer device resources. The authors tested the architecture on four datasets using transfer learning and believe that their work will establish a benchmark.

Although the proposed model can also be compared to classical machine learning solutions, it has been observed in [19] that machine learning models underperform compared to deep learning models across all benchmark datasets.

In conclusion, we have compared the models, GLULA-HAR and GLUSA-HAR, to various existing methods, considering their different architectures, performance metrics, and complexities.

5.7.2 The Analysis of Experimental Results on Body-Worn Sensors' data

In Table 5.7, the "Num. of Params" column provides four values representing the model sizes for the PAMAP2, SKODA, OPPORTUNITY, and USC-HAD datasets, respectively. If only one value is specified, it indicates that the network size remains relatively consistent with marginal changes across different datasets. The performance of the models is measured using the $F1 - weighted$ score, except for the Self-Att network [20], which utilizes the $F1 - macro$ score. We also calculated the macro

score for the proposed solution. However, for the DAPHNET dataset, we relied on the $F1-weighted$ result presented in [36], given that both works are from the same author, ensuring the reliability of the score numbers. In the table, to account for the score type differences, two scores are presented in the format of $F1-weighted/F1-macro$.

As shown, the proposed model, GLULA-HAR, achieves the highest scores in four benchmark datasets: USC-HAD, OPPORTUNITY, DAPHNET, and SKODA. GLUSA-HAR slightly outperforms the $F1w$ score for the SKODA dataset but falls behind in all other measurements.

On the PAMAP2 dataset, both models underperformed compared to R-CNN [22], HSA [36], and Self-Att [20]. The Self-Att network had eleven times more parameters than solutions, while HSA had ten times more parameters. GLULA-HAR had a slightly lower score than Lego-CNN but used 50 times fewer parameters than Lego-CNN [35]. When comparing GLULA with Attsense [19], a linear-attention-based solution, GLULA had over six times fewer parameters and a higher $F1-w$ score by almost 1%.

Since the SKODA dataset is less complex than others, all presented algorithms achieved an F1 score of no less than 91%. The proposed model obtained the highest results among recent works while maintaining a substantially lower number of parameters. For example, HSA had 40 times more parameters, while Attsense had six times more.

For the OPPORTUNITY dataset, the difference in sizes was less significant, but the performance improvement was notable. There were evident contrasts in parameter count and performance between GLULA and the classical DeepConvLSTM [23] as well as recent state-of-the-art models. Furthermore, Attsense, an expansion of DeepSense, which itself was based on DeepConvLSTM, exhibited a similar model size reduction compared to the proposed model.

In the USC-HAD dataset, the GLULA outperformed the previous state-of-the-art models while containing significantly fewer parameters than the HSA and Self-Att networks, which showed similar performance. GLULA, with only 4,000 parameters,

was much smaller compared to HSA, which had 2.3 million parameters.

The GLULA model for the DAPHNET dataset had an embedding dimension of 16, resulting in a model size almost equal to that of the USC-HAD case. The iSPLI network achieved an identical $F1-weighted$ performance score of 94.0%. The proposed model performed slightly better by 0.1% while containing 300 times fewer parameters. The second closest model in terms of performance was HSA with an F1-score of 85%.

Overall, GLULA-HAR consistently achieves similar or higher scores than the softmax self-attention-based proposed model in all benchmark datasets. It also offers higher performance speed and lower complexity. Among all known models, GLULA-HAR achieves the highest $F1-score$ results in the USC-HAD, SKODA, DAPHNET, and OPPORTUNITY datasets, surpassing state-of-the-art models. Additionally, the proposed solution exhibits significantly fewer parameters than the presented networks.

Chapter 6

Conclusion

In conclusion, this article addressed two critical challenges in Human Activity Recognition. Firstly, a unified solution that can work raw with several types of temporal data, in this case, body-worn inertial sensors' output and human joints' position coordinates in the form of skeleton-based data. Secondly, the size/performance trade-off in Human Activity Recognition (HAR) for mobile and embedded systems, is valuable in the production of monitoring systems that can follow human activity in real-time and control other systems based on that data. By correlating two data types together due to innate similarities, we can reconstruct both data in the same way and preserve raw data. Then, build a unified solution on top of that.

This work introduced GLULA, a novel approach for HAR that combines linear attention, GCN, and wide convolutions while adapting structures such as branching. By using linear attention instead of regular softmax self-attention, our model achieved faster speed and reduced time and memory complexity. The incorporation of non-recurrence and parallelization further enhanced the flexibility and efficiency of our network.

To reduce the model's parameter count, the use of feedforward (FF) layers was avoided and instead captured local dependencies more efficiently using GCN with branching. Our solution also employed various training techniques, including manifold mixup, one-cycle scheduling, and the AdaBelief optimizer, to enhance stability and handle limited data.

By extensive experiments on both skeleton-based data and body-worn inertial sensors' data and making minimal changes to the original data, it was shown that GLULA can be a unified solution for these data types.

In this work, substantial experiments were conducted on five benchmark datasets (USC-HAD, PAMAP2, SKODA, DAPHNET, and OPPORTUNITY) to evaluate the performance, speed, and variations of our model. The original GLULA network, along with the suggested training techniques, demonstrated comparable or superior results compared to other variants. The linear attention-based method outperformed regular softmax-based approaches in HAR tasks while exhibiting lower time and space complexity.

Our experiments also revealed that the proposed network outperformed state-of-the-art models on the SKODA, USC-HAD, DAPHNET, and OPPORTUNITY datasets while maintaining the lowest parameter count by a noticeable margin. This success can be attributed to the architectural structure of GLULA, which effectively captures local and global spatiotemporal features.

To show the validity of the method on skeleton-based data, extensive analysis was performed of the model by comparing it to a recent state-of-the-art transformer-based solution for skeletal data. While not getting new SOTA results on the NTU RGB+D dataset, GLULA demonstrated comparable performance depending on the scenario. Furthermore, with a larger pre-training dataset and a more potent teacher model in learning distillation, there is a potential to narrow the gap even further. Additionally, GLULA, due to its reliance on raw data and optimization via linear attention, demonstrated a significant speed-up (up to 1000x on larger sets) compared to transformer-based solutions for skeletal data, which also perform graph manipulation before using it as input for the model.

For future research, I suggest exploring the use of data generated by GANs and employing different augmentation techniques to further enhance the performance of HAR models. Additionally, considering the inspiration drawn from the Evolved Transformer and evolutionary-based neural architecture (AutoML) research, applying similar AutoML techniques to HAR tasks could lead to more robust solutions.

Another potential avenue would be to continue with knowledge distillation in this field or move to video-based activity recognition, where large language models and CLIP can be utilized [42].

Bibliography

[1] Marc Bachlin, Meir Plotnik, Daniel Roggen, Inbal Maidan, Jeffrey M Hausdorff, Nir Giladi, and Gerhard Troster. Wearable assistant for parkinson's disease patients with the freezing of gait symptom. *IEEE Transactions on Information Technology in Biomedicine*, 14(2):436–446, 2009.

[2] Paolo Bonato. Advances in wearable technology and applications in physical medicine and rehabilitation, 2005.

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

[4] Yann N Dauphin, Angela Fan, Michael Auli, and David Grangier. Language modeling with gated convolutional networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 933–941. JMLR. org, 2017.

[5] Yong Du, Wei Wang, and Liang Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1110–1118, 2015.

[6] Haodong Duan, Yue Zhao, Kai Chen, Dahua Lin, and Bo Dai. Revisiting skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2969–2978, 2022.

[7] Sumaira Ghazal, Umar S Khan, Muhammad Mubasher Saleem, Nasir Rashid, and Javaid Iqbal. Human activity recognition using 2d skeleton data and supervised machine learning. *IET image processing*, 13(13):2572–2578, 2019.

[8] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129:1789–1819, 2021.

[9] Saurabh Gupta. Deep learning based human activity recognition (har) using wearable sensor data. *International Journal of Information Management Data Insights*, 1(2):100046, 2021.

[10] Harish Haresamudram, Apoorva Beedu, Varun Agrawal, Patrick L Grady, Irfan Essa, Judy Hoffman, and Thomas Plötz. Masked reconstruction based self-supervision for human activity recognition. In *Proceedings of the 2020 International Symposium on Wearable Computers*, pages 45–49, 2020.

[11] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):221–231, 2012.

[12] Rafal Jozefowicz, Wojciech Zaremba, and Ilya Sutskever. An empirical exploration of recurrent network architectures. In *International conference on machine learning*, pages 2342–2350, 2015.

[13] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. *arXiv preprint arXiv:2006.16236*, 2020.

[14] Wazir Zada Khan, Yang Xiang, Mohammed Y Aalsalem, and Quratulain Arshad. Mobile phone sensing systems: A survey. *IEEE Communications Surveys & Tutorials*, 15(1):402–427, 2012.

[15] Raghuraman Krishnamoorthi. Quantizing deep convolutional networks for efficient inference: A whitepaper. *arXiv preprint arXiv:1806.08342*, 2018.

[16] Wenhao Li, Hong Liu, Runwei Ding, Mengyuan Liu, and Pichao Wang. Lifting transformer for 3d human pose estimation in video. *arXiv preprint arXiv:2103.14304*, 2:2, 2021.

[17] Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, and Alex C Kot. Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. *IEEE transactions on pattern analysis and machine intelligence*, 42(10):2684–2701, 2019.

[18] Mengyuan Liu, Hong Liu, and Chen Chen. Enhanced skeleton visualization for view invariant human action recognition. *Pattern Recognition*, 68:346–362, 2017.

[19] Haojie Ma, Wenzhong Li, Xiao Zhang, Songcheng Gao, and Sanglu Lu. Attnsense: multi-level attention mechanism for multimodal human activity recognition. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 3109–3115. AAAI Press, 2019.

[20] Saif Mahmud, M Tonmoy, Kishor Kumar Bhaumik, AKM Rahman, M Ashraful Amin, Mohammad Shoyaib, Muhammad Asif Hossain Khan, and Amin Ahsan Ali. Human activity recognition from wearable sensor data using self-attention. *arXiv preprint arXiv:2003.09018*, 2020.

[21] Diganta Misra. Mish: A self regularized non-monotonic neural activation function. *arXiv preprint arXiv:1908.08681*, 4, 2019.

[22] Fernando Moya Rueda, René Grzeszick, Gernot A Fink, Sascha Feldhorst, and Michael Ten Hompel. Convolutional neural networks for human activity recognition using body-worn sensors. In *Informatics*, volume 5, page 26. Multidisciplinary Digital Publishing Institute, 2018.

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

[24] Chiara Plizzari, Marco Cannici, and Matteo Matteucci. Spatial temporal transformer network for skeleton-based action recognition. In *Pattern Recognition. ICPR International Workshops and Challenges: Virtual Event, January 10–15, 2021, Proceedings, Part III*, pages 694–701. Springer, 2021.

[25] Attila Reiss and Didier Stricker. Creating and benchmarking a new dataset for physical activity monitoring. In *Proceedings of the 5th International Conference on PErvasive Technologies Related to Assistive Environments*, pages 1–8, 2012.

[26] Daniel Roggen, Alberto Calatroni, Mirco Rossi, Thomas Holleczek, Kilian Förster, Gerhard Tröster, Paul Lukowicz, David Bannach, Gerald Pirkl, Alois Ferscha, et al. Collecting complex activity datasets in highly rich networked sensor environments. In *2010 Seventh international conference on networked sensing systems (INSS)*, pages 233–240. IEEE, 2010.

[27] Mutegeki Ronald, Alwin Poulose, and Dong Seog Han. isplinception: an inception-resnet deep learning architecture for human activity recognition. *IEEE Access*, 9:68985–69001, 2021.

[28] Philip Schmidt, Attila Reiss, Robert Duerichen, Claus Marberger, and Kristof Van Laerhoven. Introducing wesad, a multimodal dataset for wearable stress and affect detection. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, pages 400–408, 2018.

[29] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1010–1019, 2016.

[30] Leslie N Smith. A disciplined approach to neural network hyper-parameters: Part 1–learning rate, batch size, momentum, and weight decay. *arXiv preprint arXiv:1803.09820*, 2018.

[31] David R So, Chen Liang, and Quoc V Le. The evolved transformer. *arXiv preprint arXiv:1901.11117*, 2019.

[32] Thomas Stiefmeier, Daniel Roggen, Georg Ogris, Paul Lukowicz, and Gerhard Tröster. Wearable activity tracking in car manufacturing. *IEEE Pervasive Computing*, 7(2):42–50, 2008.

[33] Thomas Stiefmeier, Daniel Roggen, and Gerhard Troster. Fusion of string-matched templates for continuous activity recognition. In *2007 11th IEEE International Symposium on Wearable Computers*, pages 41–44. IEEE, 2007.

[34] Guilherme Augusto Silva Surek, Laio Oriel Seman, Stefano Frizzo Stefenon, Viviana Cocco Mariani, and Leandro dos Santos Coelho. Video-based human activity recognition using deep learning approaches. *Sensors*, 23(14):6384, 2023.

[35] Yin Tang, Qi Teng, Lei Zhang, Fuhong Min, and Jun He. Efficient convolutional neural networks with smaller filters for human activity recognition using wearable sensors. *arXiv preprint arXiv:2005.03948*, 2020.

[36] M Tonmoy, Saif Mahmud, AKM Mahbubur Rahman, M Ashraful Amin, and Amin Ahsan Ali. Hierarchical self attention based autoencoder for open-set human activity recognition. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 351–363. Springer, 2021.

[37] Yao-Hung Hubert Tsai, Shaojie Bai, Makoto Yamada, Louis-Philippe Morency, and Ruslan Salakhutdinov. Transformer dissection: An unified understanding for transformer's attention via the lens of kernel. *arXiv preprint arXiv:1908.11775*, 2019.

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

[39] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning*, pages 6438–6447. PMLR, 2019.

[40] Feng Wang and Huaping Liu. Understanding the behaviour of contrastive loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2495–2504, 2021.

[41] Limin Wang, Yu Qiao, and Xiaoou Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4305–4314, 2015.

[42] Mengmeng Wang, Jiazheng Xing, and Yong Liu. Actionclip: A new paradigm for video action recognition. *arXiv preprint arXiv:2109.08472*, 2021.

[43] Felix Wu, Angela Fan, Alexei Baevski, Yann N Dauphin, and Michael Auli. Pay less attention with lightweight and dynamic convolutions. *arXiv preprint arXiv:1901.10430*, 2019.

[44] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vitpose: Simple vision transformer baselines for human pose estimation. *Advances in Neural Information Processing Systems*, 35:38571–38584, 2022.

[45] Santosh Kumar Yadav, Kamlesh Tiwari, Hari Mohan Pandey, and Shaik Ali Akbar. Skeleton-based human activity recognition using convlstm and guided feature learning. *Soft Computing*, pages 1–14, 2022.

[46] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[47] 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*, pages 351–360, 2017.

[48] Fanfan Ye, Shiliang Pu, Qiaoyong Zhong, Chao Li, Di Xie, and Huiming Tang. Dynamic gcn: Context-enriched topology learning for skeleton-based action recognition. In *Proceedings of the 28th ACM international conference on multimedia*, pages 55–63, 2020.

[49] Mi Zhang and Alexander A Sawchuk. Usc-had: a daily activity dataset for ubiquitous activity recognition using wearable sensors. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pages 1036–1043, 2012.

[50] Yuxuan Zhou, Chao Li, Zhi-Qi Cheng, Yifeng Geng, Xuansong Xie, and Margret Keuper. Hypergraph transformer for skeleton-based action recognition. *arXiv preprint arXiv:2211.09590*, 2022.