ID N. AIDR02/18801


UNIVERSITÀ CAMPUS BIO-MEDICO DI ROMA

DEPARTMENT OF ENGINEERING

Italian National Ph.D. in Artificial Intelligence
Health and Life Sciences
XXXVIII Cycle

# Artificial Intelligence Application to Lower Limb Rehabilitation Robotics

***Supervisors***

*Prof. Paolo Soda*

*Prof. Loredana Zollo*

***Candidate***

*Omar Coser*

March, 2026

To my family

# Acknowledgements

I would like to sincerely thank Professor Paolo Soda and Professor Loredana Zollo for their generous financial support through the PhD scholarship and for their continuous guidance and supervision throughout these three years. Their mentorship, insight, and encouragement have been fundamental to my academic and personal growth.

I am deeply grateful to Dr. Antonio Orvieto for his supervision and support during my time at the Max Planck Institute for Intelligent Systems and the ELLIS Institute Tübingen. The experience of conducting research in such stimulating and inspiring environments has greatly enriched my doctoral journey.

I would also like to acknowledge all the colleagues and friends I had the pleasure of meeting in Rome and in Tübingen. Your discussions, collaboration, and shared moments made these years both intellectually rewarding and personally meaningful.

I am especially thankful to my family my father, my mother, and my brother for their encouragement, and support throughout these three years. Without them, this achievement would not have been possible.

Finally, I would like to thank the few friends who have remained close since my bachelor's degree for the sporadic dinners and aperitifs, and for the constant reminder of the importance of friendship beyond academia.

# Abstract

Artificial Intelligence (AI) is increasingly transforming healthcare by enabling data-driven decision support, predictive modeling, personalized treatment planning, and continuous patient monitoring. From medical imaging and clinical risk stratification to wearable sensing and digital therapeutics, AI systems are reshaping how complex physiological signals are interpreted and translated into actionable knowledge. However, despite remarkable progress, several challenges limit the deployment of AI in real world clinical environments. Medical data is often heterogeneous, noisy, sparsely labeled, and characterized by high inter patient variability. Moreover, healthcare applications demand robustness, interpretability, safety, and generalization under distribution shifts requirements that are significantly more stringent than those in many conventional machine learning benchmarks.

These challenges become particularly evident in the domain of lower limb rehabilitation robotics. Neurological and musculoskeletal disorders such as stroke, Parkinson's disease, spinal cord injury, and age related degeneration frequently impair gait and mobility, severely affecting patients' independence and quality of life. Robotic exoskeletons and assistive devices have emerged as promising tools to support motor recovery through repetitive, task oriented, and adaptive training. By integrating biomechanical sensing, actuation, and control, these systems can deliver intensive and personalized therapy. Nevertheless, the effectiveness of robotic rehabilitation critically depends on accurate modeling and interpretation of human locomotion dynamics.

Human gait is a complex, nonlinear, and multi scale process involving coordinated interactions between joints, muscles, sensory feedback, and environmental context. Lower limb robotic systems must detect gait phases, recognize locomotion modes (e.g., level walking, stair ascent/descent, ramp negotiation), estimate kinematics, and adapt assistance in real time. Additionally, subtle deviations in gait patterns may serve as early indicators of pathological conditions, stress states, or neurodegenerative diseases such as Parkinson's disease. Designing AI models that can robustly capture these temporal dependencies from wearable sensors such as inertial measurement units (IMUs), electromyography (EMG), and pressure sensors remains an open research problem.

Traditional machine learning approaches often struggle to generalize across subjects and conditions due to limited labeled datasets and the intrinsic variability of clinical populations. Furthermore, many existing architectures either fail to capture long range temporal dependencies or are computationally inefficient for deployment on embedded robotic hardware. These limitations motivate the exploration of advanced neural network architectures capable of modeling structured temporal dynamics while maintaining efficiency and robustness.

This thesis aims to advance the field of AI-driven lower limb rehabilitation robotics by addressing two central challenges.

The first challenge concerns the investigation and comparative analysis of modern neural network architectures for human locomotion recognition. Specifically, this work systematically studies recurrent neural networks, convolutional temporal models, attention-based Transformers, state space models, and hybrid architectures for gait classification, terrain recognition, and regression tasks such as slope and stair height estimation. By evaluating these models on multimodal locomotion datasets, the thesis analyzes their ability to capture short and long range dependencies, handle sensor noise, and generalize across different walking conditions. Particular attention is given to computational complexity, and suitability for real time robotic control. The results contribute to a clearer understanding of which modeling paradigms are most appropriate for continuous locomotion analysis and exoskeleton assisted rehabilitation.

The second challenge addresses the role of Self-PreTraining (SPT) in medical time series modeling. In many healthcare scenarios, labeled data is scarce and expensive to obtain, while large amounts of unlabeled physiological signals are available. Self-pretraining strategies offer a promising solution by enabling models to learn meaningful representations from unlabeled data before fine tuning on downstream supervised tasks. However, the mechanisms through which self-pretraining improves performance—particularly in structured sequence models remain insufficiently understood, especially in medical contexts.

This thesis investigates how self-pretraining affects sequence classification architectures, with a focus on Transformer-based models and their internal attention dynamics. Through controlled experiments and ablation studies, the work examines whether performance gains stem from better initialization of specific parameters, improved representation learning, or enhanced optimization landscapes. By analyzing weight norms, parameter displacement, freezing strategies, and positional encoding interactions, the thesis provides empirical insights into how SPT shapes learned representations and facilitates downstream adaptation.

Importantly, the benefits of self-pretraining are evaluated not only in locomotion recognition tasks but also in broader medical time series applications, including stress detection and Parkinson's disease detection. Stress detection from physiological signals requires the

identification of subtle temporal patterns in heart rate variability, motion, and other biosignals. Parkinson's detection, particularly through gait and movement analysis, demands high sensitivity to micro variations in motor dynamics. In both cases, labeled datasets are limited and heterogeneous, making them ideal candidates for representation learning through self-pretraining.

Experimental results demonstrate that SPT can improve classification accuracy across multiple medical time series datasets, particularly in low data regimes. The findings suggest that self-pretraining enhances the model's ability to encode structured temporal information and accelerates convergence during fine tuning. Furthermore, different masking strategies and architectural configurations are analyzed to determine how pretraining design choices influence downstream performance. The thesis highlights that the impact of SPT varies with model depth, dataset characteristics, and signal structure, providing practical guidance for its application in healthcare oriented sequence modeling.

This work contributes to the development of robust, scalable, and data efficient AI methodologies for lower limb rehabilitation robotics and related medical time series tasks. By integrating architectural advances in long sequence modeling with principled analysis of self-pretraining mechanisms, the thesis bridges theoretical insights and applied clinical challenges. The proposed approaches support more accurate human locomotion recognition, improved stress and Parkinson's detection, and enhanced adaptability of robotic rehabilitation systems.

In doing so, this research advances the vision of intelligent, human centered rehabilitation technologies that leverage AI not merely as a predictive tool, but as an enabling framework for personalized, adaptive, and clinically meaningful motor recovery.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

Artificial Intelligence (AI) has become a central driver of innovation in healthcare, enabling new capabilities in diagnosis, decision support, patient monitoring, and personalized rehabilitation. As part of this doctoral research in AI and healthcare, the focus lies on exploring how modern machine learning methods—particularly representation learning and data driven modeling—can enhance and support effective recovery pathways for patients with motor impairments.

One rapidly developing and clinically impactful area is *lower limb rehabilitation robotics*. These systems aim to restore or improve mobility in individuals affected by conditions such as stroke, spinal cord injury, neurodegenerative disorders, or orthopedic trauma. By combining robotics, biomechanics, sensing technologies, and AI-driven algorithms, rehabilitation robots can provide precise, repeatable, and adaptive training that complements traditional physiotherapy. Their ability to capture rich physiological and biomechanical signals during movement creates new opportunities for data centric approaches to rehabilitation.

AI plays an increasingly important role in this domain. Machine learning models can help interpret complex time series data from wearable sensors and robotic actuators, estimate gait quality, predict patient progress, and personalize therapy in real time. Furthermore, advances in self supervised learning and pretraining methods hold the potential to extract clinically meaningful representations even from limited labeled datasets—an important capability in healthcare settings where annotation is costly and patient variability is high.

This thesis aims to integrate these trends by investigating AI methodologies that support assessment, personalization, and decision making in lower limb rehabilitation robotics. The work builds toward the development of robust models that operate in real world clinical contexts, addressing challenges such as irregular signals, inter patient heterogeneity, and limited annotated data. The following sections provide a detailed overview of the relevant background, related work, and methodological foundations that motivate this research direction.

**Scope of the Thesis.** Lower limb rehabilitation robotics is a broad field encompassing multiple therapeutic modalities and motor tasks, including post-surgical recovery, balance training, functional electrical stimulation, sit-to-stand assistance, stair negotiation, and multi-joint motor re-education. Representative systems in this broader domain include treadmill-based robotic gait trainers such as the Lokomat as well as wearable powered exoskeletons designed for overground walking assistance [1].

This thesis deliberately restricts its scope to gait-related rehabilitation and, more specifically, to intelligent lower limb exoskeleton systems designed to support or enhance human walking, such as wearable hip–knee powered exoskeletons and assistive robotic suits [2]. Throughout this work, the term *lower limb rehabilitation robotics* therefore refers to robotic platforms and AI-driven control strategies focused on modeling, recognizing, and adapting human locomotion during walking tasks.

By concentrating on gait analysis and exoskeleton-assisted walking, the thesis maintains a focused methodological investigation while acknowledging that the broader domain of lower limb rehabilitation extends beyond the boundaries addressed here.


(a) The ReWalk 7 exoskeleton

(b) The ReWalk 7 exoskeleton

Figure 1.1: This photo was taken at Università Campus Bio-Medico di Roma.

## 1.1 AI Architectures and Learning Paradigms for Rehabilitation Robotics and Diagnosis Improvement

### 1.1.1 Neural architectures for Lower limb rehabilitation robotics

#### Technological Needs and Modeling Challenges in Lower Limb Rehabilitation Robotics

In the rapidly advancing field of rehabilitation engineering and robotic assistance, the accurate analysis and interpretation of human locomotion have become fundamental to the development of effective lower limb robotic systems. Robotic exoskeletons and assistive devices aim to restore or improve mobility in individuals affected by stroke, spinal cord injury, Parkinson's disease, and other neurological or musculoskeletal conditions. To achieve clinically meaningful outcomes, these systems must continuously monitor, interpret, and respond to complex biomechanical and neuromuscular signals during walking and other gait-related activities.

Human locomotion analysis involves the measurement, interpretation, and prediction of joint kinematics, muscle activations, ground reaction forces, and user intent. This capability is central to multiple applications: rehabilitation robotics where detailed characterization of impaired gait is essential for personalized therapy design; exoskeleton control where real-time adaptation to user intent and biomechanical states is required; prosthetic systems where restoring natural and efficient gait relies on accurate prediction of joint trajectories and interaction forces; and clinical monitoring where subtle gait deviations may serve as early indicators of neuromuscular or musculoskeletal pathologies.

Despite its importance, locomotion modeling presents substantial technical challenges. Gait is inherently nonlinear and multi-scale, involving interactions between muscles, joints, sensory feedback, and environmental context. Fine grained dynamics occur within individual steps, while longer term dependencies extend across complete gait cycles and prolonged walking sessions. Moreover, patients exhibit significant variability due to morphology, pathology, fatigue, walking speed, and adaptation to assistive devices.

Traditional analytical and rule-based control methods often struggle to maintain robustness and generalization across users and conditions, particularly when real time performance is required for safety critical robotic applications. In addition, rehabilitation systems must operate under strict computational and latency constraints, as many devices rely on embedded hardware with limited resources. Finally, clinical datasets are typically small and heterogeneous, making data efficiency and generalization under distribution shifts central

concerns.

These challenges define the core technological needs of lower limb rehabilitation robotics:

- Robust modeling of multi-scale temporal dependencies in gait signals;
- Generalization across heterogeneous patients and locomotion conditions;
- Computational efficiency suitable for real-time and embedded deployment;
- Data efficient learning in settings characterized by limited supervision.

Recent advances in machine learning and deep neural networks offer promising tools to address these needs. In particular, modern sequence modeling architectures are capable of learning structured temporal dependencies, recognizing subtle motion patterns, and predicting joint behavior under varying contexts. The central hypothesis motivating this thesis is that different architectural paradigms provide distinct inductive biases that may address specific requirements of rehabilitation robotics.

Several major computational paradigms have shaped recent progress in locomotion analysis.

Attention-based models, inspired by the Transformer architecture proposed by Vaswani et al. [3], enable the integration of long-distance temporal relationships in gait data, allowing systems to reason over entire gait cycles rather than isolated frames. Extensions such as Transformer-XL [4] and sparse attention mechanisms [5] improve modeling efficiency for long-duration locomotion datasets, where high-frequency sensors generate large volumes of sequential data.

Recurrent neural networks, once the dominant approach in sequential modeling, continue to offer advantages for continuous lower limb dynamics, particularly through improved variants that mitigate vanishing gradients and support long-horizon biomechanical prediction. Convolutional neural networks provide an alternative perspective by exploiting hierarchical feature extraction, making them effective for recognizing spatial and spatiotemporal gait signatures. More recently, state space models (SSMs) have emerged as a promising solution, offering computationally efficient mechanisms for modeling long-range temporal dependencies while maintaining real-time performance essential for exoskeleton control [6].

A notable development is the emergence of hybrid architectures that combine attention, recurrence, convolution, and SSM formulations. Such models aim to leverage complementary strengths—local feature extraction, temporal memory, and global context reasoning—to handle diverse locomotion tasks including gait phase detection, intention recognition, trajectory prediction, and adaptive control within a unified framework.

This thesis builds upon these developments by systematically investigating how different architectural principles respond to the operational requirements of lower limb rehabilitation robotics. Rather than presenting sequence modeling techniques as isolated methodological advances, this work evaluates them through the lens of application-driven constraints: robustness, generalization, computational efficiency, and data efficiency.

The following sections synthesize current advancements in AI-based locomotion analysis for lower limb robotics, providing an integrated perspective on the architectures, methods, and applications driving progress in the field. By grounding architectural analysis in real clinical and robotic needs, this review establishes the conceptual foundation for the comparative study presented in Chapter 3.

### Current Review on the Topic

In recent years, the integration of artificial intelligence into lower limb robotic rehabilitation has attracted substantial interest, leading to a rapidly expanding body of scientific literature. Several review studies have examined different aspects of lower limb rehabilitation robots, including mechanical design, actuation systems, sensing technologies, control strategies, and human–robot interaction. However, the extent to which modern AI architectures specifically address the modeling challenges of human locomotion remains only partially explored.

Shi et al. [7] provide an extensive overview of lower limb rehabilitation exoskeletons, focusing primarily on mechanical structures, actuation principles, and classical control approaches. Although their survey is not centered on artificial intelligence, it clearly identifies the limitations of traditional model-based and rule-based controllers in handling the variability and nonlinear dynamics of human gait. This observation implicitly motivates the need for data-driven and learning-based solutions.

Similarly, Baud et al. [8] review control strategies for lower limb exoskeletons aimed at gait assistance, discussing impedance-based methods, adaptive controllers, and shared control frameworks. Their analysis highlights the increasing complexity of control requirements in personalized rehabilitation, further emphasizing the necessity for intelligent models capable of capturing multi-scale locomotion dynamics.

More recently, Shi et al. [9] presented a focused review on human–robot coordination control in rehabilitation robotics. Their work stresses the importance of motor function assessment, real-time adaptability, and individualized control strategies for patients with heterogeneous neurological or orthopedic impairments. The authors argue that effective human–robot coordination requires models capable of learning structured temporal dependencies in locomotion signals, reinforcing the relevance of advanced sequence modeling approaches.

Belal et al. [10] provide the first dedicated review explicitly centered on deep learning approaches for lower limb exoskeleton control. Their survey synthesizes methods for gait mode recognition, kinematic prediction, electromyography decoding, sensor fusion, and control policy learning. The authors report significant performance improvements achieved by deep architectures compared to classical machine learning techniques, particularly in tasks requiring long-range temporal reasoning and robust feature extraction in noisy sensing environments. However, their analysis primarily focuses on application-level performance rather than on a systematic comparison of architectural principles and their inductive biases.

Within this context, the review by Coser et al. [11] represents a more focused effort to analyze AI-based methodologies specifically in exoskeleton-assisted lower limb rehabilitation. Compared to the aforementioned surveys, this work places greater emphasis on the categorization of machine learning and deep learning approaches according to their functional roles such as locomotion classification, intention detection, trajectory prediction, and adaptive control and discusses their potential impact on personalization and autonomy in clinical settings.

### Type of Architectures

#### Attention-based Models

Attention-based architectures have emerged as a dominant paradigm for modeling long range temporal dependencies in sequential data. The Transformer architecture, originally introduced by Vaswani et al. [3], represents a fundamental shift from recurrent and convolutional designs by relying exclusively on self-attention mechanisms. Through an encoderdecoder structure enriched with positional encodings, Transformers enable efficient parallel processing and effective modeling of long temporal contexts, making them particularly suitable for sequence modeling tasks.

Despite their success, standard Transformers suffer from quadratic time and memory complexity with respect to sequence length, limiting their applicability to long duration signals. Several extensions have been proposed to address this limitation. Transformer-XL [12] introduces segment level recurrence and relative positional encodings, enabling context reuse across segments and improving long term dependency modeling. Sparse attention mechanisms further reduce computational complexity by restricting attention patterns, as demonstrated by Sparse Transformers [5], which enable efficient processing of extremely long sequences.

Memory augmented variants extend the Transformer's temporal horizon by retaining compressed or recurrent representations of past activations. Compressive Transformers [13]

store summarized representations of distant context, while Recurrent Memory Transformers [14] introduce memory tokens to maintain continuity across segments without modifying the core architecture. These approaches significantly improve long range modeling while controlling memory growth.

A parallel line of research focuses on reducing attention complexity through low rank approximations and kernel based formulations. Linformer [15] approximates the self-attention matrix with a low rank projection, achieving linear complexity without substantial performance degradation. Similarly, linear Transformers [16] reformulate attention as a kernelized dot product, enabling efficient autoregressive inference and revealing connections between Transformers and recurrent models. Other efficient designs include Reformer [17], which combines locality sensitive hashing with reversible layers, and Longformer [18], which integrates local and global attention patterns to scale linearly with sequence length. LongT5 [19] further extends these concepts within an encoderdecoder framework for long context language understanding.

Recent contributions have also explored alternative attention formulations and architectural simplifications. CosFormer [20] replaces softmax attention with cosine based reweighting to stabilize training and reduce complexity, while FlashAttention [21] and FlashAttention-2 [22] optimize exact attention computation by minimizing memory access overhead on modern GPUs. In parallel, architectural simplifications have been proposed to streamline Transformer blocks without sacrificing performance, as demonstrated by He et al. [23].

Beyond sequence modeling, attention-based architectures have been adapted to visual domains. AutoFormer [24] employs neural architecture search to identify efficient Transformer configurations for visual recognition, while Transformer-in-Transformer (TNT) [25] introduces a hierarchical attention mechanism to jointly model local and global visual structures. Additionally, analytical studies such as LEGO [26] provide insights into attention dynamics and reasoning capabilities within Transformer models.

Collectively, these developments demonstrate the versatility and scalability of attention-based architectures for modeling long range temporal dependencies. In the context of human locomotion analysis and lower limb robotics, such models offer strong potential for capturingmulti scale temporal patterns in biomechanical and sensor data, enabling more accurate prediction, classification, and adaptive control strategies.

Tables 1.1 and 1.2 summarize representative attention-based architectures proposed for long range sequence modeling. The tables compare the underlying design principles, computational complexity, benchmark performance, and potential advantages for human locomotion modeling and lower limb robotic applications.

Table 1.1: Attention-based models relevant for long range reasoning in sequence data (Part 1).

| **Publication** | **Problem Addressed** | **Model** | **Key Techniques** | **Dataset Used** | **Complexity** | **Results** | **Advantages for Locomotion Modelling** |
|---|---|---|---|---|---|---|---|
| [3] (2017) | Sequence transduction with long range dependencies | Transformer | Multi head self attention, positional encoding | WMT14 EN–DE/EN–FR | $O(n^2d)$ | SOTA BLEU scores on MT benchmarks | Strong baseline for long range temporal structure (e.g., full gait cycles). |
| [12] (2018) | Modeling longer term dependencies in language | Transformer-XL | Segment level recurrence, relative positional encoding | WikiText-103, Enwik8, PTB, One Billion Word | $O(nd)$ | SOTA perplexity on WikiText-103 and Enwik8 | Extends effective context window; useful for multi cycle locomotion trajectories. |
| [5] (2019) | Efficient long sequence modeling | Sparse Transformer | Sparse factorizations, strided and fixed attention patterns | CIFAR-10, Enwik8, ImageNet-64, music datasets | $O(n\sqrt{n})$ | SOTA on CIFAR-10 and Enwik8 bits/dim/byte | Reduces compute/memory for long EMG/IMU streams in gait analysis. |
| [13] (2019) | Long range sequence modeling with memory compression | Compressive Transformer | Compressed memory, reconstruction loss, conv based compression | WikiText-103, Enwik8, PG-19, speech datasets | $O(n^2d)$ | SOTA on long text benchmarks | Handles long multi minute locomotion recordings via compressed history. |
| [15] (2020) | Efficient attention for long sequences | Linformer | Low rank approximation of self attention, linear projections | BookCorpus, Wikipedia, GLUE, IMDB | $O(nk)$, $k \ll n$ | Similar or slightly better than full Transformer on NLP tasks | Linear time attention suitable for real time exoskeleton control pipelines. |
| [17] (2020) | Memory efficient long sequence processing | Reformer | LSH attention, reversible layers, chunked processing | Enwik8, ImageNet-64, WMT14 EN–DE | $O(n \log n)$ | Comparable accuracy to Transformer with much lower memory | Enables deployment on resource constrained wearable/robotic hardware. |
| [18] (2020) | Efficient training on very long documents | Longformer | Sliding window local attention, global tokens | Text8, Enwik8, TriviaQA, WikiHop, HotpotQA, arXiv summarization | $O(nw + gn)$ | SOTA on long document QA/-summarization tasks | Local+global pattern capture resembles local joint motion + global gait phase context. |
| [16] (2020) | Linear time attention for autoregressive tasks | Linear Transformer | Kernel-based linearized self-attention | MNIST, CIFAR-10, WSJ speech | $O(n)$ | Comparable to softmax attention; very fast inference | Linear complexity attractive for onboard gait prediction / intent decoding. |
| [24] (2021) | Automatic design of vision transformers | AutoFormer | Weight sharing supernet, NAS based transformer search | ImageNet, CIFAR-10/100, Oxford Flowers | $\approx O(nd)$ | Strong ImageNet accuracy (up to 82.4% Top-1) | NAS ideas could transfer to auto designed locomotion architectures. |

Table 1.2: Attention-based models relevant for long range reasoning in sequence data (Part 2).

| **Publication** | **Problem Addressed** | **Model** | **Key Techniques** | **Dataset Used** | **Complexity** | **Results** | **Advantages for Locomotion Modelling** |
|---|---|---|---|---|---|---|---|
| [25] (2021) | Better local+global feature representation in images | TNT (Transformer-in-Transformer) | Inner/outer transformer blocks, patch hierarchies | ImageNet, CIFAR, ADE20K | $O(nd)$ | 81.5% Top-1 on ImageNet (TNT-S) | Hierarchical design analogous to joint level vs whole body locomotion structure. |
| [19] (2021) | Scaling T5 style models to long inputs | LongT5 | TGlobal attention, PEGASUS style pretraining | CNN/DailyMail, PubMed, arXiv, BigPatent, TriviaQA | $O(nr + ng)$ | SOTA on several summarization tasks | Shows how encoder–decoder models can be adapted to long sensor sequences. |
| [20] (2022) | Efficient and accurate linear attention | cosFormer | Cosine based reweighting, non-negative similarity scores | WikiText-103, LRA, GLUE, IMDB, Amazon | $O(n)$ | SOTA on Long Range Arena | Linear complexity attention with strong long range performance for gait tasks. |
| [26] (2022) | Understanding reasoning patterns in transformers | LEGO Attention | Structured synthetic tasks, interpretable attention patterns | Synthetic LEGO reasoning tasks | $O(n^2)$ | Shows inductive biases and generalization behavior | Offers insights for designing interpretable locomotion reasoning modules. |
| [14] (2022) | Long sequence modeling with compact memory | Recurrent Memory Transformer (RMT) | Memory tokens, segment recurrence, BPTT | WikiText-103, Enwik8, synthetic memory tasks | $O(n^2)$ | Competitive with Transformer-XL on memory intensive tasks | Maintains a compact "gait memory" over long trials or sessions. |
| [21] (2022) | Faster and memory efficient attention computation | FlashAttention | IO aware tiling, online softmax, GPU optimization | GPT style language models, benchmarks | $O(n^2)$ (but IO optimal) | Large speedup + reduced memory vs. standard attention | Important for practical training/inference of long context gait models. |
| [22] (2023) | Further improving attention throughput | FlashAttention-2 | Better parallelism, work partitioning, kernel fusion | Large LMs and synthetic benchmarks | $O(n^2)$ (hardware optimized) | Up to 3× faster than FlashAttention | Enables long context models usable in near real time clinical pipelines. |
| [23] (2024) | Simplifying transformer blocks with similar performance | Simplified Transformer Block | Removes some skip connections and projection params | CodeParrot, GLUE, BERT benchmarks | $O(n^2)$ | Similar accuracy with fewer parameters and faster training | Reduced complexity is attractive for embedded gait or exoskeleton systems. |

### Recurrent-Based Models

Recurrent neural networks (RNNs) have historically played a central role in sequence modeling due to their inherent ability to process temporal data through hidden state recurrence. However, classical RNN formulations suffer from vanishing and exploding gradient problems, which limit their effectiveness when modeling long range dependencies. As a result, a substantial body of research has focused on improving the stability, expressiveness, and scalability of recurrent architectures.

Early advances in this direction include the Unitary Evolution Recurrent Neural Network (uRNN), introduced by Arjovsky et al. [27]. By constraining recurrent weight matrices to be unitary, uRNNs preserve the norm of hidden states over time, enabling stable gradient propagation across long sequences. The model leverages complex valued representations and the modReLU activation function, demonstrating superior performance over standard RNNs and LSTMs on long memory tasks such as sequential MNIST and synthetic recall benchmarks.

Further improvements have targeted the gating mechanisms of recurrent architectures. Gu et al. [28] propose refinements to LSTM style gates through a refine gate mechanism and uniform gate initialization, which mitigate gate saturation and improve gradient flow across diverse temporal scales. These modifications enhance training stability and performance across a wide range of tasks, including language modeling and reinforcement learning, while retaining the simplicity of gated recurrence.

More recently, renewed interest in recurrent models has led to architectures capable of competing with modern state space models (SSMs) on long range benchmarks. Orvieto et al. [29] introduce the Linear Recurrent Unit (LRU), which employs linear recurrences with stable parameterizations to address long range dependency modeling efficiently. Despite its simplicity, LRU matches the performance of advanced SSMs such as S4 on the Long Range Arena benchmark, while maintaining fast training and inference.

Hierarchical gating strategies have also been explored to capturemulti scale temporal dependencies within recurrent models. Qin et al. [30] propose the Hierarchically Gated Recurrent Network (HGRN), in which gating dynamics vary across layers, allowing lower layers to model short term dependencies and higher layers to retain long term information. This design improves both accuracy and efficiency on long range sequence tasks and demonstrates strong performance relative to Transformers and other RNN variants.

In parallel, extensions of the Long Short Term Memory architecture have been proposed to enhance memory capacity and parallelization. Beck et al. [31] introduce xLSTM, which incorporates exponential gating and matrix based memory representations to improve scalability and long context modeling. By stacking scalar memory and matrix memory LSTM

blocks, xLSTM achieves competitive performance with Transformers and SSMs onlarge scale language modeling benchmarks.

Building on hierarchical recurrence, Qin et al. [32] present HGRN2, which extends HGRN through an outer product based state expansion mechanism. This design increases memory capacity without a proportional increase in parameters and enables efficient training via linear attention principles. HGRN2 demonstrates strong performance on long context reasoning and retrieval tasks, competing effectively with modern architectures such as Mamba while retaining the advantages of recurrent computation.

Table 1.3 summarizes representative recurrent-based architectures for long range sequence modeling, highlighting their design principles, computational complexity, benchmark performance, and potential advantages for human locomotion modeling. These models demonstrate that, when carefully designed, recurrent architectures remain a viable and competitive solution for continuous gait analysis, long duration rehabilitation sessions, and real time lower limb robotic control.

Table 1.3: RNN-based architectures for long range sequence modeling.

| **Publication** | **Problem Addressed** | **Model** | **Key Techniques** | **Dataset Used** | **Complexity** | **Results** | **Advantages for Locomotion Modelling** |
|---|---|---|---|---|---|---|---|
| [33] (2016) | Learning very long term dependencies | uRNN | Unitary weight matrices, complex valued states, modReLU | Copy task, Adding task, pixel MNIST | $\sim O(n)$ | Outperforms LSTMs on long memory tasks | Stable gradients for long gait sequences, robust to many time steps. |
| [28] (2020) | Improving gradient flow in LSTM gates | URLSTM | Uniform Gate Initialization, refine gate mechanism | Copy/Adding tasks, SeqMNIST, sCIFAR-10, WikiText-103 | $O(n)$ | Better than standard LSTMs on long term tasks | More stable learning for long trials of locomotion or rehab sessions. |
| [29] (2023) | Reviving RNNs for long range tasks | Linear Recurrent Unit (LRU) | Linear recurrences, complex diagonal matrices, stable parametrization | Long Range Arena, synthetic tasks | $O(n)$ | Matches SSMs (S4/S5) on LRA tasks | Efficient and stable recurrence suitable for continuous gait streaming. |
| [30] (2023) | Efficient RNNs with hierarchical dependencies | HGRN | Hierarchically gated recurrence, complex valued dynamics | WikiText-103, ImageNet-1K, LRA | $O(nd)$ | Matches/surpasses Transformers/RNNs on long range tasks | Captures both short and long term gait patterns in one model. |
| [31] (2024) | Scaling LSTM for language modeling | xLSTM | Exponential gating, matrix memory, covariance updates | SlimPajama, WikiText-103, PALOMA, LRA | $O(nd^2)$ | Outperforms Transformers/SSMs in some long context tasks | Rich memory useful for multi modal locomotion signals (EMG + kinematics). |
| [32] (2024) | More expressive hierarchical RNNs | HGRN2 | Outer product state expansion, parameter efficient projections | WikiText-103, The Pile, SlimPajama, LRA | $O(nd^2)$ | Outperforms HGRN1, competitive with sub-quadratic models | High capacity RNN for complex locomotion control or intent inference. |

Convolutional Models

Convolutional neural networks (CNNs) provide a fundamentally different approach to sequence modeling by exploiting local receptive fields and weight sharing to extract hierar-

chical features. Early convolutional models already demonstrated the potential of localized and multiresolution learning for temporal data. Bakshi and Stephanopoulos [34] introduced Wave-Net, a hierarchical architecture based on orthonormal wavelets that enables efficient multiresolution analysis with explicit control over local and global error bounds. By leveraging wavelet bases, Wave-Net achieves improved localization in both time and frequency domains, making it suitable for structured time series modeling.

The modern formulation of convolutional networks for temporal and spatial data was established by LeCun and Bengio [35], who demonstrated that local connectivity, shared weights, and subsampling provide robustness to shifts and distortions while significantly reducing the number of parameters. These principles underpin the widespread adoption of CNNs in image, speech, and time series analysis, and remain highly relevant for biomechanical signal processing in locomotion studies.

To extend convolutional models to long temporal contexts, Temporal Convolutional Networks (TCNs) were proposed by Lea et al. [36]. TCNs employ stacks of dilated temporal convolutions within encoderdecoder or fully convolutional structures, enabling efficient modeling of long range dependencies without recurrent computation. These architectures outperform recurrent models on action segmentation tasks and offer faster training, making them attractive for real time gait phase detection and locomotion event segmentation.

Dilated convolutions have also been used to expand receptive fields while preserving resolution in dense prediction tasks. CSRNet [37] demonstrates how dilated convolutional layers enable accurate modeling of spatially dense signals by capturing wider contextual information without excessive pooling. Although originally developed for crowd analysis, this design principle is directly applicable to dense temporal sensing in locomotion and rehabilitation contexts.

Hybrid architectures combining convolutional and attention mechanisms further enhance sequence modeling capabilities. The Conformer architecture proposed by Gulati et al. [38] integrates convolutional layers within Transformer blocks, allowing simultaneous modeling of local patterns and global dependencies. This combination achieves state of the art performance in speech recognition and highlights the complementary strengths of convolutional and attention-based processing for structured temporal signals.

Recent research has revisited pure convolutional approaches as competitive alternatives to attention and state space models for long range sequence modeling. Li et al. [39] introduce Structured Global Convolution (SGConv), which employs efficiently parameterized convolution kernels with decaying structures to capture long range dependencies while maintaining favorable computational complexity. Similarly, Fu et al. [40] propose hardware-efficient long convolutions with regularized kernels and IO-aware implementations, demonstrating perfor-

mance comparable to state space models while significantly improving runtime efficiency. These advances suggest that convolutional architectures can scale effectively to very long sequences without quadratic complexity.

Table 1.4 summarizes representative convolution-based architectures for long range sequence modeling, highlighting their design strategies, computational complexity, benchmark performance, and advantages for human locomotion modeling. Collectively, these models demonstrate that convolutional approaches remain a powerful and efficient solution for processing high frequency biomechanical signals, long-duration gait recordings, and real time control data in lower limb robotic systems.

Table 1.4: Convolution-based architectures for long range sequence modeling.

| **Publication** | **Problem Addressed** | **Model** | **Key Techniques** | **Dataset Used** | **Complexity** | **Results** | **Advantages for Locomotion Modelling** |
|---|---|---|---|---|---|---|---|
| [36] (2017) | Action segmentation and detection in videos | Temporal Conv Nets (TCN) | Dilated temporal convolutions, encoder–decoder structure, pooling/upsampling | 50Salads, MERL Shopping, GTEA | $O(nd)$ | Outperforms BiLSTMs, reduces oversegmentation errors | Well suited for gait phase segmentation and event detection in locomotion data. |
| [39] (2022) | long sequence modeling via structured convolutions | SGConv (Structured Global Conv) | Multiscale convolution kernels, efficient parameterization, FFT-based computation | Long Range Arena, Speech Commands, ImageNet, WikiText-103 | $O(n \log n)$ | Achieves SoTA on LRA tasks, surpassing S4 | Efficient long range receptive fields for high frequency locomotion signals. |
| [41] (2023) | Efficient long convolution for sequences | Long Convolution (FLASH-BUTTERFLY) | Butterfly decompositions, SQUASH/SMOOTH regularization, IO-aware GPU kernels | LRA, Path256, OpenWebText, The Pile, CIFAR | $O(n \log n)$ | Matches or exceeds SSMs/-Transformers; 7.2× faster on Path256 | Scales to very long IMU/EMG recordings in locomotion tasks. |
| [42] (2019) | Monaural speech enhancement in noise | Gated Residual Network (GRN) | Dilated convolutions, gated linear units (GLU), residual and skip connections | WSJ0 SI-84, NOISEX-92 and others | $O(nk)$ | Outperforms DNN, LSTM, BLSTM in STOI/PESQ | Gated residual convs can transfer to robust gait/EMG feature extraction. |
| [43] (2023) | Replacing attention with efficient convolutions | Hyena | Implicit long convolutions, multiplicative gating, Toeplitz/diagonal structure | WikiText-103, The Pile, PG-19, ImageNet | $O(n \log n)$ | Matches Transformer performance with fewer FLOPs; 100× faster at 64k tokens | Strong candidate for long locomotion sequences without quadratic attention costs. |

### Statehigh frequencySpace Models

Statehigh frequencyspace models (SSMs) have recently emerged as a powerful alternative to recurrent, convolutional, and attention-based architectures for long range sequence modeling. By formulating temporal dynamics through latent state evolution and structured parameterizations, SSMs achieve favorable scalability while preserving the ability to capture long-term dependencies.

A major breakthrough in this direction is the Structured State Space Sequence model

(S4), introduced by Gu et al. [44]. S4 builds upon classical state space formulations by reparameterizing the state transition matrices using diagonal-plus-low rank structures and HiPPO-based representations. This design enables efficient convolutional computation via FFTs, significantly reducing memory and computational costs while maintaining strong performance on long range benchmarks such as the Long Range Arena and sequential CIFAR-10. S4 demonstrated that SSMs can match or surpass Transformers on tasks involving sequences of up to tens of thousands of steps.

Building on S4, several works have focused on improving expressiveness and hardware efficiency. Fu et al. [41] propose the H3 layer, which combines multiple discrete SSMs through multiplicative interactions, enabling richer token-wise comparisons and improved long range reasoning. The same work introduces FlashConv, an FFT-based implementation that substantially accelerates SSM training and inference on modern hardware, allowing hybrid SSM–attention models to outperform Transformers in large scale language modeling tasks.

Further simplification and scaling of structured SSMs are achieved with S5, introduced by Smith et al. [45]. Unlike S4, which relies on multiple single-input single-output (SISO) systems, S5 employs a multi input multi output (MIMO) formulation with efficient parallel scans. This design preserves the performance of S4 while reducing architectural complexity and improving computational efficiency, achieving state-of-the-art results on benchmarks such as the Long Range Arena and Path-X.

Recent efforts have also explored hierarchical and multi scale extensions of state space models. Bhirangi et al. [46] introduce HiSS, a hierarchical SSM architecture designed for continuous sensor data. By stacking SSMs operating at different temporal resolutions, HiSS effectively captures both local and global dynamics, significantly outperforming flat SSMs, Transformers, and Mamba on real world sensory benchmarks. This hierarchical structure is particularly relevant for locomotion modeling, where joint level, segmentlevel, and fullbody dynamics interact across time scales.

Another major advance is Mamba, proposed by Gu and Dao [47], which introduces selective state space models with inputdependent state transitions. By allowing the model to dynamically propagate or forget information based on the input, Mamba achieves linear time complexity while substantially improving performance on discrete sequence tasks. Mamba matches Transformer level accuracy across language, audio, and genomics benchmarks while offering significantly higher throughput and reduced memory usage, making it an attractive backbone for long sequence and resource-constrained applications.

Subsequent works have examined the applicability of Mamba-style architectures beyond one-dimensional sequences. Yu et al. [48] show that for image classification tasks, simpler

gated convolutional structures can outperform Mamba, suggesting that selective SSMs are most beneficial in tasks characterized by long sequential dependencies. In contrast, Huang et al. [49] introduce LocalMamba, which adapts selective scanning to two dimensional data via windowed state transitions, achieving strong performance on dense prediction tasks such as object detection and semantic segmentation.

Most recently, Soydan et al. [50] propose S7, an SSM with input-dependent state transitions and stable reparameterization that further improves robustness over long horizons. S7 demonstrates strong performance across diverse domains, including event-based vision, genomics, and human activity recognition, while maintaining linear complexity and efficient training.

Table 1.5 summarizes representative state space architectures for long range sequence modeling, comparing their design principles, computational complexity, benchmark performance, and relevance to human locomotion modeling. Collectively, these models highlight the growing importance of SSMs as scalable and expressive tools for capturing long-duration dynamics in gait analysis, rehabilitation robotics, and real time lower limb control systems.

### Mixed Models

Hybrid or mixed architectures combine multiple computational paradigms—such as convolution, recurrence, attention, and state space dynamics—to leverage their complementary strengths in sequence modeling. These models are particularly attractive for complex temporal signals, where both local feature extraction and long range dependency modeling are required.

Early examples of mixed architectures include gated convolutional designs. Tan et al. [42] introduce Gated Residual Networks (GRNs) with dilated convolutions for speech enhancement, combining large receptive fields with gating mechanisms to improve information flow. By integrating residual connections, dilation, and gating, GRNs outperform both deep feedforward networks and LSTM-based models while maintaining parameter efficiency. This design paradigm is well suited to noisy temporal signals and can be transferred to robust feature extraction in locomotion and EMG-based gait analysis.

More general hybridization strategies have been proposed to address the scalability limitations of Transformers. Jaegle et al. [51] introduce the Perceiver architecture, which decouples input dimensionality from computational cost by iteratively mapping high-dimensional inputs into a fixed-size latent space using asymmetric attention. This design enables efficient processing of multimodal data, including images, audio, and point clouds, and demonstrates that attention mechanisms can be combined with latent bottlenecks to handle large sensor inputs without quadratic scaling.

Table 1.5: state space models (SSM) for long range sequence modeling.

| **Publication** | **Problem Addressed** | **Model** | **Key Techniques** | **Dataset Used** | **Complexity** | **Results** | **Advantages for Locomotion Modelling** |
|---|---|---|---|---|---|---|---|
| [44] (2022) | Efficient long sequence modeling | S4 (Structured State Space) | NPLR parametrization, HiPPO, FFT-based convolution | LRA, WikiText-103, CIFAR-10, Speech Commands | $O(n + n \log n)$ | SoTA on LRA Path-X and strong speech results | Captures long range dynamics with good efficiency; promising for gait trajectories. |
| [41] (2023) | More expressive and efficient SSM layers | H3 (Hungry Hungry Hippos) | Multiplicative SSM layers, FlashConv impl. | OpenWebText, The Pile, WikiText-103, synthetic tasks | $O(n \log n)$ | Matches Transformers on some LMs, faster inference | Good candidate where locomotion signals are very long and bandwidth-limited. |
| [45] (2023) | Simplifying and scaling S4 | S5 (Simplified State Space) | MIMO SSM layers, efficient parallel scans | LRA, Speech Commands, Pixel MNIST | $O(nH^2)$ (H latent size) | SoTA on LRA (avg. 87.4%), 98.5% on Path-X | Robust handling of long range dependencies with simpler structure. |
| [46] (2024) | Hierarchical modelling of long sensor sequences | HiSS (Hierarchical SSM) | Stacked SSMs, temporal chunking, multiscale hierarchy | CSP-Bench (ReSkin, VECtor, TotalCapture, etc.) | $O(nk)$ | 23% median MSE improvement vs. flat SSMs | Particularly relevant for hierarchical locomotion tasks (joint–segment–full-body). |
| [47] (2024) | Linear-time SSMs for long sequences | Mamba (Selective SSM) | Selective SSMs, hardware aware scan, linear recurrence | The Pile, LRA, audio, genomics | $O(n)$ | Transformer-level quality with much better efficiency | Very appealing for million-timestep locomotion data and onboard compute. |
| [48] (2024) | Do we need SSMs for vision tasks? | MambaOut | Gated CNN blocks, SSM-free token mixing | ImageNet, COCO, ADE20K | $O(n)$ | Matches/surpasses visual Mamba on images | Suggests simpler CNN-style token mixing may suffice for some locomotion signals. |
| [49] (2024) | Local/global dependency modeling in vision | LocalMamba | Windowed selective scanning, spatial–channel attention | ImageNet, COCO, ADE20K | $O(n)$ | SOTA on some segmentation tasks, +3.1% over Vim on ImageNet | Windowed scanning transferable to local joint windows plus global gait trends. |
| [50] (2024) | Efficient SSMs with input-dependent transitions | S7 | Input-dependent state transitions, stable parametrization, gating | LRA, neuromorphic data, genomics | $O(n)$ | Outperforms S4, S5, Mamba on some long range tasks | Strong candidate where locomotion dynamics vary strongly with context (terrain, speed). |

state space models have also been integrated with gating and attention mechanisms to improve efficiency and expressiveness. Mehta et al. [52] propose Gated State Spaces (GSS), which augment structured state space layers with gating mechanisms to improve long range dependency modeling while reducing training cost. By combining GSS layers with Transformer blocks, the model benefits from both efficient long range propagation and strong short-range modeling, achieving competitive performance on large scale language benchmarks with significantly improved throughput.

Another prominent hybrid approach is Hyena, introduced by Poli et al. [43], which replaces attention with implicit long convolutions and multiplicative gating. Hyena can be interpreted as a convolution–sequence mixer hybrid that achieves subquadratic complexity while matching or exceeding Transformer performance on long-context tasks. Its ability to scale efficiently to very long sequences makes it a strong candidate for modeling extended locomotion recordings without incurring quadratic attention costs.

Bridging recurrent and attention-based paradigms, Peng et al. [53] introduce RWKV, a model that combines Transformer-style parallel training with RNN-like linear-time inference through a linear attention formulation. RWKV achieves Transformer-level language modeling performance while maintaining efficient streaming inference, making it particularly attractive for online and resource-constrained applications such as real time gait monitoring and intent decoding.

More recently, Dehghani et al. [54] propose Griffin, a hybrid architecture that mixes gated linear recurrences with local attention mechanisms. Griffin balances long-term memory retention with efficient short-range context modeling, achieving competitive performance with large Transformer models while requiring significantly fewer training tokens. Its strong extrapolation capabilities and efficient inference highlight the benefits of combining recurrence and attention within a unified framework.

Hybridization strategies have also been explored in the vision domain. Hatamizadeh et al. [55] introduce MambaVision, which combines convolutional feature extraction with Mamba-style state space layers and Transformer blocks to capture both local and global visual dependencies. The resulting architecture achieves state-of-the-art performance across multiple vision benchmarks while maintaining high efficiency, demonstrating the general applicability of mixed designs across modalities.

Table 1.6 summarizes representative hybrid architectures that combine convolutional, recurrent, attention-based, and state space mechanisms. These models illustrate how mixed designs can effectively balance efficiency, expressiveness, and scalability, making them particularly well suited for complex locomotion modeling tasks that require simultaneous handling of local joint dynamics, long-term gait structure, and real time control constraints.

Table 1.6: Hybrid (mixed) architectures combining convolution, recurrence, attention, or state space mechanisms.

| **Publication** | **Problem Addressed** | **Model** | **Key Techniques** | **Dataset Used** | **Complexity** | **Results** | **Advantages for Locomotion Modelling** |
|---|---|---|---|---|---|---|---|
| [52] (2022) | Efficient autoregressive long range language modeling | Gated State Space (GSS) | Gated SSM layers, hybrid GSS–Transformer architecture, simplified initialization | PG-19, LM1B, ArXiv Math, GitHub datasets | $O(n \log n)$ | Better perplexity and throughput than DSS; good length generalization | Hybrid design suggests combining local joint dynamics with long-term gait context. |
| [43] (2023) | Addressing quadratic attention cost with conv-like blocks | Hyena (hybrid view) | Implicit long convolutions, gating, structured operators as attention alternative | WikiText-103, The Pile, PG-19, ImageNet | $O(n \log n)$ | Matches Transformers with fewer FLOPs, very fast at long contexts | Functions as a hybrid conv/sequence mixer, attractive for long locomotion signals. |
| [53] (2023) | Bridging RNN efficiency and Transformer scalability | RWKV | Linear attention formulation, RNN-like recurrence at inference, channel-wise decay | WikiText-103, The Pile, LM1B | $O(nd)$ | Comparable to Transformer performance with linear memory at inference | RNN-style streaming ideal for online gait or intent decoding in wearable robotics. |
| [54] (2024) | Efficient sequence modeling for language tasks | Griffin | Gated linear recurrences (RG-LRU), local attention, temporal–spatial feature mixing | MassiveText, MMLU, HellaSwag, PIQA, WinoGrande | $O(n)$ | Matches/surpasses Transformers with $\sim$6$\times$ fewer training tokens | Combines recurrence and local attention, promising for multi scale locomotion analysis. |

## Discussion

Human locomotion is a highly coordinated motor activity emerging from the interaction of the musculoskeletal system, motor control mechanisms, sensory feedback, and environmental dynamics. During walking, rhythmic patterns of joint kinematics, muscle activations, and ground reaction forces are continuously regulated to ensure balance, propulsion, and adaptability. Modeling human gait is inherently challenging due to its multi scale temporal structure: fine grained dynamics unfold within individual steps, while longer term dependencies span complete gait cycles and extended sequences influenced by fatigue, pathology, terrain, or assistive devices such as exoskeletons. In the context of lower limb robotics and rehabilitation, capturing these dependencies is critical for accurate trajectory prediction, gait phase detection, user intent decoding, and the design of adaptive and safe human–robot interaction strategies.

The literature reviewed in this chapter reveals a diverse landscape of neural architectures designed for sequential modeling, each offering distinct inductive biases and computational trade offs. Attention-based models, including Transformers and their long sequence variants, excel at capturing global temporal context and long range dependencies, making them well suited for scenarios in which gait structure extends across multiple cycles. However, their computational cost can be prohibitive for real time or embedded rehabilitation systems,

motivating the exploration of more efficient alternatives such as State Space Models.

Convolutional architectures remain a cornerstone of locomotion analysis due to their ability to efficiently capture local temporal and spatial patterns, such as joint angle transitions, EMG bursts, and sensor specific signatures. Temporal CNNs provide strong robustness to noise and favorable computational efficiency, making them particularly attractive for wearable and embedded systems. Extensions such as depthwise separable convolutions further enhance representational capacity while maintaining low computational overhead.

Recurrent neural networks, and in particular Long ShortTerm Memory (LSTM) models, have historically dominated human motion modeling because of their natural suitability for sequential data. By maintaining internal memory over time, RNNs and LSTMs effectively capture temporal structure within and across gait cycles. While classical RNNs offer lightweight and interpretable baselines, LSTMs remain strong performers for nonlinear and moderately long temporal dependencies typical of rehabilitation scenarios, despite limitations in scalability to extremely long sequences.

More recently, hybrid architectures have been proposed to combine the strengths of multiple paradigms. CNN–LSTM and LSTM–CNN models integrate local feature extraction with recurrent temporal memory, enabling efficient modeling of both short term gait events and longer term dynamics. Similarly, hybrid designs combining convolution, attention, or recurrence seek to balance computational efficiency with global reasoning capability.

Within this broader methodological context, the architectural choices adopted in this thesis are guided by a principled comparison across these major families. Specifically, a classical CNN is employed to model local spatiotemporal features of gait signals, while a vanilla RNN and an LSTM are included as established baselines for sequential modeling. Hybrid CNN–LSTM and LSTM–CNN architectures are evaluated to assess whether combining local convolutions with recurrent memory improves performance on multi joint locomotion data. A vanilla Transformer is considered to examine the benefits of attention based global context integration. Xception is introduced as an efficient, high capacity convolutional architecture suitable for embedded deployment. Finally, Mamba—a modern State Space Model specifically designed for long sequence efficiency and stability is selected to evaluate state-of-the-art performance on extended locomotion sequences spanning multiple gait cycles.

Together, these models span the full spectrum of contemporary sequence modeling paradigms, including convolutional, recurrent, attention-based, hybrid, and state space approaches. Evaluating them within a unified experimental framework enables a systematic analysis of how different architectural principles capture themulti scale temporal structure inherent in human gait.

This exhaustive literature review therefore serves not only as a survey of existing methods,

but as the conceptual and methodological foundation for the experimental study presented in Chapter 3. Building on the insights derived from this discussion, Chapter 3 introduces the selected architectures, datasets, and evaluation protocols, and provides a detailed empirical comparison of their effectiveness for human locomotion analysis in the context of lower limb rehabilitation robotics.

### 1.1.2 Self-Supervised Learning vs. Self-PreTraining in the Medical Domain

#### Preliminaries

This section clarifies the distinction between self-supervised learning (SSL) and Self-PreTraining (SPT), two paradigms that are often used interchangeably despite referring to different training settings. As illustrated in Fig. 4.1, SSL typically denoteslarge scale pre-training performed on an external dataset that differs from the dataset used in the downstream fine tuning stage. In contrast, Self-PreTraining refers to a setting in which the model is pre-trained directly on the same dataset that will later be used for fine tuning, as discussed by Amos et al. [56] and further analyzed in the context of rehabilitation robotics by Coser et al. [57].

More specifically, SSL leverages out-of-domain unlabeled data to learn broadly transferable representations that can generalize across tasks and domains. Conversely, SPT focuses on in domain pretraining, aiming to improve optimization dynamics and representation quality specifically for the target task and dataset. This distinction is particularly relevant in scenarios where large external datasets are unavailable or when domain specific structure plays a critical role, such as in human locomotion analysis and rehabilitation robotics.

#### Introduction to Self-PreTraining

The importance of pre-training strategies in modern deep learning was firmly established with the introduction oflarge scale language models, which demonstrated that representation learning prior to task specific fine tuning can dramatically improve downstream performance. Early work in this direction includes BERT, introduced by Devlin et al. [58], which popularized bidirectional pre-training through masked language modeling and next sentence prediction, achieving state-of-the-art results across a wide range of NLP tasks. Subsequent studies showed that pre-training performance depends not only on model architecture, but also critically on training protocols and data scale.

A notable example is RoBERTa, proposed by Liu et al. [59], which demonstrated that

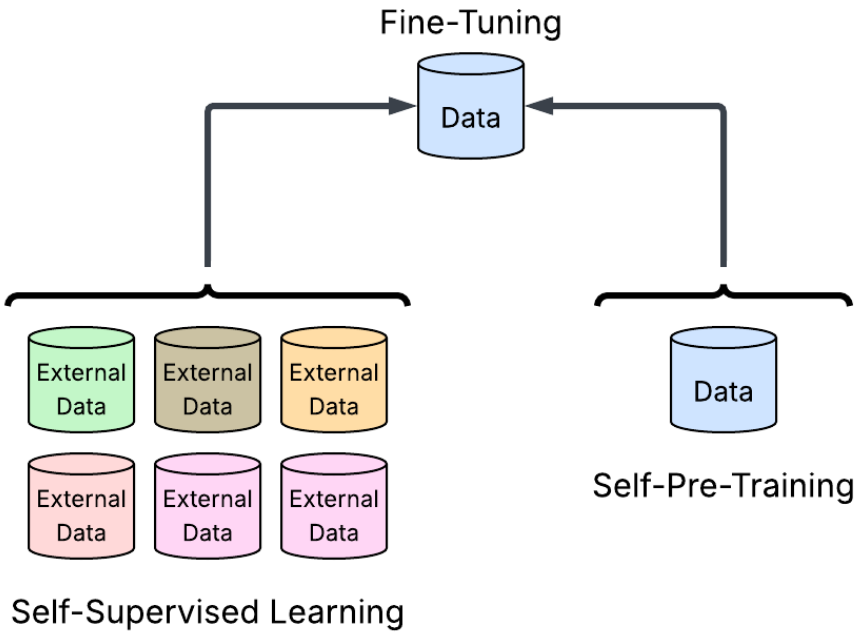


Figure 1.2: Illustration of SPT [56], compared to SSPT approaches.

longer training, larger batches, dynamic masking, and the removal of the next sentence prediction objective lead to substantial performance gains over BERT. In parallel, research began to highlight the benefits of domain adaptive pre-training. Chakrabarty et al. [60] showed that fine tuning language models on large, task-relevant corpora—such as opinionated Reddit data—significantly improves claim detection, underscoring the value of in-domain pre-training signals.

Alternative pre-training paradigms were also explored to address limitations of masked modeling. XLNet, introduced by Yang et al. [61], proposed a permutation-based autoregressive objective that overcomes BERT's independence assumptions, achieving superior performance across numerous NLP benchmarks. Similarly, Liu et al. [62] demonstrated that document-level pre-training using BERT-based encoders can significantly improve extractive and abstractive summarization through task-aware fine tuning strategies.

The scaling of pre-trained models further reshaped the field with the introduction of GPT-3 by Brown et al. [63], which revealed strong zeroshot and fewshot capabilities emerging from sheer scale. However, subsequent analyses questioned whether scale alone is responsible for pre-training gains. Chiang and Lee [64] showed that pre-training on non linguistic or synthetic data can outperform training from scratch, suggesting that optimization and representation reuse play a central role independent of semantic content.

Efficiency oriented pre-training objectives were also proposed. Clark et al. [65] introduced ELECTRA, which replaces masked language modeling with replaced token detection, achieving competitive performance with significantly reduced computational cost. In parallel, Gururangan et al. [66] systematically investigated domain adaptive (DAPT) and task adaptive pre-training (TAPT), demonstrating consistent gains—especially in low resource settings—and highlighting the benefits of multi phase pre-training strategies. Complementary advances include PEGASUS [67], which introduced gap sentence generation for abstractive summarization, achieving state-of-the-art performance across multiple datasets.

Beyond NLP, theoretical and empirical studies further clarified why pre-training works. Du et al. [68] provided guarantees showing that shared representations reduce sample complexity in few shot learning, while El-Nouby et al. [69] demonstrated that large scale datasets are not strictly necessary for effective self-supervised pre-training. Krishna et al. [70] reinforced this view by showing that pre-training on synthetic or nonsensical data yields downstream performance comparable to real data pre-training, suggesting that initialization and optimization effects dominate knowledge transfer in many settings.

Concerns related to bias, efficiency, and scaling were addressed in subsequent work. Choenni et al. [71] showed how fine tuning rapidly shifts representational biases, while Gira et al. [72] demonstrated that updating a small fraction of parameters is sufficient to mitigate bias without degrading performance. Hoffmann et al. [73] formalized scaling laws showing that many large models are under trained, leading to the development of Chinchilla, which achieves superior performance with fewer parameters but more data. Additional studies explored frozen pre-trained transformers as universal computation engines [74] and synthetic to real transfer through principled scaling laws [75].

More recently, systematic analyses such as Pythia [76] provided controlled insights into scaling, memorization, and bias by releasing families of models trained under identical conditions. Crucially for this work, Krishna et al. [70] demonstrated that self-pretraining—where models are pre-trained directly on downstream task data—can match or outperform traditional large corpus pre-training across a wide range of classification benchmarks. These findings motivate Self-PreTraining as a viable and efficient alternative tolarge scale self-supervised learning, particularly in domains where external data is scarce or domain mismatch is severe.

### Self-Supervised and Self-PreTraining for Medical Time Series

Early progress in self-supervised and self-pretraining approaches for time series analysis was marked by the work of Shi et al. [77], who proposed a self-supervised learning pipeline specifically tailored for time series classification. Their framework combined denoising-based

reconstruction with a Dynamic Time Warping (DTW)based discriminative objective, enabling the model to capture invariant temporal patterns from large unlabeled datasets such as UCR2015. This study provided early evidence that meaningful and transferable temporal representations can be learned without explicit task specific supervision.

Building on this direction, Dong et al. [78] introduced TimeSiam, a Siamese self-supervised framework in which an encoder is trained to reconstruct masked subsequences using contextual temporal information. By coupling masked modeling with temporal contrastive learning, TimeSiam achieved state-of-the-art performance across a range of forecasting and classification benchmarks, demonstrating the effectiveness of contextual reconstruction for modeling temporal dependencies. In a related vein, Zhang et al. [79] proposed a timefrequency consistency objective, enforcing representation alignment across temporal and spectral domains through contrastive learning. Their approach exhibited strong cross domain generalization, highlighting the importance of jointly modeling time domain and frequency domain characteristics in time series representations.

As the number of proposed methods has rapidly increased, several comprehensive reviews have emerged to systematize the field. Zhang et al. [80] organized self-supervised learning approaches for time series into a unified taxonomy encompassing generative, contrastive, and adversarial paradigms, while also identifying persistent challenges such as seasonality, multivariate augmentation design, and non stationary dynamics. Focusing specifically on healthcare applications, Liu et al. [81] surveyed 43 studies applying self-supervised and self-pretraining techniques to medical time series data, analyzing encoder architectures, data augmentation strategies, and loss formulations used in tasks including ICU mortality prediction, arrhythmia detection, and sleep staging.

More recently, Ma et al. [82] provided an overview of large pre-trained models for time series analysis, emphasizing the growing influence of Transformer-based architectures and masked prediction objectives inspired by advances in natural language processing. Collectively, these surveys underscore both the promise of self-supervised and self-pretraining approaches for medical time series and the need for continued investigation into models capable of capturing complex, multi scale temporal patterns under realistic data constraints.

### Pre-Training in Clinical and Multimodal Contexts

In clinical settings, the scarcity of annotated data is particularly acute, motivating the development of pre-training strategies capable of exploiting multimodal and irregularly sampled medical data. A representative example is the multimodal pre-training framework proposed by King et al. [83], which jointly aligns intensive care unit (ICU) time series signals with unstructured clinical text through a combination of contrastive learning and masked to-

ken prediction. By learning shared representations across modalities, the resulting encoders transfer effectively to downstream clinical tasks such as mortality prediction and patient phenotyping.

Closely related approaches have further explored multimodal contrastive objectives tailored to heterogeneous clinical signals. Raghu et al. [84] introduced a contrastive pre-training strategy that integrates high frequency electrocardiogram (ECG) waveforms with lower frequency laboratory measurements and vital signs. Their method improves early detection of elevated mean pulmonary arterial pressure (mPAP) and mortality risk, demonstrating the benefits of jointly modeling modalities with distinct temporal resolutions.

Beyond multimodal alignment, a growing body of work focuses on explicitly modeling the irregular sampling patterns and missingness inherent to clinical time series. Xu et al. [85] proposed TransEHR, a self-supervised Transformer-based architecture designed to handle asynchronous events and heterogeneously sampled measurements. By combining masked value prediction, replaced token detection, and event forecasting objectives, TransEHR learns representations that are robust to irregular temporal structure and missing data.

Addressing similar challenges from a contrastive perspective, Chowdhury et al. [86] introduced PrimeNet, which incorporates time aware contrastive learning and duration-based masking to better encode uneven temporal intervals in multivariate clinical time series. Earlier work by Malhotra et al. [87] laid important foundations in this area through TimeNet, which employs reconstruction-based objectives to learn transferable representations from irregular clinical and sensor data, demonstrating strong few shot and cross domain generalization.

Collectively, these approaches highlight the necessity of pre-training strategies that explicitly account for multimodality, irregular sampling, and temporal heterogeneity. Such design principles are especially critical in clinical and rehabilitation contexts, where physiological signals, clinical events, and sensor measurements interact across multiple time scales and data modalities.

### Scaling Pre-Training Across Domains

Beyond the healthcare domain, recent research has investigated whether large pre-trained time series models can generalize across heterogeneous application areas. Prabhakar et al. [88] introduced the Large Pre-Trained time series Model (LPTM), which employs adaptive segmentation strategies during masked pre-training to accommodate varying sampling rates and structural characteristics across diverse datasets, including epidemiological records, energy consumption, traffic flows, and behavioral sensor data. Their results demonstrate that adaptive segmentation is a key enabler for learning transferable representations across

domains with heterogeneous temporal properties.

Complementary evidence is provided by Dong et al. [89], who proposed SimMTM, a framework combining series wise contrastive alignment with masked temporal modeling. By jointly optimizing these objectives, SimMTM achieves consistent performance gains across forecasting, classification, and imputation tasks, highlighting the benefits of integrating contrastive and reconstruction-based pre-training strategies for general purpose time series representation learning.

Extending the notion of universality further, Zhang et al. [90] introduced UniMTS, a multimodal framework that aligns motion time series data with corresponding semantic text descriptions. This cross modal association enables strong zero shot and few shot generalization, allowing the model to transfer effectively to unseen tasks and domains without extensive retraining. UniMTS demonstrates that linking temporal signals with high level semantic information can significantly enhance the robustness and adaptability of learned representations.

Collectively, these studies indicate thatlarge scale pre-training on heterogeneous time series corpora can produce robust and transferable backbone models. Such models are capable of supporting a wide range of downstream tasks across domains, reinforcing the potential of scalable self-supervised and self-pretraining strategies beyond domain specific settings.

### Insights from Other Modalities

Several methodological advances in self-supervised and self-pretraining strategies for time series analysis have been strongly influenced by progress in neighboring domains such as computer vision and speech processing. In the vision community, masked autoencoder (MAE) approaches have demonstrated remarkable effectiveness, particularly in medical imaging contexts where labeled data is scarce and distribution shift is common. Studies by Zhou et al. [91], Li et al. [92], Tu et al. [93], Das et al. [94], and Röhrich et al. [95] show that self-pretraining on target domain medical images using masked reconstruction objectives can substantially improve downstream segmentation and classification performance. These results highlight how in domain self-pretraining mitigates domain mismatch and enhances representation robustness.

In speech processing, analogous self-supervised paradigms have driven major advances in representation learning. Baevski et al. [96] introduced wav2vec-based learning, which combines contrastive objectives with latent signal modeling to learn rich speech representations from raw audio. Building on this line of work, emotion2vec proposed by Ma et al. [97] demonstrates howlarge scale self-supervised pretraining can capture high-level affective and paralinguistic features with minimal labeled supervision. These frameworks illustrate how

contrastive and generative objectives can be effectively combined to model complex temporal structure in continuous signals.

More broadly, multimodal and hybrid training strategies provide additional insights relevant to time series pretraining. Mu et al. [98] introduced SLIP, a framework that jointly leverages self-supervised and supervised objectives to improve visual representation learning, while Zoph et al. [99] showed that self-training and pseudo-labeling can significantly enhance transferability across tasks. Complementary perspectives are provided by surveys on sequential transfer learning [100] and empirical analyses of dataset scale requirements [69], which emphasize that effective objectives and inductive biases often matter more than sheer dataset size.

Taken together, these cross-domain findings have strongly informed the design of modern self-supervised and self-pretraining strategies for time series data. By borrowing architectural principles, training objectives, and inductive biases from vision, speech, and multimodal learning, recent approaches to time series pretraining have achieved improved robustness, efficiency, and generalization, even in settings characterized by limited labeled data and strong domain specificity.

### Discussion

Across the evolution of self-supervised learning (SSL), domain-adaptive pretraining, andlarge scale foundation models, a unifying trend has emerged: the effectiveness of pretraining increasingly depends on the alignment between the pretraining distribution, the inductive biases of the model, and the structure of the downstream task. While early breakthroughs in natural language processing and computer vision were driven by massive upstream pretraining on generic corpora, growing empirical evidence indicates that such large scale external datasets are not always necessary—and, in some cases, may even be suboptimal. Instead, pretraining on indomain or downstream data itself can provide comparable or superior improvements at substantially lower computational, privacy, and deployment costs. This body of work positions *Self-PreTraining* (SPT) as a principled alternative to traditional SSL, particularly in specialized domains characterized by limited labeled data, strong distribution shifts, and strict datagovernance constraints.

A central insight emerging from the surveyed literature is that the benefits of pretraining arise not only from learning semantic or domaingeneral knowledge, but also—and perhaps more fundamentally—from improved optimization dynamics and favorable parameter initialization. The observation that models pretrained on synthetic or nonsensical data can rival those pretrained on real text or images suggests that the structure of the pretraining objective (e.g., masked prediction, contrastive alignment, sequence reconstruction) plays a

dominant role in shaping the optimization landscape. From this perspective, SPT should not be viewed merely as a pragmatic workaround for data scarcity, but rather as a coherent training paradigm that exploits self-supervised objectives to improve stability, convergence, and sample efficiency during fine tuning.

These insights are particularly salient in the context of medical time series. Unlike text or images—where upstream and downstream distributions often share structural similarities—clinical time series are governed by highly domains pecific acquisition processes, irregular sampling patterns, event driven dynamics, and multimodal interactions shaped by institutional protocols and patient physiology. External upstream datasets rarely capture these characteristics, limiting the effectiveness of generic SSL approaches. In contrast, studies such as TransEHR, PrimeNet, TimeSiam, and multimodal contrastive frameworks demonstrate that effective representation learning in healthcare requires models to internalize the precise temporal structure, correlations, and missingness patterns present in the target data. SPT naturally supports this requirement by enabling pretraining directly on the downstream distribution.

From a practical standpoint, the constraints of clinical machine learning—limited annotated datasets, high labeling costs, privacy regulations, and institutional heterogeneity—render large scale upstream SSL difficult to deploy. SPT offers an appealing balance between performance and feasibility: it avoids reliance on external corpora, mitigates distribution shift, respects privacy boundaries, and supports local specialization. Models trained through SPT can adapt to site specific patterns that generic foundation models may overlook, which is critical for real world deployment in healthcare and rehabilitation settings.

Cross modal evidence further reinforces the relevance of SPT. In medical imaging, self-pretraining on target domain data has been shown to mitigate distribution shift and improve segmentation and classification performance. In speech and audio processing, contrastive and masked prediction objectives achieve strong results even when trained on relatively modest corpora. These findings suggest that the mechanisms underpinning the success of SPT—domain alignment, inductive bias shaping, and optimization stability—generalize across modalities and are not specific to any single data type.

Beyond empirical performance, recent work on scaling laws and dataset–model trade offs provides additional theoretical grounding for SPT. Studies such as Chinchilla demonstrate that optimal performance depends on balancing model capacity with dataset size. In domains like healthcare, where large quantities of unlabeled but domain specific data is available, SPT offers a compute efficient strategy to exploit this data without resorting to massive, misaligned upstream corpora. Rather than allocating resources to pretraining on generic datasets with limited downstream relevance, SPT focuses computation on learning precisely

those structures that matter at inference time.

Importantly, the methodological developments presented in this thesis directly build upon these insights. In particular, the two works introduced in the preceding sections were explicitly designed under the Self-PreTraining paradigm. Both approaches operationalize SPT by pretraining models exclusively on the same medical and locomotion datasets used for downstream evaluation, thereby eliminating distribution mismatch and enabling the models to internalize domain specific temporal dynamics from the outset. These contributions move beyond conceptual arguments for SPT by providing concrete architectures, training protocols, and empirical evidence demonstrating its effectiveness in realistic healthcare and rehabilitation scenarios.

Taken together, the literature across NLP, vision, speech, and time series research converges on three key conclusions. First, pretraining objectives and optimization dynamics are at least as important as corpus scale, making SPT a high impact and low cost strategy for representation learning. Second, domain alignment is a critical determinant of downstream performance, and for medical time series—where upstream data rarely match target distributions—SPT offers a uniquely well suited alternative. Third, the structural and regulatory constraints of healthcare favor training paradigms that are local, privacy aware, and data efficient, all of which are naturally supported by SPT.

In summary, Self-PreTraining should not be regarded as a secondary option when external data is unavailable, but as a principled and empirically supported paradigm with distinct theoretical and practical advantages for medical time series analysis. By eliminating distribution shift, capturing domain specific temporal structure, improving optimization stability, and enabling privacy conscious deployment, SPT emerges as a powerful foundation for next generation clinical machine learning. This exhaustive literature review therefore serves not only as a survey of existing methods, but as the conceptual and methodological basis for the experimental investigations presented in Chapter 4. Building on the insights synthesized here, Chapter 5 introduces the proposed models, datasets, and evaluation protocols, and provides a detailed empirical assessment of Self-PreTraining for improve diagnosis and human locomotion analysis in the context healthcare.

## 1.2 Research Questions and Objectives

The work conducted throughout this doctoral thesis is driven by a central scientific and translational motivation: to strengthen the methodological foundations of artificial intelligence in rehabilitation robotics and medical time–series analysis, with the ultimate goal of improving the field of human locomotion anlysis and diagnosis.

Lower limb rehabilitation robotics and medical monitoring systems increasingly rely on data-driven models to interpret complex physiological signals. However, several fundamental challenges remain unresolved: modeling multi-scale temporal dependencies in human gait, ensuring robustness across heterogeneous patients, operating under real-time and embedded constraints, and learning effectively from limited labeled clinical data. At the same time, modern deep learning architectures have rapidly evolved, introducing novel sequence modeling paradigms whose applicability to rehabilitation and healthcare settings remains only partially understood.

This thesis addresses these challenges through two complementary research directions. The first focuses on identifying which architectural principles are most suitable for modeling human locomotion in rehabilitation robotics. The second investigates whether recent advances in representation learning specifically Self-PreTraining can improve data efficiency and generalization in medical time series analysis.

Although each phase of the research examines a different component of this broader objective ranging from a systematic study of AI techniques in lower limb rehabilitation robotics, to benchmarking deep learning architectures for human locomotion recognition, to analyzing the theoretical properties of Self-PreTraining, and finally evaluating its applicability to real-world healthcare datasets—these efforts are unified by a coherent set of research questions.

This section introduces the core research questions that structure the thesis. They emerged naturally from the two main pillars of the work:

- **Research Question 1 (AI for lower limb Rehabilitation Robotics):** What artificial intelligence approaches have been developed for lower limb rehabilitation robotics, and how can existing deep learning architectures and sensor configurations be advanced to provide a novel and more accurate understanding of human locomotion?

- **Research Question 2 (Self-PreTraining for Medical time series):** How can Self-PreTraining be exploited to improve transformer-based models for diagnosis and rehabilitation robotics, and to what extent does this approach generalize across heterogeneous medical time series data?

Together, these questions reflect both the conceptual trajectory of the thesis and its practical motivations: to bridge theoretical advances in sequence modeling with operational requirements in diagnosis and rehabilitation robotics. The following subsections detail each research question and clarify the scientific rationale behind them.

### 1.2.1 Research Question 1: AI for lower limb Rehabilitation Robotics

**Performing and exhaustive Literature review of AI methods for Lower limb robotics**

An essential first step in broadening the research trajectory has been to develop a solid understanding of rehabilitation robotics and the artificial intelligence methodologies applied within this field. To facilitate this transition, an extensive review of the existing scientific literature was conducted, focusing on both the technical foundations and the latest advancements in AI-driven rehabilitation systems. The insights gathered from this work form the basis of the literature review presented in 2, which aims to contextualize the state of the art and identify the key opportunities and challenges that motivate the research directions pursued in this thesis.

**Identifying the Best performing Architecture for human locomotion recognition with the minimal sensor set up**

Building upon the findings of the literature review and considering the characteristics and availability of suitable datasets, we elected to direct our research toward the domain of human locomotion recognition. This area represents a crucial component in the development of intelligent control strategies for rehabilitation and assistive robotic systems, particularly lower limb exoskeletons. Our primary objective was to investigate which state-of-the-art neural network architectures for time series analysis demonstrate the most promising performance for accurately identifying and classifying human gait patterns.

In parallel, we sought to determine the most effective sensor configurations for this application, with a focus on setups that are both practically deployable and compatible with real world exoskeleton platforms. This involved examining a range of sensing modalities—such as inertial measurement units and electromyography—and evaluating their relevance, reliability, and integration potential within lower limb robotic systems.

Through this dual approach, we aimed not only to benchmark advanced deep learning models in the context of locomotion recognition but also to outline an optimal sensing strategy that could guide future implementations in rehabilitation robotics. These considerations collectively shaped the research direction presented in the 3.

### 1.2.2 Research Question 2: Self-PreTraining for Diagnosis and Rehabilitation Robotics

#### Uderstending why Self-PreTraining allowed transformer to solve the Long Range Arena Tasks

Following the completion of my initial experimental work, it became clear that strengthening my theoretical foundation in deep learning would be essential for advancing my research. For this reason, during my research stay at the Max Planck Institute for Intelligent Systems, I undertook a project focused more explicitly on core AI methodologies. This work centered on the study of Self-PreTraining, a recently proposed training paradigm designed to enhance the performance and generalization capabilities of modern neural network architectures.

In particular, my investigation concentrated on understanding why Self-PreTraining exhibits such strong performance when applied to transformer models tasked with solving the benchmark problems of the Long Range Arena (LRA). These tasks are specifically designed to stress test a model's ability to capture long term dependencies, process sequences of substantial length, and maintain stable learning dynamics under challenging conditions—areas where transformers typically face computational and architectural constraints.

This body of work significantly expanded my understanding of deep learning principles and provided a solid theoretical groundwork that complements the more application oriented research conducted in the rest of the thesis. Evrything is explained in 4.

#### Understanding if Self-PreTraining improve diagnosis and rehabilitation robotics

In the final stage of my PhD, I aimed to assess whether the Self-PreTraining methodology I had studied during my research stay abroad could demonstrate practical value beyond controlled benchmarks, particularly when applied to real world medical time series data. To evaluate its generality and effectiveness, I conducted a comprehensive experimental study across three distinct application domains: rehabilitation robotics, stress detection, and Parkinson's disease monitoring. Each domain required the analysis of different types of physiological or biomechanical signals, thereby providing a robust and heterogeneous validation framework.

For rehabilitation robotics, the models were tested on datasets composed of inertial measurement unit (IMU) signals, capturing angular velocities, accelerations, and limb orientations during human locomotion. Such data is highly relevant for lower limb exoskeletons and wearable assistive devices, as they encode gait dynamics, transitions between locomotor states, and subtle variations in movement patterns. The goal was to determine whether Self-PreTraining could enhance gait phase recognition, improve intention decoding, or sup-

port more accurate locomotion classification by leveraging its ability to learn meaningful representations from raw sensor sequences.

In the domain of stress detection, the experiments were performed using accelerometry and heart rate derived signals (such as inter beat intervals and heart rate variability), which reflect both physical activity and autonomic nervous system dynamics. Stress related datasets often contain noisy, non stationary, and context dependent patterns, making them challenging for traditional supervised learning methods. By testing Self-PreTraining on this type of physiological data, the objective was to evaluate its robustness when dealing with high intra subject variability and its potential to capture subtle temporal signatures associated with psychological stress.

Finally, for Parkinson's disease detection, the method was evaluated on datasets based on foot force measurements, which capture pressure distributions and temporal gait asymmetries—key indicators of motor impairment in Parkinsonian gait. foot force signals provide fine-grained temporal information about step timing, weight-shifting patterns, and tremor manifestations. Because these signals exhibit long range dependencies and subtle temporal irregularities, they represent an ideal testbed for assessing whether Self-PreTraining can improve classification or severity estimation in early-stage motor impairment detection.

By systematically applying Self-PreTraining across these three domains—each characterized by distinct sensor modalities (IMU data, accelerometry, heart rate dynamics, and foot force measurements)—this work aimed to determine whether its theoretical advantages translate into consistent empirical improvements. This investigation not only assessed the method's generalizability across heterogeneous medical time series tasks but also contributed to understanding its potential role in future clinically oriented machine-learning systems. More in detail the work has been explained in 5.

## 1.3 Thesis Organization

This thesis is structured into six chapters (including the one already presented).

- Chapter 2 provides a comprehensive review of AI-based methodologies for exoskeleton assisted lower limb rehabilitation. It surveys the current literature, analyses existing AI approaches, and discusses open challenges and limitations in the field.
- Chapter 3 presents a comparative study of deep learning techniques for human locomotion analysis in lower limb exoskeletons. This chapter describes the materials, methods, experimental setup, and results, followed by a detailed discussion of the findings.

- Chapter 4 focuses on understanding Self-PreTraining for sequence classification. It investigates the underlying principles of the method and evaluates its effectiveness through experimental analysis.
- Chapter 5 examines whether Self-PreTraining generalises to medical time series data. Using transformer-based models, this chapter assesses the usefulness of Self-PreTraining for diagnosis and rehabilitation-related applications.
- Chapter 6 summarises the main contributions of the thesis, discusses limitations, and outlines directions for future research.

## 1.4 List of Pubblications

Some of the contributions in this thesis appear in the following pubblications and or are submitted:

- **Coser et al.**, AI-based methodologies for exoskeleton-assisted rehabilitation of the lower limb: a review, Journal: Frontiers in Robotics and AI
- **Coser et al.**, Deep learning for human locomotion analysis in lower-limb exoskeletons: a comparative study, Journal: Frontiers in Computer Science
- **Coser et al.** Toward Understanding Self-PreTraining for Sequence Classification (Preliminary Version). ICML 2025 Workshop on Long-Context Foundation Models
- **Coser et al.** Tower Understanding Self-PreTraining for Sequence Classification (Full Version). Under review at full conference track
- **Coser et al.** Is Self-PreTraining useful for Human Locomotion Identification and Medical Time Series Diagnosis?. In preparation.

# Chapter 2

# AI-based methodologies for exoskeleton assisted rehabilitation of the lower limb: a review

**Publication Statement**

Parts of this chapter are based on the following peer-reviewed publication: "AI-based methodologies for exoskeleton-assisted rehabilitation of the lower limb: a review", Authors: Coser et al., Journal: Frontiers in Robotics and AI.

Lower limb rehabilitation is a field of great clinical relevance, dealing with the rehabilitation of individuals with motor disabilities in the lower limbs as a result of trauma, neurological pathologies, or musculoskeletal injuries [7, 101]. Traditionally, lower limb rehabilitation has been conducted by human therapists through physical therapies and specific exercises. However, the integration of robotic technologies and Artificial Intelligence (AI) has paved the way for the design of new tools and approaches to improve the quality of therapies and increase patients' independence and mobility [102, 103, 104].

Among all the devices purposely designed to assist patients' lower limbs, exoskeletons are wearable robots that closely interact with humans. Typically they are robotic exoskeleton structures that operate alongside human limbs and increase human locomotory economy, augment joint strength, and increase endurance and strength. Nowadays, exoskeleton devices are studied and employed in many application scenarios such as industry [105], space [106] and healthcare [107]. Moreover, they can also be used in industry, for tasks such as heavy lifting or repetitive motion, reducing the risk of injury to workers [108] [109]. Exoskeletons can be used for rehabilitation, allowing patients with spinal cord injuries, stroke, or other

conditions to regain mobility and improve their physical function. Currently, there is a large interest in the design of lower limb-compliant exoskeletons aimed at gait rehabilitation or assistance [110].

Exoskeletons typically comprise a frame, actuators, sensors, and control systems that enable them to mimic human movement. Batteries or other energy sources can power them, and they can be controlled by a person's actions, by a software-based controller, or by a combination of both. Lower limb exoskeletons are composed of actuators disposed on multiple joints, in particular the hips, ankles, or knees of the users. Each joint can assist different movements produced by an anatomical joint. Therefore, the exoskeletons can be classified as follows according to the joint or joints that can be assisted during the robot-assisted gait.

- Hip Exoskeletons assist the hip joint connecting upper and lower limbs, enabling a person to perform flexion/extensions, abduction/adduction, and medial/lateral rotation. These motions are necessary for a person to walk or run. Those exoskeletons have actuators placed on the users' hips to basically enable a reduction of stress on hip and ankle muscles [111].

- Most of knee exoskeletons present only one DoF, needed for the motion on the knee flexion/extensions actions. In those cases, a soft inflatable cushion is typically used as an actuator. The inflatable part is placed behind the user's knee for the reduction of weight of the exoskeleton. It uses a pneumatic system to inflate and deflate the component. The exoskeleton is inflated during the swing phase of the walking gait and deflated during the other phases of the walking gait cycle [112].

- Ankle exoskeletons have been developed to assist one degree of freedom. The ankle joint has four bones and three planar motions (three DoF). Plantar or dorsiflexion movement is a primary movement during the gait cycle that can be assisted by means of robotic devices [113].

- Multiple joints exoskeletons integrate more than one actuator to assist users during robot-assisted gait. They actuate a combination of joints implementing sophisticated control strategies, in order to face the increase in hardware complexity, to produce a harmonious gait [114].

A key role in exoskeleton development is played by artificial intelligence (AI). In particular, AI can function as a control system of the exoskeleton, indeed the device can learn and adapt to the movements and the general state of the wearer, allowing for a more intuitive responsive experience [115]. Moreover, AI can also be used to analyze data collected from

wearable sensors [116] and cameras to better understand the user's movements and provide personalized feedback and guidance.

This section aims to provide an overview of AI-based methodologies currently investigated in lower limb robot-aided rehabilitation. The most relevant scientific studies conducted in the field will be reviewed, analyzing the different applications of AI on lower limb exoskeletons.

## 2.1 Current Review on the topic

In recent years, the field of rehabilitative robotics has garnered significant attention, resulting in a multitude of published works. Existing literature includes several comprehensive reviews that explore the application of robotics in lower limb rehabilitation [117, 118, 119, 120]. However, our current review stands apart as it uniquely centers on the integration of advanced Artificial Intelligence (AI) methods in this domain, representing a novel and innovative contribution.

Khera et al. [117] delve into the role of machine learning within gait analysis. Their examination of 43 distinct studies highlighted Support Vector Machine (SVM) as the most effective classifier. Similarly, in the realm of robotic control, which closely aligns with our study's scope, the significance of reinforcement learning and deep neural networks emerges as pivotal, consistent with our findings. Another notable work by Mu et al. [118] explores the application of AI in rehabilitation, specifically focusing on objective assessment methods. Their research encompasses a range of AI-driven parameters, such as trajectory error features, joint angles, joint angular velocities, and surface Electromyography (sEMG) signal features. Crucially, they discuss the influence of data scale and feature count, laying a foundation for our emphasis on advanced AI techniques.

Prakash et al.'s comprehensive survey [119] extensively covers gait analysis across diverse domains, including clinical diagnosis, geriatric care, sports, biometrics, rehabilitation, and industrial settings. This review meticulously presents various machine learning approaches alongside corresponding datasets. While informative, it sets the stage for our innovative contribution that uniquely integrates cutting-edge AI methodologies.

Lastly, the recent work by Harris et al. [120] explores the application of AI to human gait, identifying six key areas of focus. This work, while valuable, underscores the evolving landscape and the potential for new insights, particularly where AI methodologies converge with lower limb rehabilitation.

In essence, our review's distinctive emphasis on the integration of advanced AI methods within the context of lower limb robot-aided rehabilitation sets it apart from the existing literature. By addressing this innovative approach, this review contributes to expanding

the horizon of rehabilitative robotics, providing fresh perspectives and insights that can potentially drive future advancements in this critical field.

## 2.2 AI Approaches for lower limb rehabilitation robotics

Table 2.1 presents a comprehensive overview of the research papers reviewed in this paper. Each row includes paper references, objectives, features, models, performance, subjects (H: healthy, P: pathological), robots, and different application scenarios. In particular, we include four main scenarios, which are Robot Control (RC), Locomotion Classification (LC), Intention Detection (ID), and Human Joints Trajectory Prediction (HJTP). By structuring the information in this manner, the table offers a comprehensive overview of the included papers and their content. The table is ordered with respect to methodologies and year, and it is divided into four sections as this section does. Indeed, the 40 papers reviewed here are grouped into four categories according to the AI methodology adopted, namely Reinforcement Learning (RL), Neural Networks (NN), Support Vector Machine (SVM), and Decision Tree (DT). We are aware that we should divide the approaches into reinforcement learning and supervised learning (as no papers adopt an unsupervised approach), but for the sake of readability, both in text and in the table we opted for maintaining such four groups.

Table 2.1: AI-based solutions implemented for lower limb robot-aided rehabilitation

| **Paper** | **Objective** | **Feature** | **Model** | **Perf** | **Sub** | **Robot** | **App** |
|---|---|---|---|---|---|---|---|
| *Reinforcement Learning* | | | | | | | |
| [121] | Develop Robot Control system | Human forces | RL | safety of 0.32 m | 1H, 5P | Non Exoskeleton | RC |
| [122] | Human-robot interactive control | Joint angle and human forces | RL | Err:0.2 rad | 1H | CARR robot | RC |
| [123] | Robotic Control personalization of a robotic knee prosthesis | Joint torque and gait frequency | RL | Err:0.07 rad | 1H, 1P | Prosthesis | RC |
| [124] | Compliance control of a robotic walk assist device | Joint position angle and velocity | RL | $\leq$Err:0.08 rad | Sim | Sim | RC |
| [125] | RL for robotic mirror therapy | Motion Trajectory | RL | Err: 1.05 rad | 5P (paretic leg) | Research prototype | RC |
| [126] | Robotic control for lower limb prosthesis | Hip and knee angle | RL | Err: 0.01 rad | 2H | Prosthesis | RC |
| [127] | RC of lower exoskeleton for squat assistance | Joint position, velocity and centre of pressure | RL | Err:0.05 rad | Sim | Sim | RC |
| [128] | RC for lower limb rehabilitation exoskeleton | Joint angle | RL | Err:0.1 rad | Sim | Sim | RC |
| [129] | deep-RL for control of exoskeleton gait patterns | Force and position | deep-RL | Err:0.01 rad | Sim | Sim | RC |
| [130] | deep-RL control of lower limb rehabilitation exoskeleton | Joint angle, velocity | deep-RL | Err:0.1 rad | 1H, 3P | Research prototype | RC |
| *Neural Networks* | | | | | | | |
| [131] | Gait parameter prediction | Cadence, stride length, walking speed | NN: MLP | deviation of 0.03% | 50H | Non-robot | LC |
| [132] | Waveforms planning | Anthropometric data | NN: MLP | 0.98 a. | 15H | Non-robot | HJTP |

Table 2.2: Continuation of AI-based solutions implemented for lower limb robots

| **Paper** | **Obj** | **Feature** | **Model** | **Per** | **Sub** | **Robot** | **App** |
|---|---|---|---|---|---|---|---|
| [133] | Gait phase classification | Human forces | NN: MLP | 0.98 a. | 7H | ROBIN-H1 Exoskelton | LC |
| [134] | Knee joints modelling | Hip, knee angle | NN:Long-short term memory | $\leq$Err:0.30 rad | 20H | SIAT-Exoskelton | HJTP |
| [135] | Joint angle simulation | Ankle, knee and hip angle | NN: MLP | 0.94 a. | 34H | Non-robot | HJTP |
| [136] | NN controller for Lower limb Rehabilitation Exos | Emg | NN: Radial basis function | Err:$\leq$ $1{\cdot}10^{-3}$ rad | Sim | Sim | RC |
| [137] | Joint moment prediction | Emg and joint angles | NN:Elastic | 0.96 a. | 8H | Non-robot | HJTP |
| [138] | Control of Lower Limb Rehabilitation exos | Hip knee and joint torque and speed. | NN: MLP | Err:0.90 rad | Sim | Research prototype | RC |
| [139] | Control of lower limb robot | sEMG | NN: wavelet | 0.01 m | 4H | Research-prototype | RC |
| [140] | Trajectory tracking | Hip knee and ankle angle | NN: MLP | Err:$\leq$:0.02 rad | Sim | Research-prototype | RC |
| [141] | Individual gait generation | Hip,knee,ankle position and force | NN: recurrent NN | Err:$\leq$0.16 rad | 137H | Non-Robot | HJTP |
| [142] | Evaluation of postoperative rehabilitation of patients | Angular velocity | NN: recurrent NN | 1 a. | 36H | Non-robot | LC |
| [143] | Data augmentation and trajectory prediction | Hip and knee angle | NN: GAN | 0.88 a. | 5H | Non-robot | RC |
| [144] | NN controller for lower limb robotic | Hip and knee angle | NN: MLP | Err:0.03 rad | 10H | Research-prototype | RC |
| *Support Vector Machine* | | | | | | | |
| [145] | SVM classification of locomotion using emg | Emg | SVM | 0.95 a. | 3H | Non-robot | LC |

Table 2.3: Continuation of AI-based solutions implemented for lower limb robots

| **Paper** | **Objective** | **Feature** | **Model** | **Perf** | **Sub** | **Robot** | **App** |
|---|---|---|---|---|---|---|---|
| [146] | Motion intent recognition for wearable lower device | Hip and Knee position | SVM | 0.96 a. | 1H | Research prototype | ID |
| [147] | Gait phase prediction | Hip and knee joint angles | SVM | 0.087 rad angular accuracy | 10H | Research prototype | LC |
| [148] | Gait recognition of Lower Limb rehabilitation robot | Horizontal distances | SVM | 0.95 | 3H | Non-robot | LC |
| [149] | Lower Limb rehabilitation | Emg | SVM | 0.99 a. | 8H | Research prototype | RC |
| [150] | Classification of gait phases using emg | Emg | SVM | 0.96 a. | 8H | Non-robot | LC |
| [151] | Human gait recognition | feet pressure, waist and calf movement | SVM | 0.965 a. | 5H | Non-robot | LC |
| [152] | Gait phase classification | Hip and knee angle | SVM | 0.86 a. | 10H | Research prototype | LC |
| [153] | Control strategy for lower limb rehabilitation | sEmg | SVM | 0.95 a. | 10H | SIAT Exoskeleton | ID |
| [154] | Lower limb movement recognition | sEmg | SVM | 0.907 a. | 6H | Non-robot | LC |
| *Decision Tree* | | | | | | | |
| [155] | Walking gait identification | Force sensors, knee angle, angular velocity | Decision tree | 0.99 a. | 1H | Research-prototype | LC |
| [156] | patient specific gait training | Anthropometric data | Decision Tree | Err:0.08 rad | 113H | Research-prototype | LC |

Table 2.4: Continuation of AI-based solutions implemented for lower limb robots

| Paper | Objective | Feature | Model | Perf | Sub | Robot | App |
|---|---|---|---|---|---|---|---|
| [157] | Gait Phase Recognition | Hip knee position, walk velocity | Decision tree and NN and NARX | 1, 0.987 and 0.948 a | 1H | Research-prototype | LC |
| [158] | Stroke patient identification | Age, sex and stroke type | Decision tree | 0.85 a. | 481P (stroke) | Non-robot | LC |
| [159] | Control of motion rehabilitation robot | Hip, knee and ankle position | Decision Tree | 0.997 a | 10H | Research-prototype | RC |
| [160] | Gait Deviation Correction method | CGA curves | Decision Tree | Err:$\leq$0.11 rad | 15H | Research-prototype | RC |

### Reinforcement Learning

Reinforcement learning (RL) is a category of AI used for training an intelligent agent to perform tasks or achieve goals in a specific environment by maximizing the expected cumulative reward. It has three main components: the action space, the state space, and the reward, indeed the agent moves around the state space by taking an action, and the one that maximizes the reward is the favorite[161].

The study by Xu et al. (2015) proposed an RL-based shared control (RLSC) algorithm for a walking-aid robot that aimed to address control issues in cooperative walking-aid robot systems. The authors introduced an intelligent walking-aid robot and a human walking intention estimation algorithm to assist elderly and disabled people with limited physical and cognitive abilities in maintaining a sense of stability and comfort. The RLSC algorithm enables the robot to autonomously adapt to different user operation habits and motor abilities by dynamically adjusting user control weight based on different user control efficiencies and walking environments. The robot collects state (distance between user and angular velocity), so the final RL system in total presents 1250 states. Then to balance exploration and exploitation the action with the highest estimated value is selected. Moreover, the study verified the effectiveness of the proposed RLSC algorithm through experiments conducted on 6 subjects, 5 healthy and 1 pathological. The results showed that the algorithm could significantly improve the degree of comfort when using the device and automatically adapt to the user's behavior. Furthermore, the study demonstrated the potential of RL-based shared control algorithms for enhancing the performance and usability of cooperative walking-aid robot systems [121]. The safety metrics, computed as the minimum distance between the

robot and the obstacles, achieved 0.32 m in the worst tested condition.

In [122], the authors propose a method for controlling a lower limb undergoing training to use the exoskeleton. The goal is to provide assistance to patients in a way that is safe, effective, and tailored to the individual patient's needs. The exoskeleton robot assists both lower limbs, each with two rotational degrees of freedom. The robot is driven by a pneumatic proportional servo system, which allows for precise control of the robot's movements. To enable effective human-robot interaction, the authors adopt an adaptive admittance model. Admittance control is a method used to regulate the interaction between a robot and its environment, i.e., the patient. The admittance model used in this study is adaptive so that it can adjust its parameters based on the patient's needs and performance. The authors design an adaptive law for the admittance parameters using a sigmoid function and an RL algorithm. The agent is the exoskeleton robot, and the environment is the patient's body. By using RL to optimize the admittance parameters, the authors are able to obtain individualized parameters that are suitable for each patient's specific needs. Joint angle error and human-robot contact force are selected to form the state space, the action is selected among four possibilities that correspond to up, down, left, and right movements. The $\epsilon$-greedy policies are applied to the on-policy method, meaning that with a probability of 0.5 the maximal estimated action value is chosen. The experiment was conducted to verify the feasibility of the interactive control system with the hip and knee joint angle data in the Clinical Gait Analysis (CGA) database as reference trajectories and is conducted on the prototype of a gait rehabilitation training robot. The database is established by capturing a large number of motion information of normal people while walking by the 3-D Motion Capture System of Northern Digital Technologies Inc. The system has been tested on 1 healthy subject with an error on 0.2 rad [122].

The paper [123] presents an approximate dynamic programming (ADP) approach to automatically tune the parameters of a robotic knee prosthesis to meet the individual needs of human users. The goal is to achieve target gait kinematics for each user. The authors tested their ADP approach on two subjects, one able-bodied and one amputee, both walking at a fixed speed on a treadmill, with an error of 0.07 rad. The ADP-tuner was able to learn to reach target gait kinematics in an average of 300 gait cycles or 10 minutes of walking. In particular, the algorithm learns efficiently through interaction with the human-prosthesis system in real time, without any prior tuning knowledge from either a trained clinician or a field expert. The authors also observed improved tuning performance when they transferred a previously learned ADP controller to a new learning session with the same subject. This is the first implementation of online ADP learning control for a clinical problem involving human subjects, making it a significant contribution to personalized robotic prostheses [123].

The proposed scheme in [124] is an optimal adaptive compliance control for a robotic walk assist device. The approach is based on bio-inspired RL and is completely dynamic model free. The scheme uses joint position and velocity feedback, as well as sensed joint torque applied by the user during walking, for compliance control. In particular, the RL system has a state vector containing hip and knee joint angular position and velocities and the action is selected via an actor-critic system. The effectiveness of the controller is tested through simulations using a robotic walk-assisting device model, returning angular error $\leq$0.08 rad for both the tested DoFs. In particular, is used RWAD developed in the SimMechanics toolbox (Matlab/Simulink).

Xu et al. proposed a master-slave robotic system for mirror therapy to transfer therapeutic training from the patient's functional limb to the impaired limb using a wearable robot in [125]. The IL mimics the action prescribed by the FL with the assistance of the wearable robot, stimulating and strengthening the injured muscles through repetitive exercise. The RL approach was used in the human-robot interaction control to enhance rehabilitation efficacy and guarantee safety. The study incorporated multi-channel sensed information, including the motion trajectory, muscle activation (expressed via skin surface electromyography signals), and the user's emotion (shown as facial expressions), into the learning algorithm. The RL approach was realized by the normalized advantage functions algorithm. Furthermore, the study developed a lower extremity rehabilitation robot with magnetorheological actuators, and clinical experiments were conducted using the robot to verify the performance of the framework, in particular, five hemiplegic patients participated in the experiments. During the experiment, the patient exerts force on the FL to drive the master robot, and the IL tries to actuate its own muscles to complete the gait task. In particular the classical Q-learning, in continuous action spaces with deep neural networks is used. On the whole, the study demonstrated the potential of a master-slave robotic system with RL to enhance the effectiveness and safety of mirror therapy in lower extremity rehabilitation. The incorporation of multi-channel sensed information and the use of MR actuators in the robot design can also provide valuable insights for developing more advanced rehabilitation systems [125].

Moreover, Gao et al. (2020) proposed a knowledge-guided Q-learning (KG-QL) control method for prosthesis control with a user in the loop in [126]. The controlled prosthesis is expected to adapt quickly and smoothly to different task environments, which is challenging for most RL agents who learn or relearn from scratch when the environment changes. To address this issue, the authors collected and used data from two able-bodied subjects wearing an RL-controlled robotic prosthetic limb walking on level ground, they built a regression model and used it to propose a knowledge guide Q-learning process, namely KG-QL. The ultimate goal of their study is to build an efficient RL controller with reduced time and

data requirements and transfer knowledge from AB subjects to amputee subjects. They demonstrated the feasibility of this approach by employing OpenSim, a well-established human locomotion simulator, and showed that simulated amputee subjects improved control tuning performance by utilizing knowledge transfer from AB subjects with an error of 0.01 rad. Moreover, the authors explored the possibility of information transfer from AB subjects to help tuning for the amputee subjects. Their approach can lead to the development of more efficient and effective prosthesis control systems, which could greatly benefit amputee patients [126].

The study [127] proposed a new motion controller for a lower extremity rehabilitation exoskeleton using RL. The exoskeleton is designed for collaborative squatting exercises and equipped with ankle actuation on both sagittal and front planes and multiple foot force sensors to estimate the center of pressure, a critical indicator of system balance. The proposed controller takes advantage of CoP information by incorporating it into the state input of the control policy network and adding it to the reward during learning to maintain a well-balanced system state during motions. To improve the robustness of the controller, the study also used dynamics randomization and adversary force perturbations, including large human interaction forces during training. The effectiveness of the learning controller was evaluated through numerical experiments with different settings to understand if the learning process can generate feasible control policies to control the exoskeleton to perform well-balanced squatting motions if the learned control policies be robust enough under large random external perturbation and to sustain stable motions when subjected to uncertain human exoskeleton interaction forces from a disabled human operator. The study demonstrated the potential of RL-based motion controllers for lower extremity rehabilitation exoskeletons, particularly in terms of efficiency, stability, and robustness. The proposed controller's ability to incorporate CoP information and handle large human interaction forces during training can be valuable in real world rehabilitation settings, where maintaining balance and stability is critical [127].

The paper [128] proposed an innovative approach for adaptive control of lower limb rehabilitation exoskeleton robots. The proposed approach combines policy iteration, RL, and event-triggering mechanisms to achieve online learning and adaptation while reducing the number of control updates. In particular, an adaptive online learning structure, namely Actor-Critic Neural Network, is exploited in the event triggered Optimal Control framework, The proposed EtOC is then verified on numerical simulation, and tested of a real lower limb rehabilitation exoskeleton with a final error of 0.1 rad. In general, the proposed approach presents a promising direction for adaptive control of lower limb rehabilitation exoskeleton robots [128].

The paper [129] proposed a new event triggered control strategy based on RL and policy iteration for a lower limb rehabilitation exoskeleton robot. The proposed approach integrates an event-triggering mechanism and a novel event triggered tuning law with an actor-critic neural network for online learning and adaptation. The experimental results show that the proposed method reduces the number of control updates while maintaining a guaranteed control performance compared to traditional time triggered control methods. The experiment has been simulated with an error of 0.01 rad These methods can potentially enhance the adaptability and efficiency of control systems while reducing the computational burden and energy consumption [129].

In [130], the authors proposed a deep RL-based robust controller for a lower limb rehabilitation exoskeleton. The controller is trained using a decoupled offline human exoskeleton simulation training with three independent networks. The goal is to provide reliable walking assistance against various and uncertain human exoskeleton interaction forces. The controller acts on a stream of the LLRE's proprioceptive signals, including joint kinematic states, and predicts real time position control targets for the actuated joints. To handle uncertain human interaction forces, the control policy is trained intentionally with an integrated human musculoskeletal model and realistic human exoskeleton interaction forces. Additionally, domain randomization is employed during training to increase the robustness of the control policy to different human conditions, with different neuromuscular disorders. The trained controller is able to provide reliable walking assistance to patients with different degrees of neuromuscular disorders without any control parameter tuning. The system has been tested on 3 pathological subjects and 1 healthy with a final error of 0.1 rad [130].

### Neural Networks

Neural networks are a subset of machine learning and are at the heart of deep learning algorithms, they are comprised of node layers, containing an input layer, one or more hidden layers, and an output layer. Each node is connected to another and associated weight and threshold. If the output of any individual node is above the specified threshold value, that node is activated, sending data to the next layer of the network, otherwise, no data is passed along the next layer [162].

This paper [131] discusses the use of an MLP to predict natural gait parameters for an individual based on their age, gender, body height, and body weight. The MLP is trained to output a suitable walking speed and cadence for the given subject. The study evaluates the efficiency and accuracy of the MLP in predicting the desired outputs for two different setups. In the first setup, the MLP is trained specifically for slow walking speed conditions. In the second setup, the MLP is trained for both slow and normal walking speed conditions.

The input features of the model are cadence, stride length, and walking speed. The model has been tested on 50 healthy subjects. The proposed approach was capable of returning accurate prediction with a deviation of 0.03% with respect to the experimental data. The study demonstrates the potential of using MLP for predicting natural gait parameters for an individual, which can be useful in fields such as rehabilitation and sports science [131].

In[132] the authors present a new model for generating joint angle waveforms of the lower limb during walking, using gait parameters and lower limb anthropometric data as input. The walking motion is captured using a motion capture system with passive markers, and the waveforms of lower limb joint angles are calculated from the experimental data. The waveforms are then decomposed into Fourier coefficients, which allows for easier waveform analysis. They designed an MLP to predict the Fourier coefficient vectors for specific subjects and desired gait parameters. Assessment parameters such as correlation coefficient, mean absolute deviation and threshold absolute deviation are calculated to examine the quality of the MLP prediction. The constructed waveforms from predicted Fourier coefficient vectors are compared with the actual waveforms calculated from experimental data using the assessment parameters mentioned above. The results show that the constructed waveforms closely match the experimental waveforms based on the assessment parameter outcomes, demonstrating the potential of using MLP to accurately predict joint angle waveforms of the lower limb during walking, in particular, the system has been tested on 15 healthy subjects with an accuracy of 0.98. This can be useful in fields such as biomechanics and rehabilitation where understanding and optimizing human movement is important.

The paper [133] proposed a method for gait phase classification in lower limb exoskeleton robots, using sensor signals from foot sensors with force sensing registers, as well as the orientation of each lower limb segment and the angular velocities of the joints. The authors investigated MLP and NARX. During the experiment, the subject wore ROBIN-H1 exoskeleton and walked on a treadmill following specific rules. The performance of the proposed classifiers was evaluated using offline and online evaluations based on four criteria such as Classification Success Rate, Max Continuous Error Width, Mean and Standard Deviation, and Number of unstable regions. The results showed that the NARX-based method exhibited satisfactory performance in replacing foot sensors as a means of classifying gait phases. The feature used as input is human forces. The system has been tested on 7 healthy subjects with an accuracy of 0.98 The results suggest that this approach could lead to more accurate and reliable gait phase classification, which could be useful for improving the performance and safety of these robots [133].

The paper [134] proposes a method called Deep Rehabilitation Gait Learning (DRGL) for modeling the knee joints of lower limb exoskeletons. The method leverages Long-Short Term

Memory (LSTM) to learn the inherent spatial-temporal correlations of gait features. With DRGL, abnormal knee joint trajectories can be predicted and corrected based on the wearer's other joints, without requiring complex kinematic and dynamic models for the human body and exoskeleton. The main advantage of DRGL is that the new recovery gait pattern is not only in accordance with the healthy walking gait but also includes the wearer's own gait profile. To verify the effectiveness of DRGL, the authors obtained a new recovery gait from DRGL based on pathological gait, which was obtained by having a healthy subject imitate knee injury. The experiments demonstrated that the subject could walk normally with the SIAT lower limb exoskeleton in the new recovery gait pattern generated by DRGL. In particular the system has been tested on 20 healthy subjects. The proposed neural architecture is capable of reconstructing human trajectories with an angular error $\leq$0.30 rad and capturing the key gait features with high consistency. This suggests that DRGL is a promising method for improving gait rehabilitation using lower limb exoskeletons. In summary, the paper presents a novel approach for gait rehabilitation using deep learning techniques. The proposed DRGL method could have important implications for the design and development of lower limb exoskeletons that can help patients recover from knee injuries or other conditions affecting their gait [134].

The paper [135] aimed at developing a neural network that accurately simulates the changes in the angle of the ankle, knee, and hip joints during the gait cycle, and to use it to simulate the impact of a restricted range of ankle and hip joint angle changes on the progression of the knee joint angle. The study involved 34 young healthy students, and gait kinematics data were collected using the Vicon system, which was then analyzed with an MLP. The results showed that the developed MLP was able to accurately simulate the progression of joint angles of lower limb motion with an accuracy of 0.94, and its simulation of the impact of restricted ankle and hip joint angular ranges on the knee joint indicated that the braking phase is critical. The study highlights the potential of NNs as a useful research method in clinical biomechanics and suggests that further research in this vein could expand our understanding of compensatory functions in the lower limbs. The findings could be useful in the design and development of lower limb exoskeletons and other assistive devices, as well as in the rehabilitation of patients with lower limb injuries or conditions affecting their gait [135].

The paper [136] presents two control schemes for the exoskeleton to conduct trajectory tracking tasks in the presence of unmodeled dynamics of the exoskeleton, interaction between human and exoskeleton, and additional disturbances. The first controller presented is purely NN-based (Radial basis function NN), and adopts a combined error factor (CEF) to enhance human safety by improving transient response. The CEF consists of the weighted sum of

tracking error and its derivative. The second controller is developed based on a combined scheme of repetitive learning control (RLC) and radial basis function NN, where the add-on RLC is used to learn periodic uncertainties that contribute to the repetitive motion of the exoskeleton leg. The stabilities of the controllers are proved rigorously in a Lyapunov way. The study highlights that while the pure NN controller can deal with periodic and non-periodic uncertainties simultaneously, the main feature of exoskeleton motion during rehabilitation therapy, namely the repetitiveness, is fully ignored, and this could degrade the tracking performance. The proposed control method is shown to achieve a significant control effect with remarkable transient performance in comparison to the other methods used in the simulation. The input data used was EMG and the system has been simulated achieving an error $\leq \cdot 10^{-3}$ rad. On the whole, the study demonstrates the potential of NN-based control methods for improving the performance and safety of lower limb exoskeletons in rehabilitation therapy. The results could be useful in the design and development of advanced control strategies for assistive devices in rehabilitation and other applications [136].

The paper [137] presents a novel method for predicting joint moments using a small number of input variables selected by Elastic Net and MLP. The method is tested on experimental data collected from healthy subjects running on a treadmill at different speeds. The results show that the method can accurately predict joint moments with only 5-6 EMG signals as input, with a normalized root-mean-square error (NRMSE) lower than 7.89% and a cross-correlation coefficient (p). This method can effectively reduce the number of input variables required for joint moment prediction, which may facilitate real time gait analysis and exoskeleton robot control in motor rehabilitation [137].

This paper [138] describes a study on the application of a bionic control method based on a Central Pattern Generator (CPG) to control the lower limb exoskeleton for rehabilitation purposes. The authors improved the Hopf oscillator using the Dynamic Hebbian Learning algorithm and built a CPG oscillator network to generate gait signals to control the lower limb exoskeleton. The study aimed at improving the performance and adaptability of the exoskeleton by matching it with the control signals produced by the human body during motion cycles. It included wearing tests on patients, and the results obtained by the simulation experiments of gait curves of different motion modes showed that the CPG bionic control exoskeleton could effectively control the lower limb exoskeleton and perform rehabilitation exercises. The system has been simulated, the input data was hip, knee, and joint torque plus speed the final angle error was 0.90 rad. The study provides valuable insights into the application of CPG-based control methods for exoskeleton rehabilitation and highlights the potential for improving the performance and adaptability of lower limb exoskeletons for rehabilitation purposes [138].

The paper [139] proposes an intelligent control method for rehabilitation robots using sEMG and human force-based dual closed-loop control strategy. The method aims to adaptively control the rehabilitation robots to better function in the rehabilitation field. The proposed approach has several contributions. First, it uses a wavelet neural network (WNN) to obtain the desired trajectory of patients based on the sEMG signal. Second, it modifies the reference trajectory by the variable impedance controller (VIC) based on the sEMG and human force. Third, it uses a model reference adaptive controller (MRAC) with parameter updating laws based on the Lyapunov stability theory to force the rehabilitation robot to track the reference trajectory. The experiment results show that the proposed approach efficiently decreases the trajectory tracking error and adapts the reference trajectory to synchronize with the patients' motion intention. The model reference controller is able to outstandingly force the rehabilitation robot to track the reference trajectory. The final system has been tested on 4 healthy subjects with an error $\leq$0.01 m. The proposed method is expandable to other applications in the rehabilitation field and has the potential for designing an intelligent control of rehabilitation robots.

The paper [140] dealt with the recovery of a patient's limb and is driven to follow a planned trajectory during rehabilitation training. The accuracy of trajectory tracking is essential for effective rehabilitation training.PID control is a conventional method for trajectory tracking, but due to the dynamic model uncertainties and lack of good adjustment ability, it may not be sufficient. Therefore, the authors combined RBF neural network and PID control to improve the accuracy of trajectory tracking. The Magnetorheological (MR) damper and motor for actuation. The control is simulated in Simulink, and the trajectory tracking errors under PID control and RBF-PID control are compared. The results obtained through the simulation experiment show that RBF-PID control has better anti-interference performance, which improves the flexibility of movement, real time, and stability of trajectory tracking during the rehabilitation robot work. The system has been simulated with a final error of 0.02 rad.

The paper [141] proposed an individualized gait pattern generation method based on a recurrent neural network (RNN) for creating a function mapping from body parameters and gait parameters to a gait pattern. The proposed method is trained on the largest gait data set of this kind, which consists of 4,425 gait patterns from 137 healthy subjects [141]. The RNN is proficient in series modeling and can generate gait patterns at continuously varying walking speeds and stride lengths. The proposed model's experimental results indicate that it reduces the errors in ankle, knee, and hip measurements by 12.83%, 20.95%, and 28.25%, respectively, compared to the previous state-of-the-art methods, in particular with a generalized regression neural network [163] and a gaussian process regression [164].

It has significant implications for personalized rehabilitation training, including designing customized rehabilitation programs, identifying patients at risk of falls, and assessing the effectiveness of the rehabilitation program. The paper's findings demonstrate the potential of using RNNs for gait pattern generation in rehabilitation. [141].

The paper [142] proposed an RNN for rehabilitation evaluation. This method extracts gait characteristic parameters of patients with different ages, disease types, and courses of disease by learning existing clinical gait data. The algorithm uses repeated data iteration to simulate the corresponding gait parameters of patients on 36 healthy subjects. Experiments show that the trained ANN algorithm has a high accuracy rate when compared to human raters for most of the data (82.2%, Cohen's I kappa=0.743). The algorithm also has a strong correlation with improved Ashworth scores as assessed by human raters (r=0.825).

The paper [143] proposes a two-stage attention model placed inside an LSTMs-based encoder decoder architecture for predicting human gait trajectories using a combination of real and synthetic gait data. The authors use a Generative Adversarial Network (GAN) with temporal Convolutional Neural Network CNNs to generate synthetic human gait data (ratio 4:1 synthetic vs real data that retains the dynamics of real gait data. They collected real human gait data from five healthy subjects using a NOKOV optical motion capture platform. The authors compare the performance of their GAN model with a traditional LSTM model and found that the attention mechanism (MLP) had a higher capacity for learning dependencies between historical gait data to accurately predict the current values of the hip joint angles and knee joint angles in the gait trajectory. The results obtained by the collection of five healthy subjects' gait data indicate that GANs-based data augmentation can synthesize realistic- looking at multi-dimensional human gait data. The predicted gait trajectories based on the historical gait data can be used for gait trajectory tracking strategies. This paper demonstrates the potential of using synthetic data to augment real data and improve the performance of machine learning models for human motion analysis tasks. The final accuracy is about 0.88 [143].

This study [144] focuses on the development of a lower limb robotic exoskeleton for rehabilitation purposes, using artificial neural networks to improve its control system. The researchers first test the proposed lower limb robotic exoskeleton robot (LLRER) using a PID control with an iterative learning controller. However, they found that the knee part of the LLRER, which uses PAM actuation, does not perform very well due to nonlinearity. To compensate for this nonlinearity, the researchers used an MLP control based on the inverse model trained in advance. They also used particle swarm optimization (PSO) to optimize the PID parameters based on the MLP architecture. The results show that the MLP with PID control (PSO tuned) performs the best among the three controllers. The average Mean

Absolute Error (MAE) of the left knee joint is 0.03 rad and the average MAE of the right knee joint is 0.024 rad, red tested on 10 healthy subjects. During rehabilitation tests, the controller of MLP with PID control was found to be suitable, and its versatility for different walking gaits was verified during human tests. The researchers found that the establishment of the inverse model does not need to use complex mathematical formulas and parameters for modeling, making it more accessible. Additionally, the use of PSO to search for the optimal parameters of the PID and the architecture diagram and control signal given by the MLP compensation with the PID control effectively reduced the error.

### Support Vector Machines

This section presents the application of SVM to lower limb rehabilitation robotics, which is a supervised learning method used for classification and regression. The name came from the data points called support vectors that are closer to the hyperplane and influence the position and orientation of the boundary in the feature space. Using these support vectors, we maximize the margin of the classifier [165].

In [145], the authors explore the use of an SVM for identifying locomotion intentions from surface electromyography (sEMG) data. The study uses a phase-dependent approach, which is based on foot contact and foot push-off events, to contextualize muscle activation signals. The study demonstrates good accuracy on experimental data from three healthy subjects. The classification accuracy is also tested for different subsets of EMG features and muscles, with the aim of identifying the minimal setup required for the control of an EMG-based exoskeleton for rehabilitation purposes. The study shows that SVM can be used to accurately identify locomotion intentions from sEMG data. The phase-dependent approach used in this study helps to contextualize muscle activation signals, which improves the accuracy of the classification. The system has been validated on 3 healthy subjects with an accuracy of 0.95 The study also highlights the importance of selecting the optimal subset of EMG features and muscles for the control of EMG-based exoskeletons for rehabilitation purposes [145].

The paper [146] proposed a motion intent recognition method to control a wearable lower extremity assistive device intended to aid stroke patients during activities of daily living or rehabilitation. The primary goal is to identify the user's intended motion based on sensor readings from the limb attached to the assistive device to execute the right control actions to aid the user in his intended action effectively. To this end, the study collected a database of 1 healthy subject performing various motion tasks. The features of the signals are extracted, and Principal Component Analysis is performed to reduce the number of dimensions. Using the transformed signal, a multi-class SVM with a Radial Basis function kernel is trained

to classify the different motion patterns. A Nelder-Mead optimization algorithm is used to select the appropriate parameters for the SVM. An offline classification result of a healthy subject performing a series of motion tasks while wearing the LEAD shows that the proposed method can effectively recognize different motion intents of the user.The test results show that the SVM can correctly classify each motion pattern with an average accuracy equal to 95.8±4.1%. The study demonstrates the potential of using motion intent recognition methods to control wearable assistive devices intended to aid stroke patients during ADL or rehabilitation. The proposed method can effectively recognize different motion intents of the user, and the high accuracy rate of the SVM classification suggests that this approach may have practical applications in real world settings.

The paper [147] proposed two machine learning models for predicting gait phases from spatial and spatiotemporal perspectives using joint angle data collected from four goniometers and plantar pressure distribution data from three force-sensitive resistors. The two models are SVM optimized by particle swarm optimization algorithm and nonlinear autoregressive models with external inputs. The results of the experiment show that both models are capable of predicting gait phases, but NARX outperforms SVM in terms of accuracy since it utilizes FSR data to correct the wrong predictions. The system has been validated on 10 healthy subjects with an angular accuracy of 0.087 rad The authors suggest that it is better to predict gait phases based on both space and time dimensions simultaneously. The paper presents an interesting approach to predicting gait phases using machine learning techniques and demonstrates the effectiveness of incorporating spatiotemporal features in the prediction process [147].

The paper [148] discusses the use of a Kinect sensor and a lower limb rehabilitation robot to provide targeted rehabilitation training. The Kinect sensor is positioned in front of the robot and is used to acquire horizontal distance data from markers placed on the patient's body during the rehabilitation training. This data is used to identify different patients using an SVM. By identifying different patients, the system can provide personalized and targeted rehabilitation training to each patient based on their specific needs. This approach is more effective than a one-size-fits-all approach as it takes into account individual differences in gait and movement patterns. The system has been validated on 3 healthy subjects with an accuracy of 0.95. In summary, the paper demonstrates the potential of combining technology, such as the Kinect sensor, with rehabilitation robots to improve the effectiveness of rehabilitation training and provide personalized care to patients.

The paper [149] proposes a real time electromyography-triggered controller for a pneumatic artificial muscle actuated lower limb rehabilitation robot. The proposed controller is designed to make the rehabilitation task controllable by the patient's movement intention.

To this good, the EMG signals of the patient's muscle are captured and identified using the discrete wavelet transformation technique to acquire the feature vectors of the EMG signals. The optimal multicomponents of features are chosen based on the experimental results, and SVMs are studied to improve the classification performance. To implement the closed-loop control system for the rehabilitation robot with the movement-intention trigger control, they also used the MyRIO controller. This system allows the patient's movement intention to be accurately identified by EMG feature extraction, ensuring the safety and performance of the proposed system. The system has been tested on 8 healthy subjects with a final accuracy of 0.99. In summary, the paper demonstrates the potential of using EMG-triggered control for lower limb rehabilitation robots, which can help provide personalized care to patients by enabling the robot to respond to the patient's movement intention in real time [149].

The paper [150] proposes a method for classifying the stance phase and swing phase during healthy human gait based on the muscle activity in both legs using SVM. The paper introduces a novel EMG feature calculated from the bilateral EMG signals of muscle pairs, showing promising results with classification accuracies of up to 96% tested on 8 healthy subjects. In particular, all used motion data were taken from the HuMoD database which is an open-source human motion dynamic database [166]. The proposed method could potentially have practical applications in the design of rehabilitation devices or assistive technologies that are intended to aid individuals with gait impairments. The use of SVMs in this study suggests that this approach may be effective for other classification tasks in the field of rehabilitation, and the introduction of the novel EMG feature may inspire further research into the development of new features for classification tasks.

The paper [151] describes the development of a multi-sensor fusion gait recognition system for accurately controlling exoskeleton movement. The system acquires plantar pressure and acceleration signals of human legs, and in the experiment, the pressure signals of both feet and movement data of the waist left thigh, left calf, right thigh, and right calf of five test subjects were collected. The test lasted 3 min and consisted of standing, going up the stairs, going down the stairs, going up and down the slope, and walking on level ground. The study investigated six different gaits, including standing, level walking, going up the stairs, going down the stairs, going down the slope, and going down the slope. The authors demonstrated that the SVM outperformed Multilayer perceptron (MLP), and radial basis function (RBF) neural networks models. The study also analyzed the different sliding window sizes of the SVM algorithm. The results showed that the SVM algorithm had the highest recognition rate with an average recognition accuracy equal to 96.5%, tested on a total of 5H subjects. The accurate recognition of human gait provides a good theoretical basis for the design of an exoskeleton robot control strategy. The study's findings could have potential applications in

the development of assistive devices for people with mobility impairments and in the design of exoskeletons for industrial and military use.

The study [152] aims to improve gait phase classification in an exoskeleton robot using only the angle of hip and knee joints by introducing a kernel recursive least square algorithm to build a classification model that considers the adaptation of unique gait features. Additionally, an assist torque predictor based on the KRLS algorithm is also developed. The study compares the performance of the KRLS model with two other commonly used gait recognition methods, MLP and SVM, using gait data collected from 10 healthy volunteers wearing the exoskeleton robot. The results show that the KRLS classification accuracy is on average 3% higher than MLP and SVM, with a testing average accuracy of 86%. The KRLS algorithm also performs twice as well as MLP in assisting torque prediction experiments. Furthermore, the KRLS algorithm is demonstrated to be stable, robust, and able to generalize well to different datasets. The study suggests that the KRLS algorithm is a promising method for improving gait phase classification and assisting torque prediction in exoskeleton robots[152].

In [153] the use of SVM in this application is based on its ability to classify and recognize patterns in complex datasets, such as the sEMG signals collected from the surface of human muscles. The proposed method involves the collection of sEMG signals from the human body during different motion postures, such as walking, standing, or sitting. These signals are then processed and analyzed using an SVM classifier to recognize the intended motion posture of the wearer. The output of the SVM classifier is then used to plan the moving gait of the exoskeleton, and the decoding intention signal controls gait switching. To ensure the stability of the planned gait during movement, the researchers also analyzed the stability of the exoskeleton during the execution of different motion postures. The experimental results showed that the SVM-based method of decoding sEMG signals for human motion intention and controlling exoskeleton robot gait switching had good accuracy and real time performance. The system has been tested on 10H subjects with an accuracy of 0.95 The application of SVM to lower limb robotics has significant potential for aiding in the rehabilitation of patients. By using sEMG signals to identify human motion intentions and control the exoskeleton's movement, patients can complete rehabilitation training more safely and quickly. Additionally, the use of SVM in this application may have implications for other fields that require the real time recognition and classification of complex datasets [153].

The paper [154] describes an experiment for lower extremity action recognition using sEMG signals. The experiment was designed based on the principle of human lower extremity force generation, and the aim was to classify five common lower extremity actions Natural

walking, sitting, standing up, and going up, and down the stairs.). In particular, the subject performed specific movement, and a set of raw sEMG data were collected. The authors extracted the sEMG time-domain feature used to train an SVM classifier. The results of the experiment showed that the SVM classifier achieved an average accuracy rate equal to 90.66% evaluated from 6 healthy subjects, which verified the effectiveness of the experimental design. In summary, the paper presents a novel approach to lower extremity action recognition using sEMG signals and demonstrates the effectiveness of using time-domain features and an SVM classifier. The results of the experiments will have potential applications in fields such as rehabilitation, sports training, and human-machine interaction [154].

**Decision Tree**

A decision tree is a non-parametric supervised learning algorithm, used both for classification and regression tasks. It is characterized by a hierarchical, tree structure, which consists of a root node, branches, internal nodes, and leaf nodes [167].

The paper [155] presents a method for identifying the sub-phases of gait in human-machine coordinated motion using an exoskeleton. The authors introduce a sensor layout that includes shoe pressure sensors, knee encoders, and thigh and calf gyroscopes to measure the contact force of the foot, knee joint angle, and its angular velocity. To differentiate between human lower limb motion and human-machine coordinated motion, the authors divide the walking gait into five sub-phases: double standing, right leg swing, and left leg stance, double stance with the right leg front and left leg back, right leg stance and left leg swing, and double stance with the left leg front and right leg back. The authors use a C4.5 decision tree algorithm to fuse the sensor information blue and through classification identify the five sub-phases of walking gait. Based on simulation results, the proposed algorithm guarantees identification accuracy. The experimental results performed on 1 healthy subject verify the gait division and identification method. The input features are force knee angle and angular velocity and the final accuracy is about 0.99. Moreover, the authors suggest that the proposed method can make hydraulic cylinders retract ahead of time and improve the maximal walking velocity when the exoskeleton follows the person's motion [155].

The paper [156] proposed a method based on anthropometric features for predicting patient specific gait trajectories. The authors noted that human gaits are closely related to anthropometric features, but this relationship has not been well-researched. The proposed method uses the Fourier series to fit gait trajectories and represent gait patterns by the obtained Fourier coefficients. The authors use human age, gender, and 12 body parameters (Age, Height, Mass, Gender, Thigh length, Calf length, Bi-trochanteric width, Bi-iliac width, ASIS breath, knee diameter, Foot length, Malleolus height, Malleolus width, Foot breath) to

design the gait prediction model. To make the method applicable to lower limb rehabilitation robots, the anthropometric features are selected using an optimization method based on the minimal-redundancy-maximal-relevance criterion. The relationship between the selected features and human gaits is modeled using a random forest predicting patient specific gait trajectories. The system has been validated on data collected in a previous study [164] from 113 healthy subjects [163] achieving an error of 0.08 rad.

The paper [157] compares the performance of three algorithms for recognizing three different gait phases of a passive lower limp exoskeleton: C4.5 decision tree, MLP, and NARX. The gait phases are stance, swing, and push, and the algorithms use two IMU sensors on the hip and knee joint position and two FSR sensors in a self-made shoe to provide four inputs. Test data containing the pressure of shoes contacting the ground, knee angles, and knee angular velocities is collected at different walking speeds, and the experimental results show the classification success rate of each algorithm when trained with different pattern sizes. The system final accuracy is equal to 1 for C4.5, 0.98 for MLP, and 0.948 for NARX, tested on 1 healthy subject The experiments help to determine the most suitable algorithm for recognizing different gait phases in real time applications [157].

The work [158] aimed at identifying the factors that affect home discharge after stroke inpatient rehabilitation, including both functional and environmental factors, using the machine learning method. The authors note that while the importance of environmental factors for stroke patients to achieve home discharge has been discussed, there are limited studies on the application of the decision tree with various functional and environmental variables to identify stroke patients with a high possibility of home discharge. To address this gap, the authors collected a private dataset including information on 481 stroke patients' functional status and environmental factors, such as living arrangements and social support, and used the three classification and regression tree (CART) models to identify the factors that predict home discharge with a final accuracy of 0.85. The results showed that functional factors, such as motor function and cognitive function, were the most important predictors, followed by environmental factors, such as living arrangements and social support [158].

The paper [159] proposes a motion rehabilitation robot control system for lower limb postoperative rehabilitation training, which is based on human posture information. The system has several functions, such as active/passive training mode control, movement posture and EMG signal acquisition, WiFi communication, and safety protection. The training process is recognized and analyzed using random forest and linear regression. The experimental results performed on 10 healthy subjects show that the random forest algorithm has better performance in motion recognition than the linear regression algorithm with an accuracy of 99.7%. The features used in the system are hip knee and ankle positions. The

developed control and monitoring system can be controlled by Android and can realize the intelligent analysis of the training process through the monitoring signals in the training process [159].

The gait deviation correction method proposed in [160] aims to decrease the deviation of the wearer's gait when using a lower limb exoskeleton robot.By using the clinical gait analysis curve as a reference trajectory and incorporating body feature parameters, the gait correction model modifies the input trajectory of the exoskeleton to reduce gait deviations. In particular, the gait deviation correction method is based on the XGboost algorithm a commonly used algorithm in the industry due to its highly-efficient implementation of the gradient boosting algorithm. To train the algorithm CGA curves have been used. The results performed on 15 healthy subjects showed that the correction led to closer gait trajectories to the reference curve by reducing the error at 0.11 rad, suggesting potential benefits in the subsequent training efficacy [160].

## 2.3 Discussion

Table 2.5 reports the paper presented in this review grouped by applications and AI algorithm implemented.

Table 2.5: Overview of the works analyzed in this section grouped by application of the AI-based algorithm.

| | Robot Control (RC) | Locomotion Classification (LC) | Intention detection (ID) | Human Joints Trajectory Prediction (HJTP) |
|---|---|---|---|---|
| Reinforcement Learning | [121, 122, 123, 124, 125, 126, 127, 128, 129, 130] | | | |
| Support Vector Machine | [149] | [145, 147, 148, 150, 151, 152, 154] | [146, 153] | |
| Neural Network | [136, 138, 139, 140, 143, 144] | [131, 133, 142, 157] | | [132, 134, 135, 137, 141, 143] |
| Decision Tree | [159, 160] | [155, 156, 157, 158] | | |

Analyzing the table by column we notice that in the area of Robot Control (RC), researchers have extensively explored RL and NN. RL has garnered significant attention, as evidenced by a substantial number of research papers. RL's popularity can be attributed to

its unique ability to enable robots to learn optimal control strategies through interactions with their environment. By receiving feedback in the form of rewards or penalties, RL algorithms iteratively improve their decision-making processes, leading to adaptable and robust control in complex and dynamic tasks. On the other hand, Neural Networks have also been applied in the context of robot control, albeit to a lesser extent compared to RL. These models are known for their capacity to approximate complex functions and effectively learn from high-dimensional data [144]. When applied to robot control tasks, Neural Networks can capture intricate patterns and relationships in sensor data or articulation configurations, facilitating more nuanced and sophisticated control strategies. Both RL and Neural Networks have showcased promising results in the field of Robot Control. The choice of which algorithm to employ often depends on the specific requirements of the control task, with RL being favored in scenarios that necessitate adaptive learning and dynamic decision-making, while NNs are effective when dealing with high-dimensional sensor data and intricate control mappings. In summary, the combination of these two techniques showcases the ongoing advancements in using AI to enhance and refine robot control capabilities.

In the field of Locomotion Classification (LC), researchers have employed three machine-learning techniques: SVM, NN, and DT. Among these techniques, SVM [149] has been particularly used for pneumatic locomotion. SVM's use in this domain is limited, but it remains relevant due to its ability to effectively classify data in binary setting and thanks to the "kernel trick" its possible also to classify elements that originally aren't linearly separable. Conversely, Neural Network (NN) exhibits a more diverse range of applications in locomotion classification [144]. NNs are well-suited for thus task as they learn intricate relationships between input signals and different locomotion types. By training on vast datasets, NNs can capture complex patterns and variations in human movement, leading to accurate and robust classification results. Decision Trees, while less explored in the locomotion classification domain, are still represented by [159, 160]. They offer interpretability, allowing researchers to gain insights into the decision-making process of the model. Additionally, Decision Trees can effectively handle categorical data, making them suitable for classification tasks with discrete outcomes, such as distinguishing between different locomotion types. The three machine learning techniques used in Locomotion Classification each offer unique advantages: SVM for its effectiveness in handling non-linear classification tasks, NN for its ability to capture complex patterns, and Decision Trees for their interpretability and suitability for categorical data. The combination of these techniques showcases the diversity of approaches researchers employ to address the challenges of locomotion classification and further advance the field.

In the domain of Intention Detection (ID), researchers have predominantly employed

two primary machine learning techniques: RL and SVM. While both methods have been utilized, RL has been explored in a more limited number of papers [153, 158]. On the other hand, SVM is involved in more tasks [147, 152]. Intention detection plays a crucial role in human-robot interaction, as it enables robots to comprehend and interpret human intentions and commands accurately. By employing RL in intention detection, researchers aim to create intelligent and adaptable systems that can learn from interactions with humans and optimize their decision-making processes accordingly. RL allows robots to understand human intentions through a continuous feedback loop, enhancing the effectiveness of human-robot communication and cooperation. Meanwhile, the higher representation of in intention detection research highlights its efficacy in addressing classification tasks related to human intentions. SVM excels in binary classification, making it a suitable choice for discerning various human commands or intentions in real time scenarios. Its ability to effectively classify data based on feature vectors from sensors or inputs further enhances its applicability in human-robot interaction. In summary, the combination of RL and SVM in intention detection research emphasizes the importance of developing accurate and reliable methods for enabling seamless and intuitive human-robot communication. As researchers continue to explore and refine these machine-learning techniques, the potential for advancing human-robot interaction and collaboration in various domains becomes increasingly promising.

In the field of Human Joints Trajectory Prediction (HJTP), NN stands out as the primary machine learning technique employed, as indicated by the considerable number of cited papers up to [143]. The prevalence of Neural Networks in this domain is a testament to their remarkable ability to capture complex spatiotemporal patterns in human joint movements, making them particularly well-suited for trajectory prediction tasks. Neural Networks excel in learning from large datasets and extracting meaningful representations from sequential data, which is critical for predicting the continuous and dynamic nature of human joint trajectories. By utilizing recurrent and convolutional architectures, NNs can effectively model the dependencies and interactions between joint positions over time, allowing them to forecast future joint movements accurately. The utilization of Neural Networks in HJTP research signifies the growing recognition of their potential to enhance human-robot interactions, rehabilitation processes, and motion analysis. The ability to accurately predict human joint trajectories contributes to safer and more efficient human-robot collaboration, as robots can anticipate and adapt to human movements in real time. Moreover, the advancement of Neural Network architectures, such as Long Short-Term Memory (LSTM) networks and Transformer-based models, has further strengthened their predictive capabilities, enabling them to handle longer temporal dependencies and capture finer-grained patterns in joint movements. As HJTP continues to be a critical area of research, NN remains at the fore-

front of innovation, driving advancements in human-robot interaction, assistive robotics, and personalized rehabilitation. The ongoing developments in NNs promise to revolutionize how robots perceive and interact with human users, ultimately enhancing the performance and safety of human-robot collaborative tasks.

## 2.4 Conclusion

In this chapter, an extensive literature review on the use of artificial intelligence in lower limb robot-aided rehabilitation has been presented. The review analyzed how machine learning and deep learning techniques have been applied across the main functional components of rehabilitation robotic systems, including robot control, walking pattern classification, motion interaction detection, and motion planning.

The analysis highlights that, despite the rapid growth of AI-based methodologies in the scientific literature, most robotic devices currently deployed in clinical settings still rely on traditional control and decision-making strategies, with limited integration of artificial intelligence. This gap between research advances and clinical practice suggests a significant opportunity for the translation of AI-driven approaches into real world rehabilitation technologies, particularly in the context of robot-assisted walking.

This chapter further shows that different classes of AI algorithms tend to be associated with specific problem domains. For instance, reinforcement learning approaches are predominantly employed for control-related tasks in exoskeletal devices, whereas other machine learning paradigms demonstrate greater flexibility and applicability across multiple rehabilitation functions. These observations underscore the importance of selecting AI methodologies that are not only technically effective but also aligned with the operational and clinical requirements of lower limb exoskeleton systems.

Beyond technical considerations, this chapter emphasizes the broader implications of integrating artificial intelligence into robot-aided rehabilitation. Ethical aspects emerge as a central concern, requiring a careful balance between technological efficiency and the preservation of human centered care. Ensuring patient safety through robust validation standards, maintaining human supervision in automated decision-making, and promoting transparency and interpretability in AI-driven systems are all critical factors for fostering trust and acceptance in clinical environments. Moreover, addressing potential algorithmic biases through human oversight is essential to avoid disparities in treatment outcomes across diverse patient populations.

In this chapter provides a comprehensive foundation for the subsequent experimental investigations presented in this thesis. By identifying both the technological potential and

the current limitations of AI in lower limb rehabilitation robotics, it motivates the need for systematic empirical studies aimed at improving human–robot interaction, enhancing locomotion understanding, and supporting clinically viable, ethically grounded AI-driven rehabilitation solutions.

# Chapter 3

# Deep learning for human locomotion analysis in lower limb exoskeletons: a comparative study

**Publication Statement**

Parts of this chapter are based on the following peer-reviewed publication: "Deep learning for human locomotion analysis in lower-limb exoskeletons: a comparative study", authors: Coser et al., Journal: Frontiers in Computer Science.

## 3.1 Introduction

The field of lower limb robotics for rehabilitation and assistance leverages advanced robotic technologies to support individuals with lower limb impairments or disabilities [168]. Robotic applications in clinical practice include wearable exoskeletons that augment natural movement [169], robotic prostheses that restore ambulation in users with artificial limbs [170], and rehabilitation robots that deliver targeted exercises [171]. These systems are increasingly used in stroke recovery, spinal cord injury rehabilitation, and age-related mobility support, where precise locomotion control and adaptability to varied terrain are critical for safety and therapeutic outcomes. By enhancing rehabilitation, fostering independence, and enabling personalized therapy, these innovations show promise in improving patient outcomes. However, optimal performance in these systems requires robust data-driven parameter extraction and context-aware interventions. Control strategies must incorporate gait parameters to dynamically adapt to changing terrain conditions

A key component of such control strategies is human locomotion analysis, particularly

terrain recognition and the quantification of slope or stair height [172]. real time identification of terrain transitions—such as stair ascent or ramp descent—enables adaptive control policies that reduce fall risk and improve user confidence in daily, unsupervised environments. Wearable sensors are instrumental in capturing locomotion data, offering real time insight into human movement. Among the most widely used are Inertial Measurement Units (IMUs), which capture acceleration and angular velocity [173, 174], and electromyography (EMG) sensors, which monitor muscle activation [175, 176].

Traditional machine learning techniques, including k-Nearest Neighbor (kNN), Support Vector Machines (SVM), Gaussian Mixture Models (GMM), and Random Forests (RF), have been extensively applied to classify locomotion data from IMU and EMG sensors. However, these methods typically require handcrafted features, which are time-consuming, domain-dependent, and prone to human bias [177]. In contrast, deep learning models can automatically extract features directly from raw sensor data, improving model generalization and performance, especially for complex tasks like terrain classification and parameter estimation in wearable robotics. Deep learning models have outperformed classical machine learning techniques in both classification accuracy and adaptability, making them the de facto standard for state-of-the-art performance in human locomotion tasks, as it is shown in [178]. This methodological shift reflects a broader trend, and our study contributes to consolidating this transition through a structured evaluation.

Table 3.1 summarizes existing work in human locomotion analysis, including sensor modalities, model types, datasets, and validation approaches. It covers traditional models (LDA, kNN, DT, RF, SVM) [178, 179, 180] and deep learning architectures such as CNN, LSTM, CNN-LSTM hybrids, ResNet, ANN, DANN, MCD, and attention-based CNN-BiLSTM [181, 182, 183, 184, 185, 186, 187, 188]. These methods address tasks such as gait analysis [189], locomotion intent prediction [188], terrain classification [190], and joint moment estimation [191]. While most report accuracies above 90%, lower performance (e.g., 74%) often appears in transfer learning scenarios. Dataset sources vary, with public datasets like CAMARGO [192], ENABL3S [193], and DSADS [194], and participant numbers ranging from 5 to 500. Validation strategies—such as k-fold, hold-out, leave-one-subject-out, or leave-one-terrain-out—play a key role in assessing model generalizability.

Despite growing interest in terrain classification and parameter estimation using wearable sensor data, the current literature faces key limitations. First, validation procedures are often inconsistent and insufficiently described, making it difficult to assess model robustness. Second, many studies rely on proprietary datasets, which hinders reproducibility and benchmark comparisons. These issues impede progress toward real world-ready systems.

This study is motivated by the need to enable wearable robotic systems to accurately

interpret terrain characteristics and adapt in real time. Accurate classification of ground conditions and estimation of parameters such as ramp inclination and stair height are essential for effective human-robot interaction. Our review of current literature revealed a lack of systematic performance comparisons across state-of-the-art deep learning models for this task. To address this, we evaluate a broad set of deep learning models, including some adapted from other time series domains, and apply Explainable AI techniques to identify key sensor inputs for optimal performance.

The objective of this comparison was to overcome the key limitations identified in the extant literature. Prior studies often evaluate only one or two models in isolation, use non-public datasets that hinder reproducibility, or apply inconsistent validation strategies such as unspecified hold-out methods. In contrast, our work systematically compares eight deep learning architectures under uniform conditions using the publicly available CAMARGO 2021 dataset. To facilitate a fair comparison, the following methodologies are adopted: standardised preprocessing, consistent cross-validation, and shared training settings. Additionally, we address the limited attention to regression tasks and sensor optimization by incorporating parameter estimation and Explainable AI-based sensor reduction.

In this paper, we propose a neural-network-based system for terrain classification (level ground, stair ascent/descent, ramp ascent/descent) and parameter regression (inclination, step height) using multimodal sensor data from the CAMARGO dataset. Our contributions are as follows:

1. We perform a comparative analysis of eight deep learning models used for time series classification and regression in multimodal locomotion data. Leave-one-subject-out cross-validation ensures robustness and generalizability.

2. We investigate unimodal versus multimodal configurations, employing Explainable AI and ablation studies to determine minimal sensor setups that maintain high performance.

3. We utilize the publicly available CAMARGO 2021 dataset, which provides a high-quality, ethically sound basis for locomotion recognition research.

Table 3.1: Summary of methodologies for human locomotion analysis for terrain and slope recognition. (LG: Level Ground, RA/D: Ramp Ascent/Descent, SA/D: Stair Ascent/Descent). * indicates multimodal analysis. CWP: Collected within the paper.

| Ref. | Modality | Algorithm | Brief Description | Acc. | Dataset | # of Classes | # of Subjects | Validation Approach |
|---|---|---|---|---|---|---|---|---|
| [179] | EMG + Acc., * early fusion | kNN, SVM, RF, LDA, DT | Surface EMG from tibialis anterior and gastrocnemius muscles combined with acceleration signals were used to classify five terrain types, aiming to maximize accuracy with minimal computation and sensor usage. | 0.97–0.99 | CWP | 5 (LG, RA, RD, SA, SD) | 15 | k-fold cross-validation |
| [178] | EMG + IMU, * early fusion | LDA, SVM, ANN, CNN, DANN, MCD | An unsupervised cross-subject adaptation framework was introduced to predict locomotion intent without labeled target data, improving generalization for wearable robot control. | 0.90–0.96 | ENABL3S, DSADS | 5 (LG, SA, SD, RA, RD) | 10 | Leave-one-subject-out |
| [181] | IMU | CNN | A hierarchical CNN-based approach was proposed for real-time locomotion mode recognition in wearable prostheses and exoskeletons, enabling smoother transitions and stable predictions. | 0.94 | CWP | 10 | 8 | k-fold cross-validation |
| [182] | IMU | CNN | Spectral analysis was applied to inertial signals to generate time–frequency representations, which were then classified using CNNs for human activity recognition. | 0.99 | CWP | 6 | 30 | k-fold cross-validation |
| [183] | IMU | CNN | A residual CNN architecture was proposed for locomotion mode recognition in lower-limb exoskeletons, eliminating the need for manual feature extraction. | 0.97 | CWP | 5 (LG, SA, SD, RA, RD) | 5 | Leave-one-subject-out |
| [184] | EMG + IMU, * early fusion | CNN | A multi-channel encoder CNN was designed to improve locomotion intention recognition by fusing EMG and IMU data with spectral feature extraction. | 0.94 | ENABL3S | 5 (LG, RA, RD, SA, SD) | 10 | k-fold cross-validation |
| [185] | EMG | CNN | High-density surface EMG data were used to train a CNN that demonstrated robustness against electrode shifts, improving reliability for real-time assistive control. | 0.97 | HDsEMG | 6 | 7 | k-fold cross-validation |
| [180] | IMU | CNN–SVM | A hybrid CNN–SVM model combined deep feature extraction with a finite-state machine to correct classification errors and enhance generalization. | 0.98 | CWP | 5 (LW, SA, SD, RA, RD) | 10 | Leave-one-subject-out |
| [188] | IMU | CNN | A deep CNN was developed for locomotion intent prediction in powered prosthetic legs, with transfer learning applied to improve subject-independent performance. | 0.74 | ENABL3S | 5 (LG, SA, SD, RA, RD) | 9 | Hold-out |
| [187] | IMU | CNN | A subject-independent CNN-based locomotion classifier was proposed for hip exoskeletons using an open-source gait biomechanics dataset. | 0.93 | CAMARGO | 5 (LG, SA, SD, RA, RD) | 21 | Leave-one-terrain-out |
| [195] | EMG | CNN–LSTM, LSTM–CNN | Hybrid deep learning architectures were evaluated using EMG and robotic sensor data to classify locomotion modes across different network configurations. | 0.94–0.95 | CWP | 5 (LG, SA, SD, RA, RD) | 500 | Hold-out |
| [186] | IMU | Attention CNN–BiLSTM | An attention-based CNN–BiLSTM model was proposed to estimate lower-limb joint moments using a single IMU, with optimal performance achieved from shank-mounted sensors. | 0.85 | CAMARGO | 4 (LG, Ramp, Stair, Treadmill) | 19 | Hold-out |

## 3.2 Materials

**Dataset**

In this work, we utilize the CAMARGO dataset [192], a publicly available multimodal repository including sensor data from 21 subjects. Each participant was equipped with four IMUs (IMU type: Xsens MTw Awinda – a wireless 9-DOF (Degrees of Freedom) IMU, manufactured by Xsens Technologies B.V.). positioned on the trunk, thigh, shank, and foot, along with 11 EMG sensors (Type: Surface EMG by Delsys and model Trigno Wireless System) that monitored the activities of the gastrocnemius medialis, tibialis anterior, soleus, vastus medialis, vastus lateralis, rectus femoris, biceps femoris, semitendinosus, gracilis, gluteus medius, and right external oblique muscles. The participants completed multiple trials across five locomotion modes: level ground walking, ramp ascent and descent, and stair ascent and descent. Stair trials were performed at four different heights (102, 127, 152, 178), while ramp trials included six inclination angles (5.2, 7.8, 9.2, 11, 12.4, and 18). In particular, the sensors are positioned along the body as shown in Figure 3.1, We can see that there are eleven Electromyography (EMG) sensors on the right side, targetting major lower limb muscles and four Inertial Measurement Units (IMUs) attached to the torso, thigh, shank and foot, capturing acceleration and angular velocity, For a total of 35 signals recorded over an estimated 40 minutes, capturing high-resolution IMU and EMG data across various locomotion conditions. For a more detailed description of a dataset and the exact sensor placements, please refer to the original paper [192]

The authors of the dataset preprocessed the collected dataset: whereas the IMU data, comprising acceleration and angular velocity from the three-axis accelerometer and gyroscope sampled at 200 Hz, were processed using a lowpass filter with a 100Hz cutoff frequency (Butterworth order 6), the raw EMG data, sampled at 1000 Hz, was digitally conditioned using a bandpass filter with a cutoff frequency of 20 Hz to 400 Hz (Butterworth order 20).

In our work, we use the data as described so far, utilizing a rolling window approach to segment the data into non overlapping temporal windows of 500 ms to generate approximately 1450 samples for each subject with a class probability of $20 \pm 2.5\%$ (the number of samples and the class probability depend slightly on the subject) to feed the different NN models.

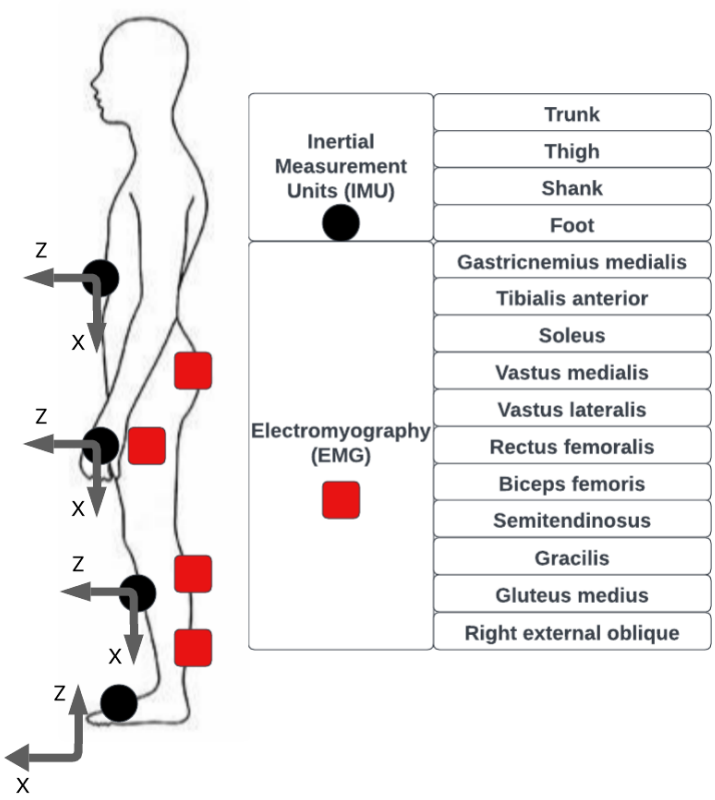


Figure 3.1: Position of the sensors along the body (Image extracted [192])

## 3.3 Methods

This section outlines the methodology for developing a system architecture to classify terrain types and estimate terrain parameters using multimodal sensory data. The system comprises two separate stages: a classification network for detecting the five terrain types and two regression models to estimate the slope of a ramp or the height of a stair. We compared state-of-the-art Deep Learning (DL)-based models for time series analysis to identify the optimal configuration. We selected the best performing sensor modality and model architecture based on the results. Additionally, we used Explainable Artificial Intelligence (XAI) techniques to determine the most influential features, providing insights into sensor importance and the minimal sensor setup required for accurate classification and regression as shown in Figure 3.2.

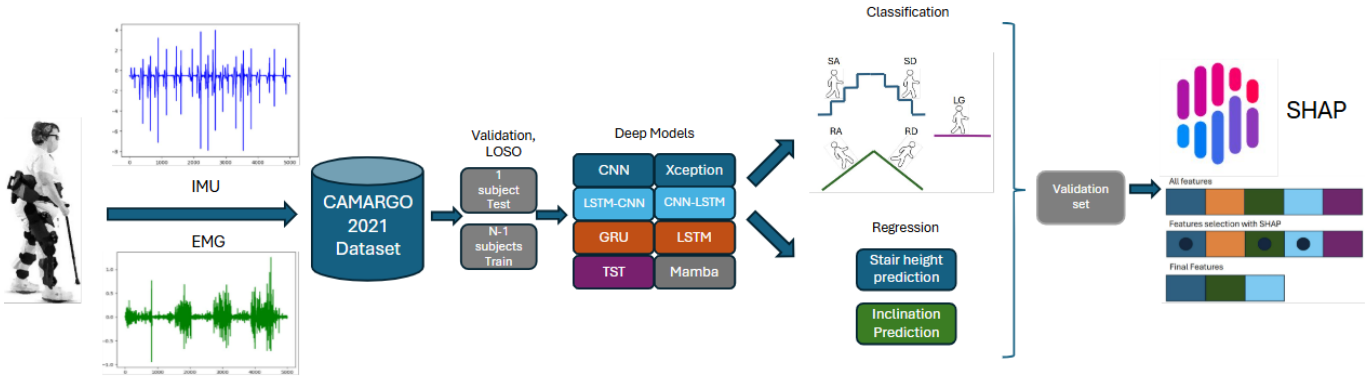


Figure 3.2: Visual Methodology

### Experimental Set Up

We validated all experiments using the Leave-One-Subject-Out (LOSO) cross-validation method, conducting a number of runs equal to the number of subjects in our study 21. In each run, the training set included data from $N-1$ subjects, while the test set consisted of data from the remaining subject. Although each model was allowed to train for up to 100 epochs, we applied an early stopping criterion that halted training after just 10 epochs, as the validation loss showed no improvement during that period. Essentially, by the time we reached 10 epochs, the network had plateaued in terms of performance, triggering the early stopping condition and preventing any further, unproductive epochs. All experiments and model training were performed on Google Colab Pro, utilizing 52 GB of RAM (CPU) and a 15 GB GPU, except for Mamba that was trained on an NVIDIA A100 GPU with 80Gb of Ram resources. For the classification task, we assessed performance using accuracy, precision, recall, F1 score, given the balanced nature of the dataset. However, for regression tasks, we evaluated performance using the Mean Absolute Error (MAE). To further assess model performance under different conditions, we performed the Wilcoxon signed rank test, a nonparametric statistical hypothesis test [196], and applied the Bonferroni correction to address the issue of multiple comparisons [197].

## 3.4 Result and Discussion

Within this section will be presented result and Discussion took from the paper [198]

### Performance Indicators

In this section, we present the main results of the experiments conducted. The primary metric used for evaluating classification performance is accuracy, used because the a priori class distribution is balanced. However, additional metrics, including recall and the F1 score, are reported in the Supplementary Material. Furthermore, for the regression analysis, we employed the Mean Absolute Error (MAE) as the primary metric.

Additionally, to determine the contribution of each sensor to the model's predictions, we computed the SHAP (SHapley Additive exPlanations) values [199], enabling an in-depth assessment of sensor importance and model interpretability. This analysis provided insights into the relative influence of each sensor on the system performance [200].

### Best performing Architecture

Our first analysis compares the performances of the eight NNs selected, both in the case of classification and regression tasks which are reported in Table 3.2 across different sensor combinations. The combined results indicate that there is a statistically significant difference between the performance achieved with Inertial Measurement Unit sensors and that with electromyography (EMG) sensors ($p = 3.35{*}10^{-5}, p = 2.20{*}10^{-3}, p = 2.98{*}10^{-5}, p = 8.32{*}10^{-4}$). Across all performance metrics—accuracy, precision, recall, and F1 score—the IMU sensor configuration outperforms the EMG sensor configuration. For example, architectures such as CNN and LSTM show higher values when using IMU data, which likely reflects the robustness of the kinematic information (such as acceleration and angular velocity) captured by these sensors. In contrast, EMG signals, which measure muscle activation, tend to be more susceptible to variability due to factors like electrode placement, skin impedance, and signal cross-talk, leading to a noisier performance profile. Moreover, the standard deviations observed with EMG data is generally higher, reinforcing the notion that its measurements are less reliable under the conditions tested. When comparing the IMU only setting to the combined IMU+EMG setting, the differences in performance are negligible across all metrics. This lack of statistically significant improvement suggests that the inclusion of EMG data does not contribute meaningful additional information beyond what is already captured by the IMU sensors. The inertial data appears to encapsulate the essential dynamic features needed for accurate classification, thereby rendering the additional complexity of EMG integration unnecessary. It is also possible that the nature of the movements being classified is such that the gross motion captured by the IMU is sufficiently discriminative, reducing the potential benefits of fusing EMG data. The redundancy in the information provided by the EMG signals may not offer any extra value in this specific scenario, especially when

considering the potential for increased noise and processing overhead. In summary, these results support the conclusion that the IMU sensors are adequate for the intended application, providing robust performance without the need for additional EMG data. The LSTM architecture outperformed the other models across all evaluated performance metrics. In particular, for both IMU and IMU+EMG configurations, the LSTM achieved higher accuracy, precision, recall, and F1 scores compared to the second best performing architecture (blue in table). The computed p-value ($p = 0.0015$, $p = 0.0018$, $p = 0.0014$, $p = 0.0016$) confirms that this improvement is statistically significant and not merely a result of random variation. This robust performance can be attributed to the LSTM's inherent ability to capture long-term dependencies in sequential data. Its memory cells effectively filter noise and model temporal dynamics, which is especially important in handling the complex patterns found in sensor signals.

Moreover, the LSTM's architecture allows it to generalize better by retaining relevant contextual information over time, a feature that is crucial for accurately classifying time series data. This capability gives it an edge over architectures such as CNNs, which are generally better at spatial feature extraction but may fall short in capturing temporal correlations. The lower variability in performance metrics for the LSTM further supports the idea that its superiority is intrinsic to its design rather than due to chance. Consequently, the statistically significant improvement, as evidenced by the p-value, highlights the robustness and effectiveness of the LSTM for this classification task.

Table 3.2: Classification Results: Each cell reports the average score followed by standard deviation. In black bold and blue bold the best and second-best results, respectively.

| | **Accuracy** | | | **Precision** | | | **Recall** | | | **F1 Score** | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Architecture** | **IMU** | **EMG** | **IMU + EMG** | **IMU** | **EMG** | **IMU + EMG** | **IMU** | **EMG** | **IMU + EMG** | **IMU** | **EMG** | **IMU + EMG** |
| CNN | $0.92 \pm 0.05$ | $0.73 \pm 0.10$ | $0.92 \pm 0.04$ | $0.92 \pm 0.03$ | $0.79 \pm 0.05$ | $0.92 \pm 0.06$ | $0.89 \pm 0.08$ | $0.73 \pm 0.04$ | $0.91 \pm 0.06$ | $0.89 \pm 0.05$ | $0.74 \pm 0.07$ | $0.91 \pm 0.06$ |
| LSTM | $\mathbf{0.94 \pm 0.04}$ | $0.60 \pm 0.06$ | $\mathbf{0.94 \pm 0.03}$ | $\mathbf{0.95 \pm 0.04}$ | $0.39 \pm 0.07$ | $\mathbf{0.95 \pm 0.05}$ | $\mathbf{0.94 \pm 0.06}$ | $0.60 \pm 0.07$ | $\mathbf{0.94 \pm 0.08}$ | $\mathbf{0.94 \pm 0.05}$ | $0.46 \pm 0.07$ | $\mathbf{0.94 \pm 0.06}$ |
| CNN-LSTM | $0.90 \pm 0.05$ | $0.75 \pm 0.08$ | $0.93 \pm 0.04$ | $0.93 \pm 0.04$ | $\mathbf{0.80 \pm 0.04}$ | $0.93 \pm 0.07$ | $0.92 \pm 0.05$ | $\mathbf{0.75 \pm 0.03}$ | $0.90 \pm 0.07$ | $0.92 \pm 0.08$ | $\mathbf{0.76 \pm 0.03}$ | $0.91 \pm 0.07$ |
| LSTM-CNN | $0.91 \pm 0.05$ | $0.75 \pm 0.10$ | $0.91 \pm 0.04$ | $0.93 \pm 0.05$ | $0.78 \pm 0.04$ | $0.92 \pm 0.08$ | $0.91 \pm 0.03$ | $0.75 \pm 0.09$ | $0.90 \pm 0.06$ | $0.91 \pm 0.04$ | $0.76 \pm 0.06$ | $0.90 \pm 0.05$ |
| GRU | $0.78 \pm 0.08$ | $0.61 \pm 0.03$ | $0.79 \pm 0.07$ | $0.78 \pm 0.07$ | $0.61 \pm 0.02$ | $0.78 \pm 0.06$ | $0.77 \pm 0.08$ | $0.60 \pm 0.03$ | $0.77 \pm 0.06$ | $0.79 \pm 0.07$ | $0.60 \pm 0.03$ | $0.79 \pm 0.10$ |
| TST | $0.92 \pm 0.04$ | $0.74 \pm 0.03$ | $0.91 \pm 0.05$ | $0.89 \pm 0.08$ | $0.60 \pm 0.03$ | $0.89 \pm 0.06$ | $0.92 \pm 0.04$ | $0.74 \pm 0.02$ | $0.91 \pm 0.04$ | $0.92 \pm 0.05$ | $0.74 \pm 0.03$ | $0.91 \pm 0.05$ |
| XceptionTime | $0.88 \pm 0.09$ | $0.73 \pm 0.03$ | $0.89 \pm 0.09$ | $0.88 \pm 0.09$ | $0.73 \pm 0.03$ | $0.88 \pm 0.01$ | $0.90 \pm 0.05$ | $0.74 \pm 0.03$ | $0.89 \pm 0.05$ | $0.88 \pm 0.09$ | $0.73 \pm 0.03$ | $0.88 \pm 0.09$ |
| MAMBA | $0.92 \pm 0.04$ | $\mathbf{0.76 \pm 0.05}$ | $0.91 \pm 0.06$ | $0.92 \pm 0.05$ | $0.75 \pm 0.05$ | $0.91 \pm 0.05$ | $0.89 \pm 0.05$ | $0.73 \pm 0.05$ | $0.91 \pm 0.05$ | $0.89 \pm 0.05$ | $0.75 \pm 0.05$ | $0.92 \pm 0.05$ |
| Mean | $0.89 \pm 0.05$ | $0.71 \pm 0.06$ | $0.90 \pm 0.05$ | $0.90 \pm 0.05$ | $0.70 \pm 0.05$ | $0.90 \pm 0.05$ | $0.89 \pm 0.06$ | $0.71 \pm 0.06$ | $0.89 \pm 0.07$ | $0.89 \pm 0.06$ | $0.78 \pm 0.05$ | $0.89 \pm 0.07$ |

The average training time of the neural network architecture is about 21.25 second per epochs, with a range going from $11.42 \pm 0.15$ to $55.19 \pm 0.57$. On the other hand, all models demonstrated an inference time of 1 over a 500 lookback window.

In summary, our analysis demonstrates that IMU sensors outperform EMG sensors, likely due to their robust capture of dynamic kinematic information. Furthermore, the negligible differences between the IMU only and IMU+EMG configurations indicate that adding EMG data does not provide a significant advantage. Among the eight architectures evaluated, the LSTM emerged as the top performer. For this reason, the regression analysis was conducted

solely with IMU data, as the architectures showed no benefit when incorporating EMG signals, aligning with our aim to minimize the sensor setup.

Moreover, Table 3.3 The table demonstrates that the LSTM architecture achieves an MAE of for slope prediction, which is not only the lowest among all architectures but also statistically significantly different from the second-best performer (in blue in table). This significance confirmed by computed p-values ($p = 0.015$) well below the standard threshold—indicates that the improvement is not due to chance. The LSTM's ability to capture long-term temporal dependencies and effectively filter noise in sequential data likely underpins this robust performance. Consequently, its lower error is a direct result of its architectural strengths rather than random variation.

Table 3.3: Regression Analysis of Slope Inclination: MAE Performance.In black bold and blue bold the best and second-best results, respectively

| **Architecture** | **MAE [°] (Slope)** |
|---|---|
| CNN | $2.21 \pm 0.58$ |
| LSTM | $\mathbf{1.93 \pm 0.53}$ |
| CNN-LSTM | $2.10 \pm 0.68$ |
| LSTM-CNN | $2.14 \pm 0.73$ |
| GRU | $2.32 \pm 0.61$ |
| TST | $2.21 \pm 0.60$ |
| XceptionTime | $2.08 \pm 0.54$ |
| Mamba | $2.03 \pm 0.46$ |
| Mean | $2.12 \pm 0.59$ |

On average, the training process takes $24.00 \pm 0.33$ seconds per epoch for slope prediction ranging from $13.57 \pm 0.17$ to $64.34 \pm 0.23$. The inference time remains consistent across all models for both classification and regression tasks, averaging 1 ms.

Lastly, the table Table 3.4 indicates that the CNN-LSTM architecture achieves the lowest MAE for stair height prediction, at $15.65 \pm 6.33mm$, outperforming the other architectures. In particular, its performance is statistically significantly better than that of the second-best architecture (LSTM, which has an MAE of $15.97 \pm 6.29mm$), as confirmed by the computed p-values ($p = 0.32$). This robust improvement is likely due to the CNN-LSTM's ability to combine the spatial feature extraction capabilities of CNNs with the temporal modeling strengths of LSTMs. By integrating these two approaches, the network is better equipped to capture the complex patterns and variations in stair height measurements, leading to a more accurate regression outcome that is not attributable to random chance. On average, the training process takes $20.14 \pm 0.33$ seconds per epoch for slope prediction ranging from $10.09 \pm 0.36$ to $58.45 \pm 0.36$. The inference time remains consistent across all models for both classification and regression tasks, averaging 1 ms.

Table 3.4: Regression Analysis of Stair Height: MAE Performance. In black bold and blue bold the best and second-best results, respectively

| **Architecture** | **MAE [mm] (Height)** |
|---|---|
| CNN | $18.00 \pm 8.51$ |
| LSTM | $15.97 \pm 6.29$ |
| CNN-LSTM | $\mathbf{15.65 \pm 6.33}$ |
| LSTM-CNN | $16.05 \pm 7.40$ |
| GRU | $21.10 \pm 9.11$ |
| TST | $17.56 \pm 6.65$ |
| XceptionTime | $17.30 \pm 7.57$ |
| Mamba | $16.02 \pm 6.89$ |
| Mean | $17.20 \pm 7.34$ |

### Most Informative Sensor

In this section, we delve deeper into the analysis of the results, focusing on identifying the most informative sensors using the SHAP methodology. Building on our previous findings, we conduct this analysis exclusively with IMU data and the LSTM and CNN-LSTM model to enhance interpretability and insight.

Figure 3.3 shows how much each sensor influences the ground type prediction where f, s, t, and k represent the foot, shank, thigh, and trunk, respectively. Additionally, the superscript denotes the axis (x, y, or z), while the subscript indicates the sensor type: G for the gyroscope and A for the accelerometer. The SHAP analysis indicates that foot sensors have the greatest influence on ground type prediction, followed by those on the thigh, shank, and trunk. This ranking aligns closely with the biomechanics of human locomotion and how different body segments interact with the ground.

It is evident that the foot-mounted IMU provides the most informative kinematic data for terrain classification, as the foot undergoes the most significant motion variations during locomotion. Being the primary point of contact with the ground, it is logical to deduce that the foot experiences changes in acceleration and angular velocity that directly reflect terrain properties [201]. Step transitions, foot placement, and ankle dynamics induce distinctive kinematic signatures, which are more pronounced than those captured by sensors positioned elsewhere on the body. Consequently, a Foot-mounted IMU is capable of detecting terrain-induced variations more effectively, making them the optimal choice for extracting kinematic features relevant to surface characterization.

IMU sensors positioned on the shank and thigh are capable of capturing key kinematic adaptations to terrain changes, though their contribution is less pronounced than foot-mounted sensors. Variations in surface incline or height have been shown to influence knee flexion, stride length, and hip motion, leading to distinct acceleration and angular ve-

locity patterns. These segments have been demonstrated to play a role in adjusting limb trajectories and modulating propulsion, particularly during slope ascent, descent, or stair negotiation.

In contrast, trunk mounted sensors provide the least informative kinematic data for terrain classification. While the trunk stabilizes movement and reacts to lower limb dynamics, its motion is less directly affected by ground properties. As a result, kinematic variations measured at the trunk are less pronounced, leading to lower relevance for identifying terrain characteristics compared to sensors positioned on the foot and lower limbs.

The observed sensor hierarchy is consistent with how humans adjust their movement to different terrains. Foot sensors capture direct interactions with the ground, thus providing the most informative data, while the shank and thigh assist in adjusting movement. The trunk plays a more supportive role in stabilization than in direct terrain response, which leads to its lower contribution to ground prediction.

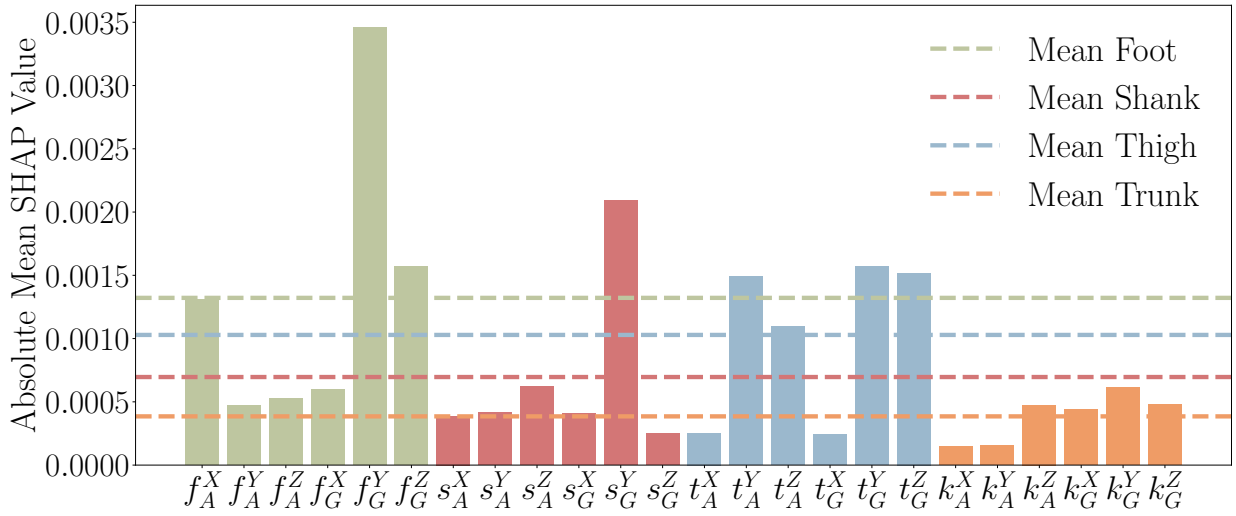


Figure 3.3: Feature importance for the IMU modality, derived by explaining the LSTM model used for the classification task (Level Ground, Stair Ascent, Stair Descent, Ramp Ascent, Ramp Descent) with the SHAP methodology. The labels x-axis are formatted as $B_S^A$, where $B$ denotes the body part (foot: $f$, shank: $s$, thigh: $t$, trunk: $k$), $S$ indicates the sensor type (accelerometer: $A$, gyroscope: $G$), and $A$ specifies the axis (x-axis: $X$, y-axis: $Y$, z-axis: $Z$). The dotted horizontal lines represent the average sensor weights.

Focusing on the two regression tasks, Figure 3.4 adopts the same notation and shows the results of the SHAP analysis both for slope prediction using LSTM (upper panel) and for stair height prediction using the CNN-LSTM model (lower panel). Both plots confirm that foot sensors are the most influential of these two regressions, with the shank and thigh sensors following closely.

As expected, the trunk sensor contributed the least. These insights underscore the critical role of foot sensors in capturing essential gyroscope and acceleration data during gait [202]. Also in this case, analyzing each sensor's features, we see that the most influential is the gyroscope on the y-axis of the foot. In second place is the thigh, where all sensors contribute equally except for the accelerometer on the x-axis. For what concerns the Shank the two most influential features are the acceleration on the z-axis and the gyroscope on the y-axis. The results indicate that the gyroscope on the y-axis of the foot is the most influential feature for predicting stair height and slope inclination, which is quite expected. This makes sense because the y-axis of the foot gyroscope captures rotational movement in the sagittal plane, which directly correlates with stair-climbing movements and slope changes. For the thigh, the fact that all sensors contribute equally—except for the accelerometer on the x-axis—is also reasonable. The thigh experiences both rotational and linear movements during stair ascent and descent, making all sensor modalities relevant. The lower influence of the x-axis accelerometer suggests that lateral movements of the thigh are less critical in determining stair height or slope. Regarding the shank, the dominance of the z-axis acceleration and the y-axis gyroscope aligns with expectations. The z-axis acceleration captures vertical displacement and impact forces, which are crucial for stair-related tasks. Meanwhile, the y-axis gyroscope reflects rotational motion in the sagittal plane, which is heavily involved in the adaptation to different stair heights and inclinations. These results align well with the kinematics of stair negotiation, where foot and shank dynamics play a critical role in adapting to slope and step height, while thigh movement remains more evenly distributed across sensors. The findings reinforce the importance of gyroscopic data, particularly along the y-axis, in capturing stair-climbing mechanics.

The previous analysis led us to reassess our sensor setup to simplify it without compromising performance. We evaluated four sensor combinations, starting with one IMU sensor and progressively adding sensors placed on different body districts based on their SHAP-derived importance. This iterative retraining process allowed us to assess the impact of each sensor combination on classification and regression performance.

Figure 3.5 illustrates the classification accuracy of the LSTM model as a function of the IMU sensors considered. The results reveal a statistically significant difference between setups with four sensors and those with one or two sensors. However, there was no significant difference between using three sensors versus four, suggesting that the trunk sensor does not substantially contribute to system functionality. Consequently, reducing the sensor setup to three sensors (foot, thigh, and shank) does not result in a significant performance drop, offering a more practical and cost-effective configuration.

Similarly, Figure 3.6 presents the MAE as a function of the IMU sensors, with sensors

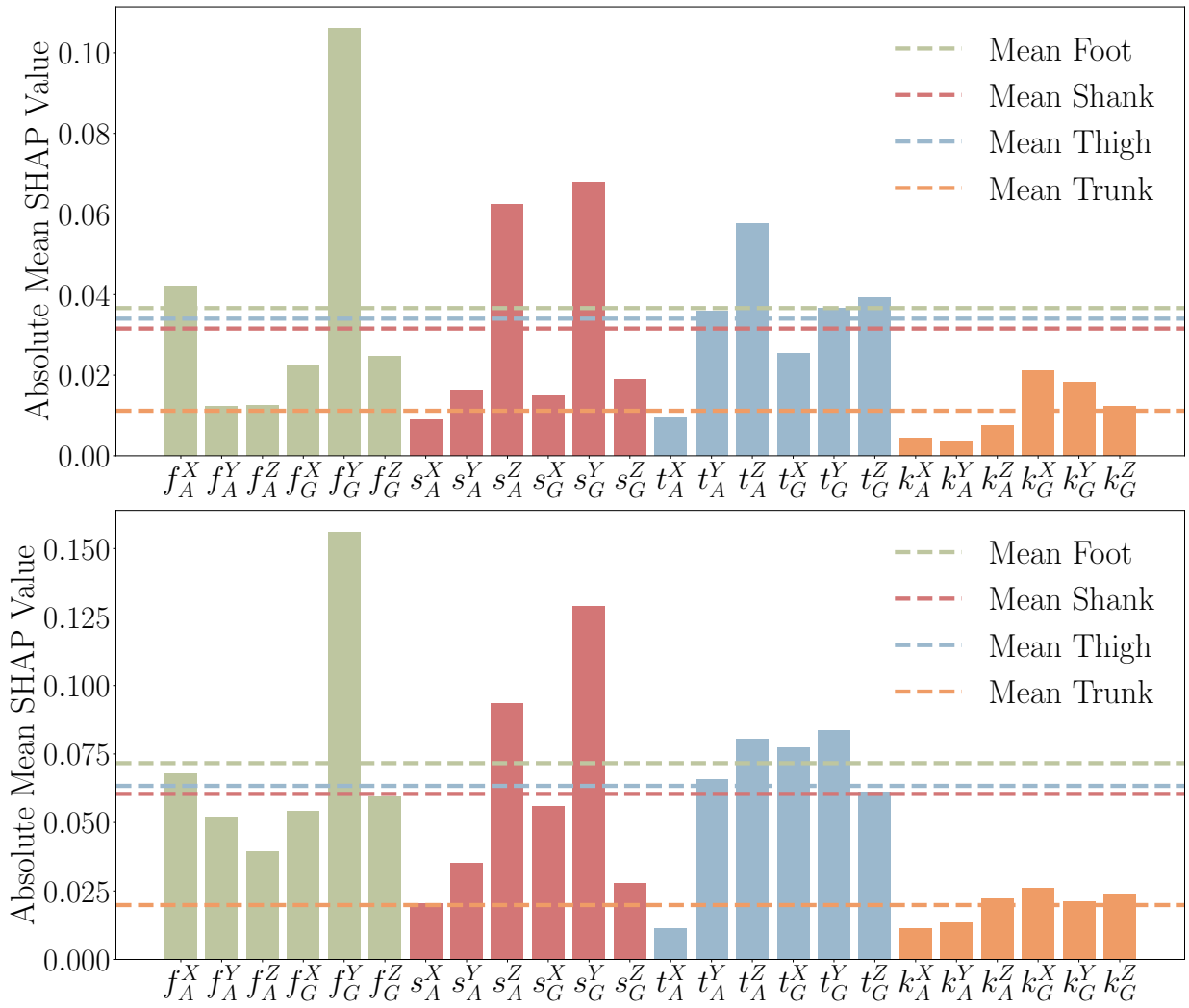


Figure 3.4: Feature importance for the IMU modality derived using the SHAP methodology, with the upper plot showing the LSTM model for slope prediction and the lower plot showing the CNN-LSTM model for stair height prediction, respectively. The features on x-axis are formatted as $B_S^A$, where $B$ denotes the body part (foot: $f$, shank: $s$, thigh: $t$, trunk: $k$), $S$ indicates the sensor type (accelerometer: $A$, gyroscope: $G$), and $A$ specifies the axis (x-axis: $X$, y-axis: $Y$, z-axis: $Z$). The horizontal lines represent the average sensor weights.

added iteratively based on their SHAP-derived importance for slope prediction (upper plot) and stair height prediction (lower plot). The findings indicate that excluding the trunk sensor did not significantly affect performance in either regression task, supporting the conclusion that we can exclude it from the setup.

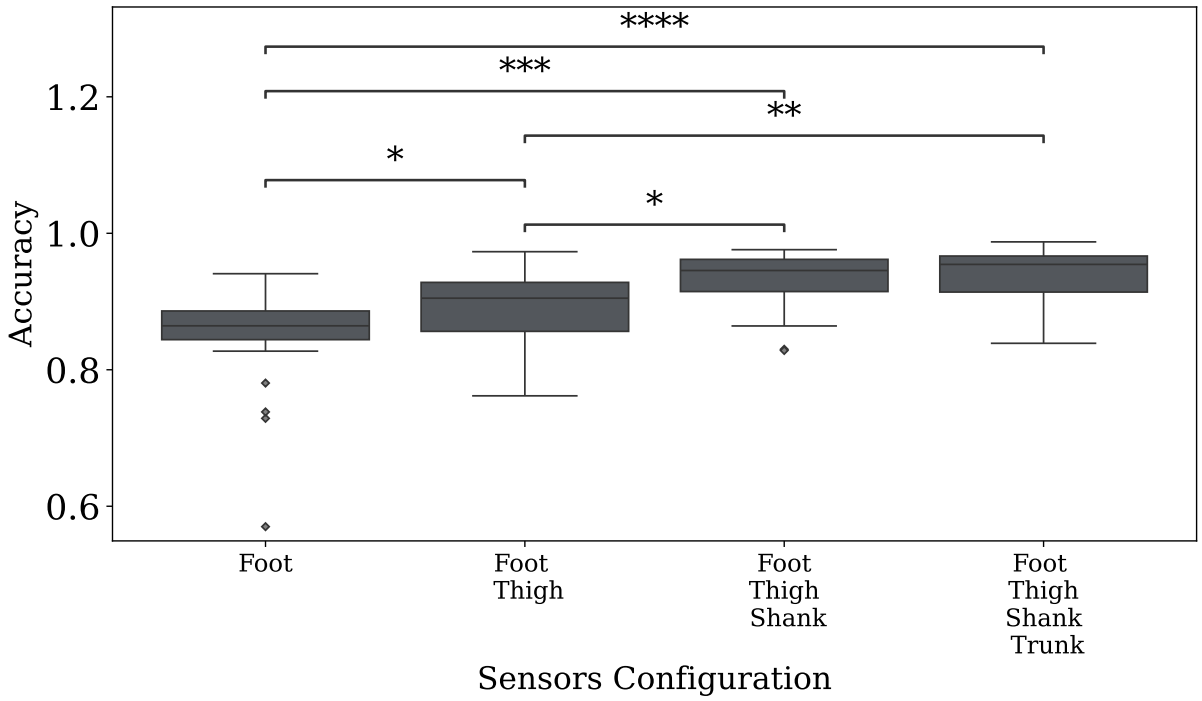


Figure 3.5: Accuracy of the LSTM model in the classification task as a function of the number of IMU sensors iteratively added based on their informativeness ranked by the SHAP methodology across the four body sectors. Statistical significance is denoted by asterisks, with * indicating $p \leq 0.05$, ** indicating $p \leq 0.01$, *** indicating $p \leq 0.001$, and **** indicating $p \leq 0.0001$.

**Discussion**

This study provides a comprehensive comparative analysis of deep learning models applied to human locomotion classification and terrain parameter estimation, particularly for lower limb robotic exoskeletons. The results demonstrate that different architectures excel in locomotion modelling with and accuracy higher than 0.90, with LSTM emerging as the most effective model. However, a hybrid CNN-LSTM approach proves superior for stair height regression, highlighting the nuanced relationship between spatial and temporal dependencies in movement analysis.

LSTM's strength lies in its ability to capture long-term dependencies in sequential data, a fundamental requirement for understanding locomotion patterns. Unlike CNN, which is highly effective at learning spatial representations but lacks an inherent mechanism for handling temporal relationships, LSTM processes sequences by maintaining a memory of past information through its gated architecture. These gates regulate the flow of information, allowing the network to retain relevant features while discarding noise, which is particularly

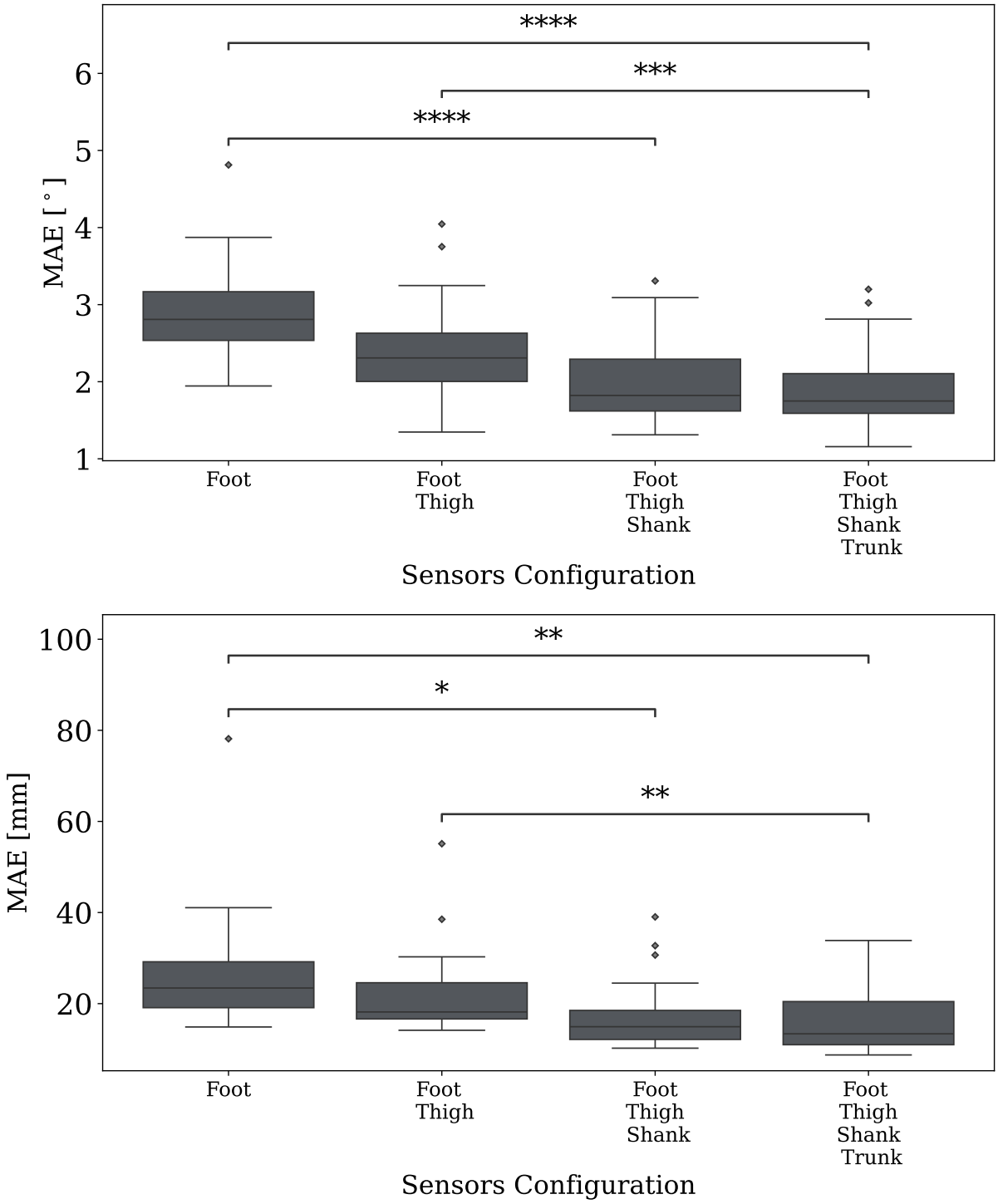


Figure 3.6: The MAE metric as a function of the number of IMU sensors iteratively added based on their informativeness ranked by the SHAP methodology across the four body sectors, with the upper plot showing slope prediction case and the lower plot showing stair height prediction case. Statistical significance is denoted by asterisks, with *** indicating $p \leq 0.001$, and **** indicating $p \leq 0.0001$.

useful in gait analysis where movements evolve dynamically over time. For ground-level classification, LSTM demonstrates the highest performance due to its ability to recognize movement transitions and maintain continuity in gait sequences, making it well-suited for distinguishing between different locomotion modes.

In contrast, stair height regression benefits more from the CNN-LSTM hybrid model, which integrates the advantages of both convolutional and recurrent architectures. Stair climbing involves distinct movement phases, including lifting, pushing off, and landing, where CNN effectively extracts localized features related to these abrupt transitions. By combining CNN with LSTM, the model retains the ability to track the step-by-step progression while also refining the height estimation through convolutional feature extraction. This hybrid approach ensures that both spatial variations and long-term dependencies are adequately modeled, leading to more accurate predictions of stair height compared to an LSTM-only approach.

For slope regression, however, LSTM again outperforms other models, suggesting that gradual changes in movement mechanics are best captured by recurrent architectures. Unlike stair climbing, where discrete transitions between steps occur, slope walking involves a continuous evolution of movement patterns. The recurrent design of LSTM allows it to process these gradual shifts effectively, preserving the relationship between past and present gait states without the constraints imposed by convolutional layers. While transformers, such as the Time Series Transformer (TST) and MAMBA, offer alternative paradigms for sequential modeling through self-attention mechanisms, their advantage is more pronounced in datasets with extremely long dependencies. In gait analysis, where temporal relationships are structured and periodic, LSTM remains a more efficient and reliable solution.

The study also examined the performance of MAMBA, XceptionTime, and GRU, which were found to be less effective in comparison to LSTM and CNN-LSTM. MAMBA, a state space model designed to handle sequential data without recurrence, struggles with locomotion analysis due to its reliance on a different computational framework that may not be as finely tuned to the biomechanical constraints of human movement. While MAMBA excels in handling general sequential patterns, it lacks the ability to preserve structured periodic dependencies crucial for gait modeling. XceptionTime, a convolution-based model adapted for time series tasks, faces inherent limitations due to its primary reliance on spatial feature extraction rather than the sequential nature of locomotion data. While it can learn feature hierarchies effectively, its inability to capture temporal transitions over time reduces its effectiveness in gait analysis. Similarly, GRU, despite being a recurrent model like LSTM, operates with a simpler gating mechanism that makes it computationally efficient but potentially less expressive in handling complex sequential patterns. GRU is effective at modeling

short-term dependencies but may not retain sufficient context for more nuanced motion patterns that evolve over longer time spans, which explains why LSTM remains preferable in this scenario.

Beyond model selection, the study also emphasizes the critical role of sensor configuration. Although it was initially expected that a multimodal approach incorporating both IMU and EMG data would yield superior performance due to the additional physiological information from muscle activity, the findings indicate otherwise. IMU data alone proves sufficient and even more effective, suggesting that the richness of acceleration and gyroscope signals in capturing movement dynamics outweighs the benefits of integrating EMG. The additional complexity introduced by EMG signals does not lead to a significant performance gain, reinforcing the idea that simpler sensor setups can be both practical and highly effective. Moreover, an analysis of feature importance using SHAP confirms that a minimal yet strategic sensor placement—covering the foot, shank, and thigh—achieves an optimal balance between accuracy and usability.

This study builds upon and extends the growing body of literature on terrain classification and gait parameter estimation using wearable sensor data. Compared to previous work summarized in Table 3.1, our approach offers several novel contributions and improved performance across multiple dimensions. First, unlike many prior studies, such as those by [178, 179] which primarily focus on unimodal approaches using either EMG or IMU signals, our work provides a comprehensive evaluation of both unimodal and multimodal setups. Importantly, we demonstrate that IMU only models not only match but often outperform multimodal configurations (IMU+EMG), offering a more practical and cost-efficient solution for real world robotic applications. Second, while previous efforts like [181, 184] have explored CNN-based classification models, they generally do not extend their analysis to continuous terrain parameters such as ramp inclination or stair height. In contrast, our study uniquely integrates both classification and regression tasks—using LSTM for slope prediction and CNN-LSTM for stair height estimation—thus enhancing adaptability in real time robotic control systems.

Third, our work improves upon methodological rigor. Several prior studies employ basic hold-out validation methods or do not clearly specify their data split strategies, as seen in [188]. This lack of standardization limits reproducibility and complicates the evaluation of model generalizability. By employing leave-one-subject-out (LOSO) cross-validation on the publicly available CAMARGO dataset [192], we ensure robust and interpretable performance evaluation, enabling more meaningful comparisons across studies. Moreover, our inclusion of Explainable AI (XAI) techniques via SHAP values enables a transparent evaluation of sensor importance, a feature rarely addressed in prior work. This allowed us to conduct an

ablation study to identify a minimal sensor configuration—foot, shank, and thigh—without significant loss in performance. This contrasts with most of the literature, which either uses large sensor arrays or does not explore sensor minimization strategies. In terms of performance, our best classification accuracy ($0.94 \pm 0.04$ with LSTM) is on par with or exceeds results from studies such as [182, 187], while our regression models achieve low MAE values for ramp slope ($1.93^\circ \pm 0.53$) and stair height (15.65 mm $\pm$ 6.33)—metrics that are rarely reported in the existing literature. Our study provides a more holistic, efficient, and reproducible framework for terrain-aware control in wearable robotics, addressing many of the limitations observed in existing literature.

The findings of this study carry significant implications for wearable robotic exoskeletons working in unstructured environments. The ability to classify ground conditions and estimate locomotion parameters with high accuracy and minimal latency would directly influence the adaptability, safety, and usability of such systems. In real world applications, exoskeletons should continuously interpret the environment and anticipate transitions to ensure seamless support, thereby preventing the user's destabilization or excessive cognitive burden. The ability to reliably distinguish between different locomotion contexts (e.g. level ground, ramps, stairs) would enable exoskeletons to preemptively adjust assistance strategies, optimizing the trajectories, the interaction control, and step timing. This is particularly crucial in rehabilitation, where precise adaptation to terrain variations can enhance training efficacy by encouraging more natural and effortful gait patterns. In the context of assistive applications for individuals with mobility impairments, real time ground adaptation reduces the need for manual intervention, thereby fostering greater autonomy and reducing user fatigue.

Furthermore, the accuracy of parameter estimation (e.g., stair height, ramp inclination) impacts the granularity of control adjustments. A mismatch between perceived and actual locomotion demands could lead to either overcompensation, resulting in unnatural and inefficient movements, or undercompensation, which may compromise stability. The accuracy obtained by the proposed approach suggests that fine-grained terrain adaptation is feasible, paving the way for exoskeletons that can seamlessly modulate assistance not only across different terrains but also in response to subtle environmental variations.

A further aspect that warrants discussion concerns the *timing* of the classification within the gait cycle, which has direct implications for the deployment of these models in real-time robotic control. In the present study, the classification pipeline operates on non-overlapping windows of 500 extracted continuously from the IMU streams, without explicit alignment to specific gait events such as heel strike or toe-off. As a consequence, the analysis quantifies the average classification performance across the gait cycle, but does not characterize how

this performance varies depending on the position of the sliding window within the cycle. This formulation reflects the typical operating regime of a wearable controller, which must produce predictions continuously and at fixed intervals, but it does not provide explicit information about when within a stride the terrain transition becomes reliably detectable.

This aspect is particularly relevant for exoskeleton control, where the latency between a terrain transition and its correct identification determines whether the assistance can be adapted in time to support the upcoming step. An ideal classifier would not only achieve high accuracy, but also recognize transitions sufficiently early in the gait cycle ideally during the swing phase preceding ground contact so that joint torques and trajectories can be modulated proactively. Conversely, a classifier whose accuracy is concentrated late in the cycle, for instance after heel strike on the new terrain, would offer limited benefit for predictive control, even if its average accuracy is high.

The CAMARGO dataset [192] provides gait-phase annotations that, in principle, allow this temporal characterization to be performed by stratifying classification accuracy according to the gait phase in which each window is centered, and by quantifying the delay between a terrain transition and its first correct detection. Such an analysis was not included in the present study, which focused on the comparative evaluation of architectures and sensor configurations under a uniform protocol. However, the inference times reported here approximately 1 per window across all evaluated architectures are well below the duration of a single gait cycle, indicating that the proposed models are computationally compatible with phase-resolved, real-time deployment. A systematic analysis of detection timing across the gait cycle, together with the quantification of transition-to-detection latency, is therefore identified as a natural and necessary extension of this work, and represents a key step toward validating the proposed framework in closed-loop exoskeleton control scenarios.

The findings of this study serve to reinforce the critical role of real time perception in the domain of wearable robotics. The reliable classification of terrain and the estimation of locomotion parameters are not merely technical benchmarks; they are fundamental enablers of more intuitive, responsive, and independent mobility assistance, thereby bridging the gap between robotic exoskeletons and natural human movement.

## 3.5 Conclusion

In this chapter, a neural network–based framework for real time ground condition analysis in lower limb exoskeleton applications has been presented. The proposed approach combines three specialized models to estimate key locomotion parameters required for adaptive robotic control: an LSTM-based classifier for terrain recognition, a second LSTM model for

ramp slope estimation, and a hybrid CNN–LSTM architecture for stair height prediction. Together, these models enable a comprehensive interpretation of environmental conditions relevant to human locomotion.

A comparative analysis with existing approaches in the literature, conducted using the public CAMARGO dataset, guided the selection of the final architectures and sensing configuration. Explainability analyses based on SHAP were employed to assess sensor relevance and model decision-making processes, providing insight into how different body segments contribute to classification and regression tasks. This analysis supported the adoption of a minimal sensing setup based on three IMU sensors positioned on the foot, shank, and thigh, achieving a favorable balance between sensing complexity and predictive performance.

The results demonstrate that the proposed system can achieve high accuracy in terrain classification, as well as precise estimation of ramp slope and stair height, while maintaining inference times compatible with real time operation. Importantly, the explainability-driven sensor selection revealed that trunk mounted sensors contributed marginally to model performance, allowing their removal without degrading accuracy. This finding reinforces the feasibility of lightweight and wearable sensing configurations for exoskeleton-based applications.

Furthermore, this chapter challenges the common assumption that multimodal sensing necessarily improves performance. The experimental results indicate that IMU only configurations can outperform combined IMU and electromyography setups, highlighting the efficiency and robustness of inertial sensing for locomotion analysis. This observation has practical implications for the design of wearable robotic systems, where reducing sensor burden and system complexity is often critical.

The findings presented in this chapter contribute to the development of intelligent exoskeleton systems capable of low-latency and accurate terrain awareness, supporting adaptive control strategies in real world scenarios. While the current analysis focuses on data collected from healthy individuals, the proposed methodology is designed with future clinical translation in mind. Extending this framework to populations with neurological or musculoskeletal impairments represents a natural next step and will be essential for evaluating the generalization capabilities of the models and their applicability in assistive and rehabilitative robotic systems.

# Chapter 4

# Towards Understanding Self-Pretraining for Sequence Classification

**Statement on Publication Status**

The contents of this chapter are confidential, as they form part of a manuscript currently under peer review.

## 4.1 Introduction

**From Applied Locomotion Analysis to a Theoretical Study of Self-PreTraining**

The previous chapters established both the landscape and the practical performance of AI architectures for lower limb rehabilitation robotics. Chapter 2 systematized the state of the art, identifying recurring methodological gaps most notably the limited size of clinical datasets, the high inter-subject variability of gait patterns, and the difficulty of generalizing models across walking conditions and populations. Chapter 3 then provided an empirical comparative analysis of modern neural architectures on multimodal locomotion data, showing that competitive performance can be achieved with reduced sensor configurations and that Transformer-based models, despite their theoretical advantages in capturing long-range dependencies, do not consistently outperform simpler recurrent or convolutional baselines in this domain.

This last observation is not incidental. It reflects a well-documented phenomenon in sequence modeling: Transformers tend to underperform on small and moderately sized datasets, where their high capacity and weak inductive biases hinder generalization. In rehabilitation robotics and more broadly in medical time series this limitation is particularly

consequential, because labeled data are intrinsically scarce, expensive to collect, and characterized by strong subject-specific variability. As a result, the most expressive architectures available today are also the most difficult to train reliably in the regimes where they would be most valuable.

A natural strategy to mitigate this limitation is *Self-PreTraining* (SPT), in which a model is first trained to reconstruct masked portions of its own input on the same downstream dataset, and is subsequently fine-tuned on the supervised task. Unlike large-scale self-supervised learning, SPT does not require external corpora and is therefore well suited to clinical settings where data cannot be aggregated across institutions. Empirically, SPT has been shown to improve downstream performance, but the *mechanisms* through which it does so particularly in Transformer-based sequence classifiers remain poorly understood. This gap is methodologically relevant: without a clear understanding of why SPT helps, it is difficult to design pretraining schemes that are robust, data-efficient, and transferable to new clinical tasks.

The present chapter addresses this gap. Stepping back from the application-driven analyses of Chapters 2 and 3, it adopts a more controlled and theoretical perspective, examining how SPT shapes the internal dynamics of attention-based models. Through toy tasks, ablation studies, and analyses of weight displacement, attention structure, and positional encoding, the chapter investigates whether the benefits of SPT stem from improved initialization of specific parameters, the emergence of structured attention patterns, or favorable changes in the optimization landscape. The findings developed here are subsequently leveraged in Chapter 5, where SPT is applied to locomotion recognition and to other medical time series tasks that share the same small-data, high-variability characteristics observed in lower limb rehabilitation robotics. In this sense, this chapter provides the methodological foundation needed to deploy Transformer-based architectures effectively in the clinical regimes targeted by this thesis.

### Introduction to Self-PreTraining

The long range Arena [203] has served great importance for the development of new efficient token mixing strategies over the last few years. A prime example is that of the first state space-model (SSM): S4 [204], that became popular for surpassing transformers on the LRA with 20% average increase in accuracy. S4 along with later variants also developed by benchmarking on the LRA [29, 43, 205, 206] serve as the basis for modern architectures such as Mamba [207, 208], GLA [209], RetNet [210], RWKV [53, 211] and DeltaNet [212] among others.

While the latest developments in token mixing mechanisms are mainly inspired by lan-

guage modeling performance [213], researchers maintained a strong interest in explaining the gap between transformers and SSMs on LRA (e.g. [214]). In their brilliant contribution, awarded the *Outstanding Paper Award at ICLR 2024*, [56] showed that training transformer models from scratch (as done in the LRA benchmark) leads to an under-estimation of their performance and demonstrates that dramatic gains can be achieved with a pretraining → finetuning setup (see [56]). At a first glance, this finding may not seem too surprising yet, **pretraining is here performed on the same data** (self-pretraining, **SPT**), that is: the transformer is first tasked to learn to predict masked tokens (or the next token) in the sequence, and only later is trained on tasks labels. While this procedure differs substantially with the usual pretrainig paradigm (large pretraining corpus, small finetuning dataset, e.g. [63]), it draws conceptual similarities both with classical curriculum learning [215] and more recent applied research: among others, [69] showed the efficacy of self-pretraining in vision and more recently [216] showed that pretraining large language models directly on downstream datasets could often match pretraining on massive external data, highlighting that much of the observed *performance gains may be driven by the pretraining objective itself rather than the data volume*.

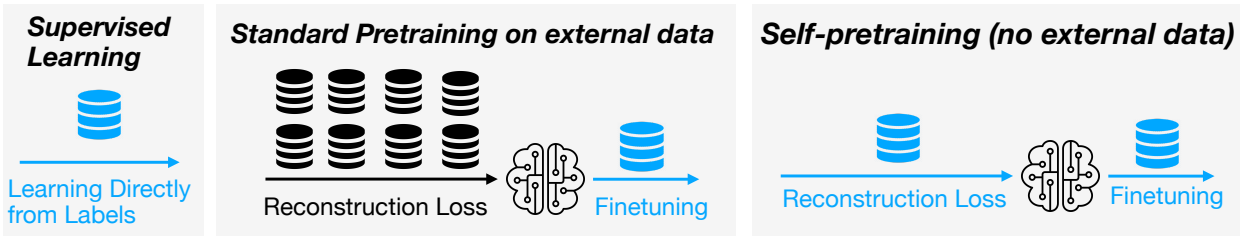


Figure 4.1: Illustration of different training pipelines for classification. SPT is the method by [56] that we study here.

## 4.2 Materials

**Datasets:**

**Long range arena**

We conduct experiments on a subset of the Long Range Arena (LRA) benchmark. Specifically, we consider the following sequence classification datasets: *ListOps*, *Text*, *Retrieval*, *Image* (grayscale CIFAR-10), and *Pathfinder*. All datasets are converted to one-dimensional token sequences following the LRA protocol. Sequence lengths range from 1,024 tokens (Image, Pathfinder) up to 4,096 tokens (Retrieval). The PathX dataset is excluded due to its high computational requirements.

### Tasks

- **ListOps:** Inspired by [217]. Sequences are nested lists describing mathematical operations applied to token sets. The task requires understanding hierarchical structures. The sequence length is 2048.
- **Text:** Character-level IMDb reviews [218] for binary sentiment classification. Sequences are up to 2048 tokens.
- **Retrieval:** Character-level binary classification of document similarity scores with sequences up to 4096 tokens. The task was introduced by [219].
- **Image:** Grayscale CIFAR-10 [220] images flattened into 1D sequences for 10-way classification without explicit 2D inductive bias, sequence length 1024.
- **Pathfinder, PathX:** Synthetic 2D visual tasks introduced by [221], treated as 1D sequences for tracing capabilities (sequence lengths 1024 and 16384, respectively). They are both binary classification tasks.

### Toy Dataset

We first describe the synthetic two dimensional binary time series classification dataset used in Section 4.4, and then present additional plots supporting our claims.

### Task Description

Each sample is a sequence $x = (x_j)_{j=0}^{L-1} \in R^{L\times 2}$ with $L = 100$ and a binary label $y \in \{0, 1\}$. With

$$a_0 = (0.66,\, 0.55), \quad b_0 = (-0.495,\, -0.605), \quad a_1 = (0.605,\, -0.55), \quad b_1 = (-0.55,\, 0.605).$$

**Conditioned on the binary label**, we define anchor points

$$(c_1, c_2) = \begin{cases} (a_0, b_0), & y = 0, \\ (a_1, b_1), & y = 1. \end{cases}$$

For each sequence, we sample independently

$$s \sim \mathcal{U}(0.12, 0.28), \qquad \phi \sim \mathcal{U}(0, 2\pi), \qquad \alpha \sim \mathcal{N}(1, 0.1^2).$$

Define normalized time $t_j = j/(L-1)$ and a "warped" time variable

$$\tilde{t}_j = t_j + s\sin(2\pi t_j)\big(0.5 + 0.5\cos(2\pi t_j)\big).$$

The mixing coefficient and mean trajectory are defined as

$$m_j = 0.5 + 0.35\sin(2\pi\tilde{t}_j + \phi), \qquad \mu_j = m_j c_1 + (1 - m_j)c_2.$$

Finally, a smooth oscillatory drift is added: $d_j = 0.30\big(\sin(2\pi\tilde{t}_j), \cos(2\pi\tilde{t}_j)\big)$. Observations are generated as

$$x_j = \alpha(\mu_j + d_j) + \varepsilon_j + \delta_j, \qquad \varepsilon_j \sim \mathcal{N}(0, \sigma^2 I_2),\ \sigma = 0.55.$$

To introduce outliers, sample $N_{\text{sp}} \sim \text{Unif}\{0,1,2,3\}$ and choose a random index set $I \subset \{0,\dots,L-1\}$ with $|I| = N_{\text{sp}}$. For $j \in I$, draw $\delta_j \sim \mathcal{N}(0, I_2)$; otherwise set $\delta_j = 0$.

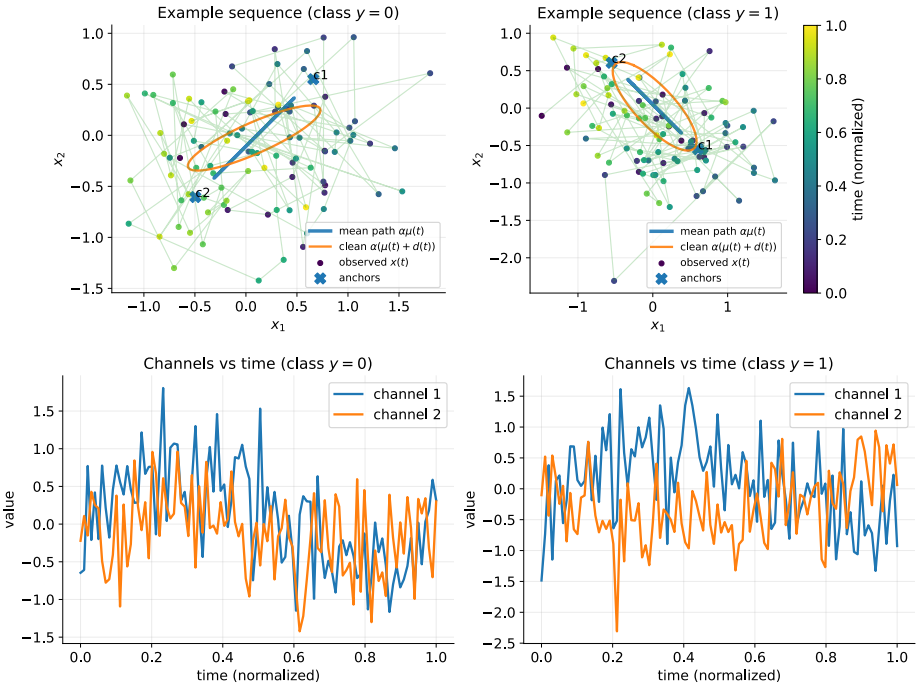


Figure 4.2: Illustration of the toy task for a random seed.

## 4.3 Methods

### Models

All experiments use transformer encoder architectures with bidirectional attention. Models employ rotary positional embeddings and standard multi-head self-attention blocks followed by feed-forward networks. Architectural configurations (embedding size, depth, number of heads, and feed-forward width) vary across datasets and closely follow the settings introduced by [56]. For ablation studies, we also consider shallow models with one to four transformer layers.

**Encoded Data.** The model is fed a batch of sequences

$$\mathcal{D} = \{(s_b, \ell_b)\}_{b=1}^{B}, \qquad s_b = (s_{b,1}, \ldots, s_{b,\ell_b})$$

with lengths $\ell_b \leq L$. First, tokens are mapped to vectors

$$X_{b,t} = E(s_{b,t}) \in R^D, \qquad X \in R^{B \times L \times D},$$

with padding for $t > \ell_b$. Next, we add positional information:

$$Z_{b,t}^{(0)} = X_{b,t} + P_t,$$

where

$$P_{t,2i} = \sin\Big(\frac{t}{10000^{2i/D}}\Big), \qquad P_{t,2i+1} = \cos\Big(\frac{t}{10000^{2i/D}}\Big),$$

Dropout is then applied:

$$Z^{(0)} = \mathrm{Drop}(Z^{(0)}).$$

**Prenorm backbone.** Let $T$ be the total number of residual blocks. For $t = 1, \ldots, T$, denoting standard LayerNorm by "Norm", one alternates multi-head Attention (MHA) with an MLP:

$$Z' = Z + \mathrm{Drop}\big(\mathrm{MHA}(\mathrm{Norm}(Z))\big),$$

$$Z^{+} = Z' + \mathrm{Drop}\big(\mathrm{FFN}(\mathrm{Norm}(Z'))\big),$$

where the position-wise MLP is

$$\mathrm{FFN}(x) = W_2\, \sigma(W_1 x + b_1) + b_2.$$

For $H$ heads, $d_k = D/H$, the MHA operation is:

$$Q = ZW_Q, \quad K = ZW_K, \quad V = ZW_V, \quad \mathrm{Att}(Z) = \mathrm{Concat}_h \left(\mathrm{softmax}\Big(\tfrac{QK^\top}{\sqrt{d_k}}\Big)V\right) W_O.$$

**Decoder (sequence reduction + readout).** A restriction operator $g$ produces

$$U = g(Z) \in R^{B\times \ell_{\text{out}} \times D}, \quad g \in \{\text{last, first, mean, sum, max, ragged}\}.$$

The final output is $Y = f(U)$, where $f$ is a linear map.

### Training Objectives

We compare two training paradigms: standard supervised training from scratch and a self-pretraining (SPT) pipeline. In SPT, models are first pretrained on the same dataset using a masked sequence modeling objective before being fine-tuned on task labels. Masking ratios follow prior work and vary by modality (e.g., higher masking for images, lower for symbolic data). Cross-entropy loss is used for token reconstruction in most datasets, while an L2 reconstruction loss is employed for image data.

### Optimization and Hyperparameters

Optimization is performed using AdamW with cosine learning rate scheduling. Learning rates, batch sizes, and weight decay values are dataset-specific and match those reported in the experimental appendix. fine tuning is performed for 100 epochs in all settings, while self-pretraining lasts up to 200 epochs depending on the ablation.

### Experimental Set Up

### Hardware and Software

All experiments are run on a single NVIDIA A100 GPU with 80GB of memory. Models are implemented in PyTorch, and experiments build upon the publicly available codebase released by Amos et al. (2023), with minor modifications to support additional ablations.

## 4.4 Results and Discussions

### Ablations

After our successful reproduction of the results by [56] in Table 4.1, we present here four set of ablations bringing insights into SPT mechanisms: varying pretraining duration (Section 4.4),

changing model depth and data source (Section 4.4), freezing layers when training from scratch (Section 4.4), and using only a subset of self-pretrained weights 4.4.

Table 4.1: Replication of the results of [56].

| **Training Modality** | ListOps | CIFAR10 | Pathfinder | Retrieval | Text |
|---|---|---|---|---|---|
| Standard (from scratch) | 0.402 | 0.499 | 0.67 | 0.776 | 0.653 |
| + Self-pretraining (Results from Amos et al.) | 0.59 +0.19 | 0.74 +0.24 | 0.88 +0.21 | 0.88 +0.11 | 0.89 +0.24 |
| + Self-pretraining (Our reproduction of Amos et al.) | 0.56 +0.16 | 0.72 +0.22 | 0.85 +0.18 | 0.88 +0.11 | 0.89 +0.24 |

### Self-pretraining Duration

In our experiments leading to Table 4.1, we noticed that pretraining can be computationally expensive[1]. While [56] present detailed ablations regarding dataset sizes, we are most curious of decreasing the epochs budget from 200 (baseline) down to a single epoch. Our results in Table 4.2 show a curious phenomenon: *only 10 epochs* of self-pretraining are needed to significantly boost performance on CIFAR10 and ListOps. Instead, for PathFinder, major gains can only be observed after 100 epochs. Additionally, we found that improvements can already be observed after a single epoch. For this experiment, our hyperparameters are kept the same as in Table 4.1, with the only difference in the 1-epoch setting where we reduced the learning rate warm-up settings to fit the reduced step budget.

Table 4.2: Self-pretraining epochs ablation. We first self-pretrain with $X$ number of epochs (full cosine annealing in that number of epochs). We then finetune for 100 epochs on each dataset.

| **Pretr. Len.** | ListOps | CIFAR10 | Pathfinder |
|---|---|---|---|
| 0 epochs | 0.403 | 0.499 | 0.67 |
| 1 epoch | 0.48+0.08 | 0.56+0.06 | 0.69+0.02 |
| 10 epochs | 0.55+0.15 | 0.71+0.21 | 0.71+0.04 |
| 50 epochs | 0.55+0.15 | 0.71+0.21 | 0.71+0.04 |
| 100 epochs | 0.55+0.15 | 0.71+0.21 | 0.73+0.06 |
| 150 epochs | 0.56+0.16 | 0.71+0.21 | 0.77+0.10 |
| 200 epochs | 0.56+0.16 | 0.72+0.22 | 0.85+0.18 |

[gray!8] **Takeaway 1:** Self-pretraining can yield substantial improvements even after a single epoch of pretraining.

[1]In CIFAR the wall clock time for 1 epoch of pretraining on our hardware (a single A100 with 80GB of RAM) is 48s, for ListOps is 19.1m and for Pathfinder is 3.20min. At finetuning, the time for 1 epoch in CIFAR is 56s, for ListOps is 18.95m and for Pathfinder is 3min. Since as [56] we first pretrain for 200 epochs and then finetune on 100, the total time compared to basic training from scratch for 100 epochs is tripled.

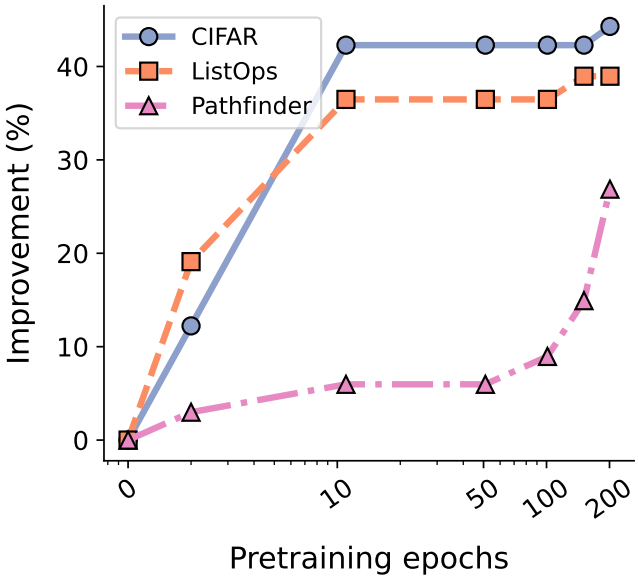


Figure 4.3: We found that on ListOps, CIFAR10, and PathFinder the relative finetuning performance except for pathfinder saturates after 10 self-pretraining epochs.

**Evidence for an optimization bottleneck (Depth + Dataset Swap)**

A natural explanation for the observed increase in test accuracy could be the ability of self-pretraining to learn crucial patterns and **hierarchical representations** of the data, providing a *better data-distribution-informed initialization* when learning labels [222, 223]. While in Section 4.4, we showed how such representations may arise with just a few epochs of self-pretraining, here we inspect the efficacy of self-pretraining as we vary the **number of layers** in the model - testing if the success of self-pretraining is contingent on the presence of a hierarchy in representations. At the same time, to test whether self-pretraining is indeed learning task-specific representations, we consider **swapping pretraining datasets**: we pretrain on Pathfinder data and fine-tune on CIFAR and vice versa.

For $A, B \in \{\text{CIFAR10, Pathfinder}\}$, we (1) train from scratch on data $A$, (2) pretrain on data $A$, fine-tune on data $A$, or (3) pretrain on data $A$, fine-tune on data $B$. In Fig. 4.4 we report both train and test accuracy after fine tuning.

1. We observe **gains even when pretraining on a single layer**. While test accuracy at finetuning is only improved by $\sim 2\%$ after self-pretraining a 1-layer model on Pathfinder, on CIFAR10 we observe a $\sim 15\%$ increase. Even though the gap in deeper models is higher, especially for Pathfinder, we find it interesting that it can be observed at such small scales.

2. On both datasets, **from-scratch test accuracies are almost independent of the number of layers**. Moreover, there is a substantial gap in training accuracy between

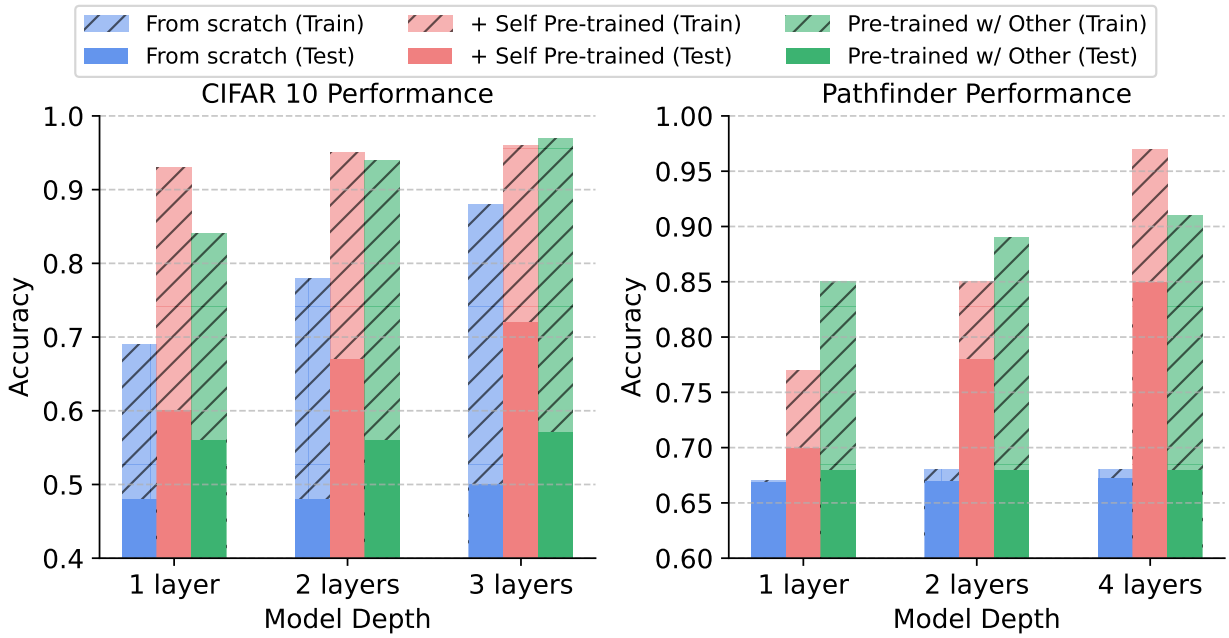


Figure 4.4: Settings are exactly the same as in Sec. 4.2 (200 pretraining epochs).

from-scratch and SPT performance, pointing to optimization issues.

3. Some gains in test accuracy after finetuning can be **observed when switching data source**. This also points to SPT targeting, to some degree, suboptimal initialization and unlocking high training accuracy.

[gray!8] **Takeaway 2:** The advantage of SPT is not necessarily related to generalization: it can greatly improve *training* accuracy after finetuning even when models are just 1-layer deep. This can also hold when pretraining on a completely different data distribution from another LRA task. At a deeper level our results hint at fundamental optimization problems when training Attention-based sequence models directly from labels.

### Freezing layers : From Scratch Sensitivity Analysis

In Section 4.4, we saw that short pretraining, either on task data or on data with a different distribution, can improve training accuracy (and sometimes, test accuracy) when learning from labels. Following the discussion from [56], we learn that models with a different sequence mixer (e.g. SSMs instead of Softmax Attention) can be less sensitive to self-pretraining. We therefore postulate that poor trainability in our setting may be linked to the Attention module's features. To this end, we consider a drastic experiment: training from scratch after freezing all Attention weights in the architecture. If attention is not effectively learned from labels, then freezing Attention at random initialization should have little effect on

performance. This finding would also be in agreement with insight coming from early-stage vision Transformers training [224], indicating that Transformers are data-hungry and may profit from additional supervision beyond hard labels.

In Table 4.3, we observe that (with the exception of Pathfinder) **training the Attention layer has little to no effect when training from-scratch**. Freezing learning in Attention layers can also improve performance, e.g. on the CIFAR10 task. This is surprising and indicates that Attention layers are unable to effectively integrate information when learning from labels. Instead, as expected (with the exception of the Text task) freezing also the intermediate MLP layers in the transformer block always decreases performance.

[gray!8] **Takeaway 3:** From-scratch performance is generally unaffected by freezing Attention weights to random initialization. This indicates the model struggles to learn how to combine sequential information directly from labels.

Table 4.3: Comparison of different **freezing strategies** when training deep models **from scratch**. Reported are test accuracy results, the setting is the same as in Section 4.2. We either clamp only Attention weights or Attention weights and intermediate feedforward layers to random initialization, and learn other parameters from labels. Encoder (linear), normalization weights, and decoder layers (linear) are always trained.

| **Dataset** | **From Scratch** | **with Attention frozen** | **with Attention & MLP frozen** |
|---|---|---|---|
| CIFAR-10 | 0.50 | 0.56 +0.06 | 0.43 -0.07 |
| Pathfinder | 0.67 | 0.50 -0.17 | 0.50 -0.17 |
| ListOps | 0.40 | 0.40 +0.00 | 0.30 -0.10 |
| Retrieval | 0.78 | 0.78 +0.00 | 0.67 -0.09 |
| Text | 0.65 | 0.65 +0.00 | 0.66 +0.01 |

### Using SPT initialization only on a subset of weights

Section 4.4 suggests that the gains from self-pretraining are not uniformly distributed across model components, with the Attention mechanism emerging as particularly important. Motivated by this observation, we study hybrid initializations in which, for each layer, only selected subsets of parameters are initialized from self-pretrained weights, while all remaining parameters are randomly initialized.

The results in Table 4.4 show a clear separation between architectural components. **Initializing only MLP-related parameters yields little to no improvement over random initialization** (see *MLP+Norm+Encoder*), indicating that self-pretraining provides limited transferable signal to the MLP branch. **In contrast, initializing the Attention**

Table 4.4: Same setup as Table 4.1. Please refer to Section 4.3 for our notation in this caption. For each dataset, models are initialized in a **hybrid fashion**: we consider, for all layers, initializing to self-pretrained values **only** (1) the Attention weights ($W_K, W_Q, W_V, W_O$) (2) the Attention weights, their LayerNorm parameters, and the initial embedding layer, ..., (5) Attention and MLP weights, and so on. All weights which are not initialized from self-pretraining are initially random. All weights are then being optimized with the same setting as Table 4.1. "No Init" and "Full Init" here correspond to From-Scratch and Self-Pretrained variant in Table, 4.1, respectively. We show classification accuracy, and notice that substantial gains from random initialization can be obtained by initializing to self-pretrained weight only the parameters in the Attention branch, while specialized initialization of parameters in the MLP branch has little to no effect.

| Dataset | No Init | Full Init | Att | Att+Norm+Enc | Att+Norm | Att+Enc | Att+MLP | Att+MLP+Enc | Att+MLP+Norm | MLP+Norm+Encoder | MLP+Norm |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CIFAR-10 | 0.502 | **0.722** | 0.583 | 0.714 | 0.562 | 0.596 | 0.415 | 0.604 | 0.563 | 0.498 | 0.483 |
| Pathfinder | 0.674 | **0.8506** | 0.687 | 0.78 | 0.673 | 0.676 | 0.651 | 0.705 | 0.664 | 0.670 | 0.671 |

Table 4.5: For one layer models, SPT improved test accuracy is due to better initialization of queries and keys parameters.

| Dataset | From Scratch | SPT → Full Model init | SPT → Only QK init |
|---|---|---|---|
| CIFAR-10 | 0.482 | 0.601 +0.12 | 0.607 +0.12 |
| PATHFINDER | 0.671 | 0.703 +0.03 | 0.706 +0.03 |

**parameters has a pronounced effect** (see *Attention+Norm+Encoder*). In particular, using self-pretrained values solely for the Attention weights $W_K, W_Q, W_V$, and $W_O$ can already lead to substantial accuracy gains in CIFAR10. While performance close to full self-pretraining is achieved only when Normalization and embedding parameters (both close in proximity to Attention) are also initialized from self-pretrained weights, these results demonstrate that the Attention block is the primary carrier of the benefits of self-pretraining, with other components playing a secondary role.

To interpret the remaining gap, we note that the results in Table 4.4 are obtained using relatively deep architectures (3 layers for CIFAR and 4 layers for PathFinder). While initializing from self-pretraining only the Attention parameters does not fully recover the performance of full self-pretraining, this gap is expected in such multi-layer settings. If the benefit of self-pretraining lies in inducing effective Attention patterns across layers, then it is logical to expect that partial reversion to random initialization can disrupt this effect. In particular, randomly re-initialized normalization parameters alter the scale and distribution of activations feeding into the Attention block, on which the queries and keys directly depend. As a consequence, even pretrained $W_Q$ and $W_K$ may fail to realize their intended Attention structure, limiting the performance gains achievable through Attention-only self-pretraining.

Motivated by this, we present one last experiment using **1-layer models**: we compare

training from scratch with SPT followed by fine tuning. As opposed to before, we now consider **initializing to SPT weights only the parameters inside the softmax layers**: the queries/keys weights. Table 4.5 confirms our insights: for 1 layer models, where parameter initialization does not have downstream implications in later layers, the effect of SPT can be attributed to a better initialization of $W_Q$ and $W_K$. **Takeaway 4:** Downstream benefits can be observed even when self-pretraining is applied to only a subset of parameters. In line with Takeaway 3, we find that retaining self-pretrained initialization in the Attention layers is particularly impactful, in contrast to MLP layers. Once depth-related confounding effects are removed, special initialization $W_Q$ and $W_K$ alone is sufficient to explain for the performance gains attributed to self-pretraining.

**How much do weights travel?**

Across both CIFAR and PathFinder, we observe that parameters move substantially less when training from scratch than when finetuning (FT) from self-pretrained (SPT) weights (Tables 4.6, 4.7, 4.8, and 4.9). Both displacement in weight space and changes in parameter norms are small for the random-to-finetuned transition, indicating that supervised learning alone induces limited deviation from initialization. In contrast, trajectories involving self-pretraining, particularly SPT→FT and Rand→FT, account for the dominant share of weight movement, showing that finetuning actively shapes pretrained representations rather than merely adjusting them. This pattern is consistent across layers and components and is especially pronounced on Pathfinder. Moreover, MLP layers exhibit substantially larger displacement than attention projections, suggesting higher plasticity and the ability to adapt under supervised training, whereas attention layers change little unless initialized from SPT, indicating a stronger reliance on self-pretraining to acquire meaningful relational structure.

**Analysis for CIFAR**

| Component | R→SPT | R→SC | SPT→FT | SC→FT | R→FT | $\|Q_R\|$ | $\|Q_{SPT}\|$ | $\|Q_{SC}\|$ | $\|Q_{FT}\|$ |
|---|---|---|---|---|---|---|---|---|---|
| Encoder input | 7.94 | 0.66 | 12.22 | 18.64 | 18.75 | 4.39 | 5.84 | 4.14 | 17.34 |

Table 4.6: Encoder input parameter displacement and norms across training stages. The difference between the weight norms obtained from training from scratch and random initialization is small ($\Delta_{\text{norm}} = \|Q_{SC}\| - \|Q_R\| = -0.25$), indicating that training from scratch does not significantly alter the scale of the encoder input parameters relative to random initialization.

| Layer | Component | R→SPT | R→SC | SPT→FT | SC→FT | R→FT | $\|Q_R\|$ | $\|Q_{SPT}\|$ | $\|Q_{SC}\|$ | $\|Q_{FT}\|$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Attn Q | 24.51 | 7.90 | 49.55 | 59.50 | 59.43 | 4.66 | 25.48 | 9.51 | 60.22 |
| 0 | Attn K | 25.87 | 8.86 | 77.88 | 90.54 | 90.73 | 4.66 | 26.98 | 10.31 | 91.58 |
| 0 | Attn V | 11.61 | 5.35 | 14.16 | 10.98 | 9.44 | 4.66 | 12.07 | 7.32 | 8.20 |
| 0 | Attn O | 12.16 | 5.08 | 20.15 | 19.81 | 19.26 | 4.66 | 12.01 | 6.35 | 18.93 |
| 0 | Norm | 5.76 | 2.71 | 3.84 | 5.87 | 6.75 | 8.00 | 7.20 | 6.95 | 4.80 |
| 1 | MLP Up | 28.41 | 11.01 | 76.04 | 84.73 | 84.34 | 6.60 | 29.30 | 13.44 | 84.98 |
| 1 | MLP Down | 26.96 | 9.98 | 96.68 | 107.65 | 107.17 | 4.62 | 27.38 | 11.36 | 107.41 |
| 1 | MLP All | 39.17 | 14.86 | 123.00 | 136.99 | 136.38 | 8.05 | 40.10 | 17.60 | 136.97 |
| 1 | Norm | 4.05 | 1.59 | 6.86 | 8.19 | 8.63 | 8.00 | 6.64 | 7.61 | 11.24 |
| 2 | Attn Q | 20.61 | 7.79 | 49.30 | 60.86 | 60.76 | 4.67 | 21.68 | 9.46 | 61.84 |
| 2 | Attn K | 22.47 | 8.07 | 59.87 | 74.09 | 74.11 | 4.67 | 23.65 | 9.85 | 75.14 |
| 2 | Attn V | 17.56 | 5.97 | 50.13 | 59.42 | 59.18 | 4.67 | 18.06 | 7.86 | 59.67 |
| 2 | Attn O | 15.83 | 5.58 | 58.62 | 65.75 | 65.54 | 4.67 | 16.35 | 7.51 | 66.03 |
| 2 | Norm | 3.56 | 1.77 | 3.35 | 4.57 | 5.34 | 8.00 | 6.34 | 7.17 | 4.56 |
| 3 | MLP Up | 24.95 | 11.82 | 102.90 | 109.50 | 109.44 | 6.63 | 25.25 | 14.24 | 109.73 |
| 3 | MLP Down | 27.74 | 11.46 | 111.26 | 118.17 | 117.38 | 4.65 | 27.87 | 12.85 | 117.45 |
| 3 | MLP All | 37.31 | 16.46 | 151.55 | 161.10 | 160.49 | 8.10 | 37.60 | 19.18 | 160.73 |
| 3 | Norm | 3.62 | 1.30 | 6.96 | 6.26 | 6.18 | 8.00 | 6.19 | 8.33 | 11.70 |
| 4 | Attn Q | 18.96 | 9.08 | 48.32 | 53.14 | 52.66 | 4.73 | 19.88 | 10.67 | 53.24 |
| 4 | Attn K | 19.16 | 8.84 | 57.86 | 64.84 | 64.36 | 4.73 | 20.23 | 10.53 | 64.99 |
| 4 | Attn V | 18.59 | 6.30 | 72.57 | 79.92 | 79.89 | 4.73 | 19.36 | 8.30 | 80.34 |
| 4 | Attn O | 18.60 | 5.60 | 82.62 | 89.12 | 88.90 | 4.73 | 19.47 | 7.90 | 89.39 |
| 4 | Norm | 3.07 | 0.97 | 3.66 | 4.53 | 4.65 | 8.00 | 7.50 | 7.85 | 4.50 |
| 5 | MLP Up | 24.12 | 12.93 | 125.84 | 131.78 | 131.52 | 6.58 | 24.53 | 15.03 | 131.73 |
| 5 | MLP Down | 31.98 | 13.98 | 135.84 | 143.46 | 142.55 | 4.64 | 32.28 | 15.15 | 142.70 |
| 5 | MLP All | 40.06 | 19.04 | 185.17 | 194.80 | 193.96 | 8.05 | 40.54 | 21.34 | 194.21 |
| 5 | Norm | 2.88 | 1.63 | 18.84 | 17.62 | 18.48 | 8.00 | 7.12 | 8.96 | 25.47 |

Table 4.7: Delta between norm of random and norm of scratch in order: +4.85, 5.65, +2.66, 1.69, -1.05, +6.84 +6.74, 9.55, -0.39, 4.79, 5.18, 3.18, 2.84, -0.83, 7.6, 8.2, 11.08, 0.33, 5.94, 5.80, 3.57, 3.17, -0.15, 8.45, 10.51, 13.29, 0.96

**Analysis for Pathfinder**

| Component | R→SPT | R→SC | SPT→FT | SC→FT | R→FT | $\|Q_{\mathrm{R}}\|$ | $\|Q_{\mathrm{SPT}}\|$ | $\|Q_{\mathrm{SC}}\|$ | $\|Q_{\mathrm{FT}}\|$ |
|---|---|---|---|---|---|---|---|---|---|
| Encoder input | 17.96 | 3.57 | 1.85 | 19.47 | 18.92 | 6.70 | 16.90 | 6.54 | 18.05 |

Table 4.8: Encoder input parameter displacement and norms across training stages. The difference between the weight norms obtained from training from scratch and random initialization is small ($\Delta_{\mathrm{norm}} = \|Q_{SC}\| - \|Q_R\| = -0.16$), indicating that training from scratch does not significantly alter the scale of the encoder input parameters relative to random initialization.

| Layer | Component | R→SPT | R→SC | SPT→FT | SC→FT | R→FT | $\|Q_R\|$ | $\|Q_{SPT}\|$ | $\|Q_{SC}\|$ | $\|Q_{FT}\|$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Attn Q | 127.64 | 48.66 | 54.24 | 138.04 | 129.23 | 6.57 | 127.76 | 49.11 | 129.32 |
| 0 | Attn K | 172.04 | 50.52 | 54.30 | 185.61 | 178.34 | 6.57 | 172.33 | 50.99 | 178.60 |
| 0 | Attn V | 15.42 | 52.44 | 11.64 | 55.59 | 18.32 | 6.57 | 13.99 | 53.05 | 17.15 |
| 0 | Attn O | 98.96 | 62.65 | 32.75 | 116.54 | 97.86 | 6.57 | 98.83 | 63.49 | 97.73 |
| 0 | Norm | 10.69 | 6.69 | 0.94 | 11.23 | 10.59 | 11.31 | 4.70 | 11.86 | 4.34 |
| 1 | MLP Up | 108.61 | 53.88 | 46.07 | 128.11 | 116.72 | 6.57 | 108.93 | 54.81 | 117.05 |
| 1 | MLP Down | 173.71 | 47.34 | 54.66 | 189.87 | 184.05 | 6.59 | 174.00 | 48.07 | 184.33 |
| 1 | MLP All | 204.87 | 71.73 | 71.48 | 229.05 | 217.95 | 9.30 | 205.28 | 72.90 | 218.36 |
| 1 | Norm | 10.30 | 5.39 | 3.68 | 10.90 | 9.93 | 11.31 | 9.45 | 11.96 | 11.07 |
| 2 | Attn Q | 137.44 | 50.44 | 51.81 | 146.12 | 137.71 | 6.57 | 137.67 | 50.85 | 137.94 |
| 2 | Attn K | 159.23 | 48.64 | 49.88 | 167.84 | 160.89 | 6.57 | 159.67 | 49.10 | 161.34 |
| 2 | Attn V | 103.28 | 31.03 | 53.13 | 123.11 | 119.14 | 6.57 | 103.49 | 31.64 | 119.41 |
| 2 | Attn O | 108.03 | 33.27 | 57.09 | 126.12 | 122.11 | 6.57 | 108.13 | 33.79 | 122.23 |
| 2 | Norm | 7.82 | 7.31 | 2.68 | 7.65 | 8.52 | 11.31 | 8.35 | 8.03 | 6.86 |
| 3 | MLP Up | 121.88 | 48.94 | 60.51 | 141.23 | 133.06 | 6.54 | 121.89 | 49.51 | 133.12 |
| 3 | MLP Down | 189.59 | 49.10 | 70.81 | 205.75 | 199.57 | 6.60 | 189.76 | 49.59 | 199.72 |
| 3 | MLP All | 225.39 | 69.33 | 93.14 | 249.56 | 239.86 | 9.29 | 225.54 | 70.07 | 240.02 |
| 3 | Norm | 7.30 | 6.25 | 2.13 | 7.22 | 6.64 | 11.31 | 9.03 | 9.73 | 8.99 |
| 4 | Attn Q | 89.88 | 48.52 | 50.80 | 108.37 | 97.04 | 6.56 | 90.05 | 49.04 | 97.17 |
| 4 | Attn K | 108.07 | 49.87 | 52.84 | 125.93 | 116.10 | 6.56 | 108.47 | 50.24 | 116.48 |
| 4 | Attn V | 196.25 | 30.69 | 72.89 | 211.59 | 209.45 | 6.56 | 196.62 | 31.30 | 209.80 |
| 4 | Attn O | 166.50 | 35.85 | 64.68 | 183.96 | 179.94 | 6.56 | 167.14 | 36.26 | 180.54 |
| 4 | Norm | 8.27 | 7.85 | 1.40 | 6.22 | 8.73 | 11.31 | 6.10 | 7.32 | 5.42 |
| 5 | MLP Up | 145.63 | 52.05 | 69.95 | 168.30 | 160.38 | 6.54 | 145.63 | 52.47 | 160.44 |
| 5 | MLP Down | 245.78 | 52.32 | 81.35 | 259.95 | 255.42 | 6.57 | 245.92 | 52.77 | 255.57 |
| 5 | MLP All | 285.68 | 73.80 | 107.29 | 309.68 | 301.60 | 9.27 | 285.81 | 74.42 | 301.76 |
| 5 | Norm | 7.81 | 6.27 | 2.73 | 9.48 | 7.97 | 11.31 | 10.12 | 9.71 | 11.60 |
| 6 | Attn Q | 104.73 | 48.12 | 61.59 | 121.54 | 111.79 | 6.58 | 104.81 | 48.53 | 111.83 |
| 6 | Attn K | 127.40 | 48.75 | 63.64 | 145.94 | 137.34 | 6.58 | 127.44 | 49.16 | 137.38 |
| 6 | Attn V | 247.10 | 36.29 | 77.28 | 258.34 | 255.48 | 6.58 | 247.23 | 36.83 | 255.60 |
| 6 | Attn O | 217.16 | 44.17 | 73.04 | 229.59 | 225.50 | 6.58 | 217.34 | 44.79 | 225.69 |
| 6 | Norm | 9.59 | 7.45 | 1.44 | 7.04 | 8.84 | 11.31 | 6.47 | 7.10 | 6.73 |
| 7 | MLP Up | 128.53 | 54.64 | 73.37 | 156.92 | 147.30 | 6.57 | 128.75 | 55.06 | 147.45 |
| 7 | MLP Down | 292.75 | 61.59 | 120.01 | 315.85 | 310.40 | 6.56 | 292.94 | 61.97 | 310.53 |
| 7 | MLP All | 319.73 | 82.33 | 140.66 | 352.68 | 343.57 | 9.29 | 319.98 | 82.90 | 343.76 |
| 7 | Norm | 6.96 | 5.02 | 7.41 | 9.34 | 7.72 | 11.31 | 8.04 | 10.77 | 14.29 |

Table 4.9: Layer-wise parameter displacement and parameter norms across training stages. Delta between random init and from scratch: 42.54, 44.42, 46.48, 56.92, 0.55, 48.24, 41.48, 63.60, 0.65, 44.28, 42.53, 25.06, 27.22, -3.28, 42.97, 42.99, 60,78, -1.58, 42.48, 43.68, 24.74, 29.70, -3.99, 45.93, 46.20, 65.15, -1.60, 41.95, 42.58, 30.25, 38.21, -4.21, 48.49, 55.41, 73.61, -0.54.

### A minimal synthetic task for understanding self-pretraining

In Section 4.4, we carefully ablated the Self-Pretraining (SPT) paradigm on the long range-Arena, and showed that the benefits of SPT can be related to successful fitting of the training data (Takeaway 2), can be observed even in 1-layer setups, and that self-pretraining-informed initialization is particularly effective on Attention weights (Takeaway 4, Table 4.4 and Table 4.5), which if randomly initialized struggle to establish relations between tokens (Takeaway 3, Table 4.3).

In this section, we move away from the long range-Arena, and design and ablate a minimal synthetic task that is able to showcase the benefits of self-pretraining. Besides broadening the scope of our findings, this task allows more controlled ablations, smaller models, and a full identification of the self-pretraining action on the Attention weights. Specifically, we identify that self-pretraining is effectively modifying the Attention matrix, removing standard positional encodings towards a simpler yet flexible proximity bias.

#### Setup

**Task and Data.** Our synthetic task is a binary classification problem on sequences. Sequences are of length 100, and have different statistics depending on the associated label. Two anchor points (different for each label) are used to define a region in the landscape where trajectories are concentrated, depending on the label. Latently, each datapoint is a ellipsoidal-shaped trajectory moving back-and-forth between the anchors. To make the problem difficult, and showcase non-trivial learning, each trajectory moves at a different speed, obtained by time reparametrization using a random phase. On top of this, each trajectory has a different orbital amplitude, has slightly "tilted" orientation, and is corrupted by both Gaussian noise and outliers. A description of this task and an example plot are provided in Appendix 4.2.

The dataset contains 100 training sequences (balanced labels, with 15% random label flips), 200 validation sequences (balanced), 400 test sequences (balanced), and 12000 unlabeled sequences used for self-supervised pretraining.

**Model.** A one layer Attention module is tasked to predict the label. Such module is drastically simpler than the one adopted by [56] and used in our Section 4.4: the 2-dimensional trajectories are encoded linearly to a hidden dimension of 32, and then are processed by standard softmax Attention (see Section 4.2) with weights $W_Q, W_K, W_V, W_O$ (no biases are used). A single head is used, no MLP follows the Attention block, and do not make use of LayerNorm. Crucially, we make use of absolute **positional embeddings**, added to the encoded input before this is processed by the Attention block exactly as in Section 4.4.

The power of this setup is that, if successful, it allows us to understand self-pretraining by only looking at the evolution of five weights.

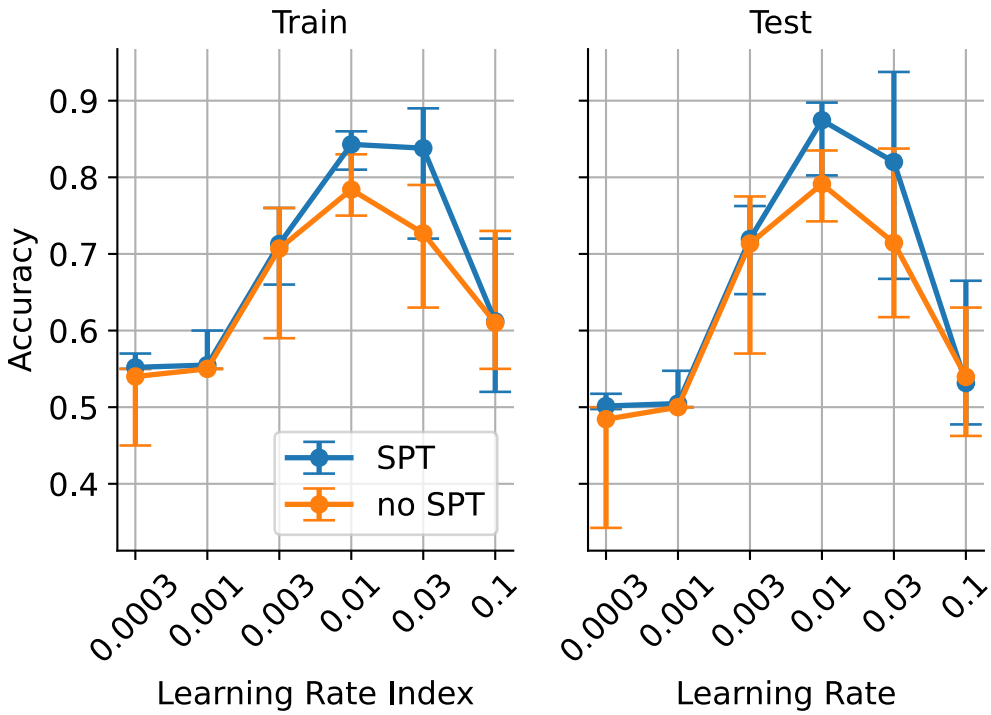


Figure 4.5: Performance on the toy task (1-layer model) described in Section 4.4. Train and test accuracies are averaged over 10 random seeds (shown is max-min interval) for different learning rates. SPT outperforms the no-SPT baseline, achieving higher peak accuracy at intermediate learning rates.

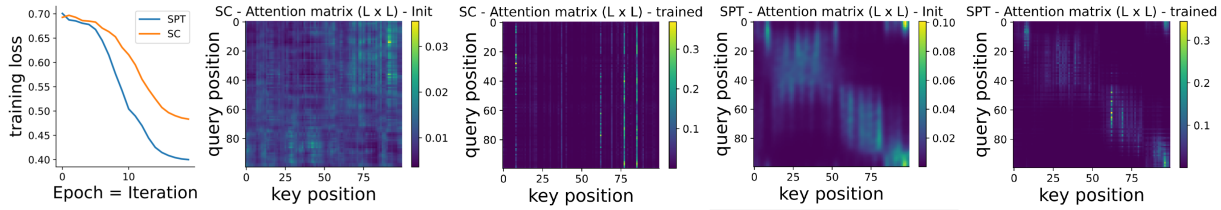


Figure 4.6: Toy Attention evolution. Left: training loss over iterations for self-pretraining (SPT) and From-Scratch training (SC). Right: Attention matrices ($L \times L$) at initialization and after training. SPT rapidly develops structure during pretraining exhibiting a proximity bias. Compared to random initialization, such structure is able to develop into a richer sequence mixer after finetuning (noted as "trained").

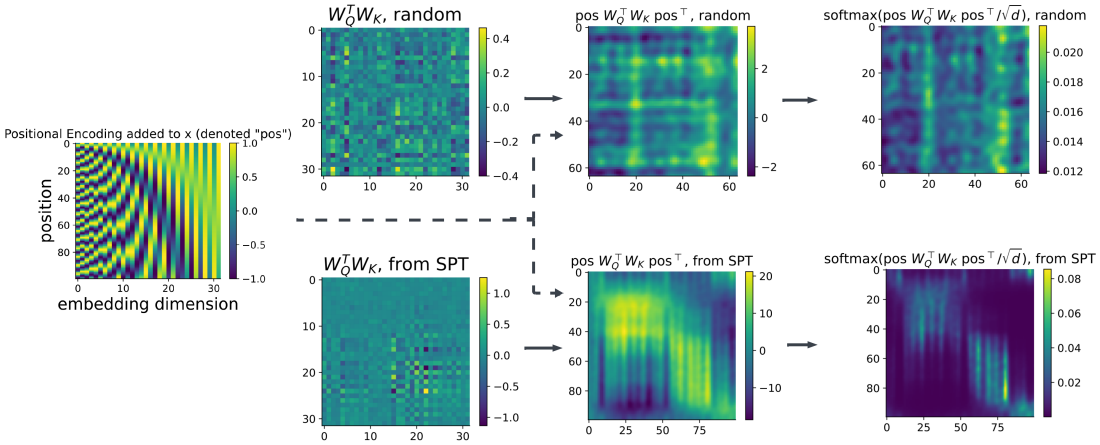


Figure 4.7: Toy position undoing. Visualization of Attention components with random initialization (top) and after SPT training (bottom). While random weights fail to recover positional structure, SPT learns weights that effectively undo positional encoding, producing coherent, position-aligned Attention after softmax. Here we set the input content $X = 0$ and feed only positional embeddings pos through Q/K to isolate the effect of positional information.

### Toy Dataset

In Figure 4.5, we consider 20 epochs of training from labels (supervised learning), preceded by a potential self-pretraining step (10 epochs) using a masked objective (as in Section 4.4). For each learning rate, we gather performance over 10 seeds, and show the average along with a max-min confidence interval. **We observe consistent benefits from self-pretraining in this controlled setting**, with a boost of $\sim 10\%$ in both training and test accuracies at the optimal learning rates. We also provide plots for the training loss and a different epoch budget in Appendix 4.2.

**Attention Matrix.** We proceed to inspect how the Attention matrix evolves from random initialization during pretraining. To do this, we inspect runs associated with the learning rate maximizing the gap in training loss, and then check if similar findings can also extend to the long range-Arena in Section 4.4. In Figure 4.6, we see a clear pattern emerging: when feeding a single input into the self-pretrained Attention mechanism, a **diagonally dominated structure emerges** resembling ALiBi [225] and thus drawing a direct link to the findings of [214] on the long range-Arena. Compared to random initialization of $W_Q, W_K$, which produces a uniform Attention pattern (see theoretical results in [226]), this pattern establishes a proximity bias in the sequence, which is then picked up during finetuning to produce better sequence mixer.

**Crucial role of Positional Encodings.** Fascinated by the structure pretraining imposes to the Attention matrix, we next inspect how this pattern emerges and its relation to additive positional embeddings used in the architecture. Figure 4.7 is perhaps the crucial plot of the section: we feed to the Attention matrix directly the positional encodings i.e., we marginalize out the input and observe that the same pattern observed in Figure 4.6 emerges. This is a clear, data-independent signal: **self-pretrained Attention weights are in this setting effectively "undoing" sinusoidal positional encodings**, building up a structure which instead allows for direct comparisons of nearby tokens already at the first steps of finetuning accelerating convergence.

Intriguingly, Figure 4.7 also shows how weights act on positional encodings: a special structure can be observed in $W_Q^\top W_K$. As shown in Figure 4.8, the distributions of $W_Q, W_K$ do not exhibit any particular shift from random initialization yet their product does, and is able to act on positional embeddings in an effective way.

### Back to LRA

In the last subsections, we have established that self-pretraining effectively builds a diagonally-dominated Attention structure (Figure 4.6) through a direct action on the positional encodings controlled by the product $W_Q^\top W_K$ (Figure 4.7). We now are curious to visualize if a similar pattern can be observed in the long range-Arena. To do this, we turn our Attention back to the simplest setup in Section 4.4, i.e. the one of Table 4.5.

**Attention Structure.** Figure 4.8 shows the query-key product and the resulting softmax Attention matrix obtained after feeding a CIFAR10 or a PathFinder example to the self-pretrained network. What we notice is a pattern which is strikingly similar to Figure 4.6: pretrained $W_Q, W_K$ establish a diagonal-dominated Attention mechanism.

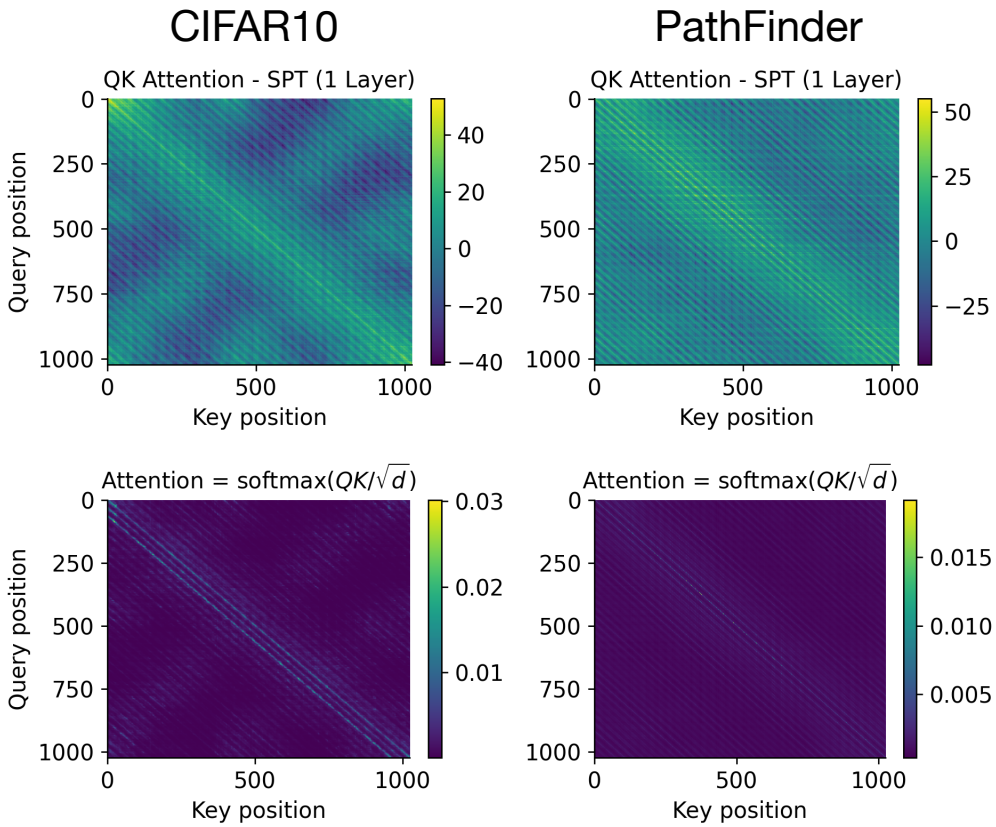


Figure 4.8: Visualization of raw $QK^\top$ similarity matrices (top) and softmax normalized attention weights (bottom) for a single-layer self-pretrained transformer on CIFAR-10 (left) and PathFinder (right). The raw $QK^\top$ matrices show clear diagonal structure, with noisier patterns for CIFAR-10 and smoother, more coherent structure for PathFinder. Softmax normalization yields sparse, predominantly diagonal attention, emphasizing local interactions. Weights are taken after SPT (before finetuning) for the 1-layer setting of Table 4.5.

**One last ablation.** At this point, a natural question emerges: wouldn't it be better to directly replace absolute positional encodings, i.e. (using the notation introduced in Section 4.3) the computation

$$Z^{(0)} = X + \text{pos}, \qquad \text{pos} \in R^{L \times D}$$

with a direct modification of the Attention structure at initialization, using for example the ALiBi [225] approach:

$$\text{softmax}\left(\frac{QK^\top}{\sqrt{d}} + A\right), \quad A_{ij} = -m|i-j|$$

directly imposing an off-diagonal decay structure? on the Attention matrix? In Table 4.10 we see that this idea, to some extent, can work: it improves from-scratch trainability on CIFAR10; yet it brings about random-chance performance in every run we performed on PathFinder (even after self-pretraining). This is not surprising for two reasons (1) CIFAR10 classification could be performed by aggregating local features, yet PathFinder clearly necessitates long range interactions; (2) [214] also report that ALiBi does not work well on PathFinder data, yet establish that this strategy can be successful for other tasks in the LRA as we also found the case to be in CIFAR10.

This result matches the intuition we have built up in this section: self-pretraining biases the initialization of the Attention matrix through smart initialization of the $W_Q$ and $W_K$ parameters yet does not constrain further modifications that may be necessary during finetuning. In contrast, ALiBi exhibits a strong proximity bias at initialization, but its structure is not flexible enough to support long range information exchange during finetuning on challenging tasks like pathfinder. Interestingly, we find in Table 4.10 that the best option is actually to combine the two approaches.

Table 4.10: Same setup as Table 4.1, ablating the effect of ALiBi positional encodings on self-pretraining accuracy. We notice that the ALiBi approach is able to improve from-scratch performance on CIFAR by $\sim 6\%$, yet it is not a good fit for Pathfinder.

| **Setting** | **Variant** | **From Scratch** | **SPT** |
|---|---|---|---|
| 3 Layers CIFAR-10 | PE | 0.501 | 0.722 |
| | PE + ALiBi | 0.588 | 0.754 |
| | ALiBi | 0.564 | 0.575 |
| 4 Layers Pathfinder | PE | 0.671 | 0.8506 |
| | PE + ALiBi | 0.678 | 0.9007 |
| | ALiBi | 0.500 | 0.500 |

## 4.5 Conclusion

In this chapter, Self-PreTraining has been investigated as a strategy to improve sequence modeling performance, with a particular focus on understanding its impact on learning dynamics, representation quality, and generalization behavior. Through a series of controlled experiments, this chapter examined how self-pretrained initialization differs from standard supervised training when applied to deep sequence models.

The analysis demonstrated that Self-PreTraining can provide a more structured and informative initialization by exposing the model to a reconstruction-based objective before fine tuning. This pretraining phase encourages the model to capture intrinsic temporal

dependencies present in the data, thereby facilitating more stable optimization and improved convergence during downstream supervised learning. The results show that these benefits are especially evident in scenarios characterized by limited labeled data or shallow model architectures, where training from scratch often leads to suboptimal solutions.

Beyond performance improvements, this chapter offered insights into why Self-PreTraining is effective. By analyzing different training pipelines, pretraining durations, and layer-wise freezing strategies, the study highlighted how self-pretrained models develop representations that are better aligned with the temporal structure of the input signals. These findings suggest that Self-PreTraining acts as a form of inductive bias, guiding the model toward more meaningful regions of the parameter space before task-specific learning begins.

This chapter establishes Self-PreTraining as a principled and flexible approach for enhancing time series classification models, providing both empirical evidence and conceptual understanding of its advantages. The insights gained here form the methodological foundation for the subsequent chapter, which explores the applicability of Self-PreTraining in transformer-based architectures and heterogeneous medical time series datasets, bridging the gap between controlled experimental analysis and real world clinical applications.

# Chapter 5

# Is Self-PreTraining useful for Human Locomotion Identification and Medical Time Series Diagnosis?

**Manuscript in preparation**

The contents of this chapter are confidential, as they are intended for submission to a peer-reviewed venue as First author.

## 5.1 Introduction

**Rationale for the Selected Application Domains**

The three datasets considered in this chapter CAMARGO 2021, the Non-EEG Dataset for Assessment of Neurological Status, and Gait in Parkinson's Disease are not selected as independent clinical applications, but as components of a graded validation framework for the Self-PreTraining methodology developed in Chapter 4. The objective of this chapter is therefore not to address three separate medical problems in isolation, but to systematically assess whether the benefits of SPT observed in Chapter 4 persist when the methodology is applied to real medical time series characterized by small, heterogeneous, and subject-specific data the same regime that motivates the use of SPT in lower limb rehabilitation robotics.

The CAMARGO dataset provides the in-domain test case, directly aligned with the lower limb rehabilitation robotics setting that constitutes the central application of this thesis. Locomotion recognition on multimodal IMU signals shares the same temporal structure, sensing modality, and inter-subject variability already analyzed in Chapter 3, and therefore

allows the impact of SPT to be evaluated within the precise context for which it is intended.

The Gait in Parkinson's Disease dataset extends the analysis to a clinically relevant diagnostic task built on the same biomechanical signal modality. Foot force signals exhibit temporal dynamics and stride-level structure that are conceptually adjacent to the gait-based classification tasks of Chapter 3, but the downstream objective is reframed from locomotion classification for control to disease detection. This setting therefore tests whether SPT remains effective when the underlying signal domain is preserved but the clinical question changes, providing a first axis of generalization for the methodology.

The Non-EEG stress dataset constitutes the most distant application, deliberately chosen to probe a regime that departs substantially from gait analysis. The signals involved electrodermal activity, skin temperature, heart rate, oxygen saturation, and wrist accelerometry differ from IMU and force data both in modality and in temporal structure, and the underlying physiological phenomena are governed by autonomic rather than biomechanical dynamics. Evaluating SPT in this setting therefore assesses whether the methodology generalizes beyond gait and beyond the rehabilitation robotics context, toward broader medical time series tasks that share the small-data, high-variability characteristics targeted by SPT.

Taken together, these three domains form a graded validation framework that moves outward from the core rehabilitation robotics setting toward increasingly distant medical time series. This design allows the experimental analysis presented in this chapter to assess not only the raw effectiveness of SPT, but also the extent to which its benefits generalize within and beyond the rehabilitation robotics regime that motivates this thesis.

### Time Series in the World

Time series data is pervasive in domains such as healthcare, industrial monitoring, finance, and autonomous driving. Despite their ubiquity, developing high performance models for time-series analysis remains challenging, particularly in domains where labeled data is scarce or expensive to obtain. This limitation has made *pre-training* a central paradigm for learning robust and transferable representations.

In this context, it is important to distinguish between **Self-Supervised Learning (SSL)** and **Self-PreTraining (SPT)**, two closely related yet conceptually distinct strategies for performing pre-training on time-series data. SSL typically refers to large scale pre-training performed on an external dataset that differs from the one used during downstream fine-tuning. In contrast, SPT denotes a setting in which the model is pre-trained directly on the same dataset that will later be used for fine tuning [56, 57].

In SSL, the model is trained using automatically generated supervisory signals (e.g., forecasting, masking, or contrastive objectives) on unlabeled data, often collected from diverse

sources or domains. The goal is to learn broadly transferable representations that can generalize across tasks and datasets. After this pre-training stage, the model is fine-tuned on labeled data specific to the downstream task.

In contrast, SPT exploits the unlabeled portion or alternative views of the target dataset itself. Rather than leveraging out-of-domain data, SPT focuses on in-domain pre-training to improve optimization, representation quality, and convergence specifically for the target task.

Conceptually, SSL can be viewed as *out of domain representation learning*, whereas SPT corresponds to *in domain pre-training*. While both approaches rely on self-supervised objectives, they differ fundamentally in data usage and generalization scope.

This distinction is particularly important in real world time series applications. In industrial or multimodal settings, large external corpora may be available, making SSL attractive. However, in highly specialized domains such as healthcare, external datasets may be limited, heterogeneous, or inaccessible due to privacy constraints. In such cases, SPT provides a practical and effective alternative by maximizing the utility of the available in domain data.

Motivated by these challenges, there has been a surge of research in self-supervised pre-training and transfer learning for time series tasks, aiming to leverage abundant unlabeled data to learn useful representations before fine-tuning on specific downstream objectives. In the following, we review recent developments in self-supervised and transfer learning frameworks for time series modeling, with particular emphasis on medical, industrial, and multimodal applications.

### Self-Supervised Pre-training for Time Series

Early work such as Shi et al. introduced a self-supervised pre-training pipeline tailored for time-series classification, using denoising and Dynamic Time Warping (DTW)–based similarity discrimination to learn transferable representations from large unlabeled corpora like UCR2015 [77]. Following this direction, Dong et al. proposed TimeSiam, a Siamese-network framework that reconstructs masked subsequences by leveraging past temporal contexts, achieving state-of-the-art results across forecasting and classification tasks [78]. Similarly, Zhang et al. explored time–frequency consistency as a pretext, aligning representations in time and frequency domains through contrastive learning and showing impressive cross-domain transfer [79]. These methods highlight how diverse self-supervised objectives—ranging from masked reconstruction to contrastive alignment—can extract latent temporal structures without requiring labels.

The field has also benefited from comprehensive surveys that synthesize these advances. Zhang et al. provided a taxonomy of self-supervised learning (SSL) for time series, cate-

gorizing approaches into generative, contrastive, and adversarial paradigms and identifying key challenges such as seasonal variability and multivariate augmentation design [80]. Liu et al. offered a domain-specific review focused on medical time-series, discussing data augmentations, encoder choices, and loss functions used in 43 studies for applications like ICU mortality prediction and sleep scoring [81]. Most recently, Ma et al. reviewed large pre-trained models for time-series mining, noting the growing adoption of transformer backbones and masked modeling strategies [82]. These surveys underscore both the promise of pre-training and the open research questions in designing generalizable representations.

### Pre-training in Clinical and Multimodal Contexts

In the medical domain, labeled data scarcity is particularly acute. King et al. introduced a multimodal pretraining framework that aligns ICU time-series measurements with clinical notes, using contrastive objectives and masked token prediction to build encoders that transfer effectively to mortality prediction and phenotyping [83]. Similarly, Raghu et al. explored contrastive pre-training for multimodal clinical time series, integrating high-frequency ECG signals with labs and vitals to enhance downstream detection of elevated mPAP and mortality [84]. Xu et al. proposed TransEHR, a self-supervised transformer architecture that models asynchronous events and measurements through masked value prediction, replaced-token detection, and event forecasting [85]. These works demonstrate that multimodal alignment and event-aware architectures are critical when transferring self-supervised representations in healthcare.

Several frameworks also address irregular or asynchronous sampling patterns, which are common in EHRs and IoT settings. Chowdhury et al. introduced PrimeNet, combining time-sensitive contrastive learning with duration-based masking to better encode irregularities in multivariate time series [86]. Malhotra et al.'s earlier TimeNet also tackled irregular data, leveraging self-supervised reconstruction and contrastive losses to improve few-shot transfer in clinical and sensor datasets [87]. These methods highlight the importance of domain-specific pretext designs that account for non-uniform temporal structures.

### Scaling Pre-training Across Domains

Beyond clinical settings, researchers have explored how large pre-trained models can generalize across disparate domains. Prabhakar et al. proposed LPTM, introducing adaptive segmentation during masked modeling to handle varying sampling rates and temporal dynamics in multi-domain datasets (epidemiology, energy, traffic, and behavioral sensors) [88]. Dong et al.'s SimMTM further demonstrated that series-wise contrastive alignment com-

bined with masked reconstruction improves performance on forecasting, classification, and imputation across domains [89]. Similarly, Zhang et al. introduced UniMTS, which unifies motion time-series with semantic text descriptions to achieve robust zero-shot and few-shot performance [90]. These advances suggest that large unlabeled datasets, combined with carefully crafted pretext tasks, can yield universal backbones for time-series analysis.

### Insights from Other Modalities

Parallel research in computer vision and speech has inspired many pre-training strategies adopted for time series. For instance, masked autoencoders have been widely explored for medical imaging tasks, where self-pretraining directly on the target dataset mitigates domain shift and improves segmentation and classification [91, 92, 93, 94, 95]. Techniques such as emotion2vec in speech emotion recognition [97] and wav2vec-based speech pretraining [96] also illustrate how contrastive and reconstruction losses can be combined to learn rich representations with minimal labels. Recent multimodal frameworks such as SLIP [98] and self-training strategies [99] further show the potential of integrating multiple supervisory signals. Surveys on sequential transfer learning [100] and dataset size requirements [69] emphasize that even small, domain-specific datasets can yield effective pretraining when coupled with appropriate self-supervised objectives.

### Rationale

To the best of our knowledge, no prior work has systematically examined Self-PreTraining (SPT) in the context of medical time series. With this study, we aim to fill this gap by pursuing the following objectives:

- Compare transformer models trained from scratch with those initialized through SPT to quantify the benefits of this method.
- Evaluate SPT in the simplest setting, where only univariate time series inputs are available.
- Extend the analysis to multivariate time series using four distinct masking strategies designed to probe different aspects of temporal and cross-channel structure.
- Assess whether the scale of the model influences the effectiveness of SPT, particularly for deeper transformer architectures.
- Provide an optimization-based interpretation to explain the observed performance differences between training regimes.

## 5.2 Materials

Table 5.1 summarizes the principal characteristics of the datasets considered in this work, highlighting their domain of application, classification task, cohort size, sensing modalities, segmentation strategy, evaluation protocol, and class distribution. The selected datasets cover distinct yet complementary domains, allowing for a comprehensive assessment across multimodal biomechanical, physiological, and neurological time-series data.

Comprehensive methodological details, including acquisition protocols and preprocessing procedures, are described in the preceding subsections dedicated to each dataset.

Table 5.1: Summary of datasets used in this study.

| Dataset | Domain | Task | #Subj. | Modalities | Window | Eval. | Class Dist. |
|---|---|---|---|---|---|---|---|
| CAMARGO 2021 | Biomechanical / Multimodal Gait | Locomotion Classification | 21 | 4 IMUs + 11 EMG (35 signals) | 100 ms | LOSO | 20±2.5% |
| Non-EEG | Wearable Physiological / Stress | Stress Detection | 20 | EDA, Temp., Acc., HR, $SpO_2$ | 10 s | LOSO | 60% / 40% |
| Gait PD | Neurological / Gait Analysis | PD Detection | 166 | 16 Force + 2 Sum Signals | 5 s | Subject-wise | 69% / 31% |

### CAMARGO 2021

The same dataset described here 3.2

### Non-EEG Dataset for Assessment of Neurological Status

This work also incorporates the Non-EEG Dataset for Assessment of Neurological Status (v1.0.0), a publicly available collection hosted on PhysioNet. The dataset includes recordings from 20 healthy adult participants, acquired by the Quality of Life Laboratory at the University of Texas at Dallas. Each subject was monitored using a set of non-EEG physiological sensors that measured electrodermal activity (EDA), skin temperature, three-axis wrist acceleration, heart rate (HR), and arterial oxygen saturation (SpO). All data is provided in WFDB format, with each participant having one file containing EDA, temperature, and accelerometry, and a second file containing HR and SpO signals; an accompanying annotation file specifies the timing and labels of experimental phases.

Recordings were collected across seven consecutive stages designed to elicit varying physical and emotional responses. Participants began with a 5-minute resting period, followed by a physical-stress block that included standing, slow walking at 1 mph, and faster walking or jogging at 3 mph. After a second resting interval, subjects completed a cognitive-stress segment involving serial subtraction and a Stroop test, followed again by a relaxation period. An emotional-stress segment then took place, consisting of a brief anticipation interval and a

5-minute horror-movie clip, before concluding with a final resting phase. This structure yields continuous multimodal physiological data spanning relaxation, physical exertion, cognitive load, and emotional stimuli. For full methodological details, including sensor specifications and experimental protocol, refer to the original publication by [227]. For the stress dataset, windowing is performed within a leave-one-subject-out (LOSO) evaluation framework, where windows are generated only after selecting the held-out subject to avoid any form of inter-subject leakage. Each recording is segmented into temporal windows of 10 seconds, capturing sufficiently long physiological responses to characterize transitions between relaxed and stressed states. This process produces a substantial number of windowed samples per participant. The resulting class distribution remains inherently imbalanced—typically with a higher proportion of non-stress windows than stress windows—reflecting the natural occurrence of these states within the experimental protocol. No artificial class balancing or resampling is applied during window creation. The resulting class distribution remains naturally imbalanced—approximately 60% control and 40% Stress's in our windowed dataset—reflecting the underlying proportions of the two groups. No artificial class balancing or reweighting is applied during window generation.

**Gait in Parkinson's Disease**

In this study we employ the publicly accessible Gait in Parkinson's Disease v1.0.0 dataset from MIT Laboratory for Computational Physiology / PhysioNet. The dataset comprises multichannel recordings of foot-ground reaction forces collected from 93 patients diagnosed with idiopathic Parkinson's Disease (mean age 66.3 years, 63 % male) and 73 healthy control subjects (mean age 66.3 years, 55 % male). Under each foot of every participant, eight force sensors (Ultraflex Computer Dyno Graphy, Infotronic Inc.) were placed, yielding 16 individual sensor outputs in addition to two combined sum signals (one per foot). These signals were digitized at a sampling rate of 100 samples per second. The participants walked on level ground at their self-selected pace for approximately two minutes. The available data facilitate analysis of stride-to-stride dynamics and variability, such as centre-of-pressure trajectories and timing measures (e.g., stride and swing times). In addition to the force sensor data, demographic information and measures of disease severity (via the Hoehn & Yahr scale and/or the Unified Parkinson's Disease Rating Scale) are included in the dataset. A subset of the recordings also contains dual-task walking (walking while performing serial subtraction) to explore gait under cognitive load [228]. For the Gait in Parkinson's Disease dataset, windowing is performed after the subject-wise train/validation/test split to prevent cross-subject information leakage. Each recording is segmented into temporal windows of 500 samples (corresponding to 5 s at the dataset's 100 Hz sampling rate). This procedure yields

a large collection of windowed gait segments for each participant. The resulting class distribution remains naturally imbalanced—approximately 69% control and 31% Parkinson's in our windowed dataset—reflecting the underlying proportions of the two groups. No artificial class balancing or reweighting is applied during window generation.

## 5.3 Methods

This section outlines the architecture of the Transformer models adopted in our study—used respectively for Self-Supervised Pre-Training (SPT), subsequent fine tuning, and training from scratch—and describes the four-fold masking strategy employed for multivariate time series learning. All three training paradigms rely on the same backbone. During SPT, the model learns to reconstruct masked segments of the input sequence, enabling it to internalize modality-specific dynamics, inter-sensor correlations, and shared temporal patterns without requiring labels. fine tuning then adapts this pre-trained representation to the downstream classification task, while training from scratch uses identical architecture but learns all parameters solely from supervised data.

The four-fold masking strategy plays a central role in shaping the representation learned during SPT. Specifically, the time series is perturbed along multiple dimensions—temporal spans, channel subsets, point-wise samples, and structured contiguous blocks—allowing the model to experience a diverse set of information-removal patterns. This multi-masking design forces the encoder to infer missing information from both temporal context and cross-channel interactions, which is particularly beneficial for multivariate physiological data where dependencies unfold across different timescales and sensor modalities. Together, the shared Transformer backbone and the robust masking framework ensure a consistent and fair comparison across pre-training, fine tuning, and training-from-scratch settings, while maximizing the expressiveness and generalization capacity of the learned representations.

#### Transformer architecture

Is the same described in 4.3.

#### Masking policy for SPT

While all masking strategies remove approximately the same proportion of input elements, they differ fundamentally in their structural organization. The key distinction lies in the pattern according to which information is removed across time and feature dimensions. By

varying this structure, we impose different inductive biases on the model and encourage it to learn complementary aspects of the underlying temporal data.

Unstructured masking (e.g., point-wise) removes isolated scalar values, promoting local interpolation and distributed representations. In contrast, structured masking along the temporal axis (e.g., block-wise removal) introduces contiguous gaps, forcing the model to reason over longer temporal contexts and capture global sequence dynamics. Similarly, structured masking along the feature axis (column-wise masking) removes entire sensor channels, requiring the model to infer missing modalities from cross-channel dependencies.

Thus, although the overall masking ratio remains constant, the geometric arrangement of the masked entries determines whether the model primarily learns short-range continuity, long-range temporal coherence, cross-modal redundancy, or a combination of these factors. Exploring multiple masking patterns allows us to systematically analyze how different structural corruptions influence representation learning during Self-PreTraining.

**Masking Strategies.** Figure 5.1 illustrates the masking configurations. Each strategy produces a mask with identical shape but different structural properties, thereby imposing different inductive biases.

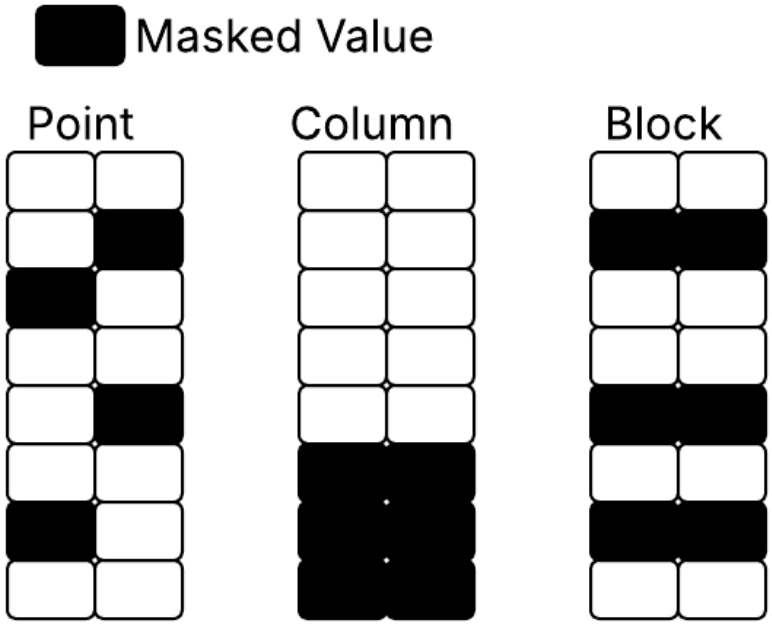


Figure 5.1: Masking strategies used during SPT. From left to right: point-wise masking, timestep masking, feature masking, and their mixed union.

**Point-wise masking.** In this configuration, each scalar entry is independently masked

with probability p This produces a fine-grained, unstructured occlusion pattern. The model must reconstruct missing values using both short range temporal context and cross feature information. This strategy resembles classical denoising autoencoding and encourages distributed, locally coherent representations.

**Column masking.** In this configuration, entire feature channels are removed from the input sequence. For each sample, we randomly select a fraction p of the available features and mask them across all timesteps. In other words, once a feature is selected, its entire temporal trajectory is hidden for that sequence.

This means that the model cannot rely on that sensor at any time point and must reconstruct its values using information from the remaining channels. As a result, feature masking encourages the model to learn cross-channel relationships and capture dependencies between different sensors or modalities. This is particularly relevant in multivariate physiological data, where signals often exhibit redundancy or coordinated behavior across channels.

**Block masking.** Block masking removes contiguous temporal segments across all features. Instead of selecting isolated timesteps, we iteratively sample starting indices and mask blocks of length $L$ along the time axis until approximately $p \cdot t$ timesteps are covered. This concentrates supervision on longer structured gaps and requires the model to infer extended temporal dynamics rather than isolated points.

**Mixed Masking Policy.** The mixed strategy combines multiple masking axes by taking the logical union of independently sampled masks:

$$M^{\text{mix}} = M^{\text{point}}(p_1) \ \vee \ M^{\text{column}}(p_2) \ \vee \ M^{\text{feature}}(p_3).$$

The final mask $M^{\text{mix}}$ contains all positions masked by any individual strategy. The individual rates $p_1, p_2, p_3$ are chosen such that the expected overall masking proportion is approximately equal to the target ratio $p$. This mixed policy simultaneously introduces fine-grained perturbation, structured temporal gaps, and cross-channel occlusion. As a result, the model must reconstruct information across multiple axes of variation, encouraging richer and more generalizable latent representations.

### Experimental Set Up

**Evaluation Protocol.** For the CAMARGO dataset, which is formulated as a multiclass locomotion classification task, we adopted a Leave-One-Subject-Out (LOSO) cross-validation strategy. In each fold, data from $N - 1$ subjects were used for training, and the remaining subject was used exclusively for testing. This procedure was repeated once per subject to obtain subject-independent performance estimates.

The Non-EEG and Parkinson's Disease datasets correspond to binary classification problems (stress vs. control, and Parkinson's vs. healthy control, respectively). For the Non-EEG dataset, we similarly applied LOSO.

For the Parkinson's Disease dataset, LOSO was not feasible due to dataset organization and sample distribution constraints. Instead, we performed repeated subject-wise splits, ensuring that all samples from a given subject were assigned exclusively to either training or testing. This splitting procedure was repeated multiple times with different random partitions to obtain robust performance estimates.

**Performance Metrics.** Performance was evaluated using accuracy, precision, recall, and F1-score. For the multiclass CAMARGO dataset, metrics were computed using macro-averaging. For the binary datasets (Non-EEG and Parkinson's Disease), metrics were computed in their standard binary formulation.

**Training Configuration.** All experiments were conducted on an NVIDIA A100 GPU with 80 GB of memory.

For the training-from-scratch setup, models were trained for 10 epochs. A learning-rate sweep over of 0.01 was.

For the Self-PreTraining (SPT) phase, models were pre-trained for 100 epochs using a learning rate of 0.001, selected empirically for stable convergence. During fine-tuning, models were optimized for 15 epochs with a learning rate of 0.01.

Early stopping was applied by monitoring the validation loss. Training was terminated if no improvement was observed for 100 consecutive epochs. In practice, convergence occurred well before reaching this limit, and the runs completed within the scheduled number of epochs.

**Statistical Analysis.** To assess whether performance differences between SPT and training-from-scratch were statistically significant, we employed the Wilcoxon signed-rank test with Bonferroni correction to account for multiple comparisons [196, 197].

## 5.4 Result and Discussion

**Results**

**Univariate Analysis**

**Experimental Design.** The univariate experiments evaluate the impact of Structured Pre-Training (SPT) when models are trained on single-sensor modalities. For each dataset, we selected the most informative individual sensor channel based on prior domain knowledge and preliminary performance screening. The objective of this analysis is twofold: (i) to isolate how SPT interacts with individual signal characteristics, and (ii) to assess how model depth influences performance in the absence of multimodal fusion.

For each selected sensor, we trained Transformer models with 1, 2, and 3 Sequence Model layers under two initialization regimes: (i) training from scratch using Xavier initialization, and (ii) SPT initialization followed by fine-tuning. Performance metrics are reported in Table 5.2, while Figure 5.2 illustrates the depth-dependent improvement trends.

**Sensor Selection.** The chosen sensors correspond to those exhibiting the strongest standalone discriminative capability within each dataset:

- **CAMARGO dataset:** Gyroscope signals from the foot-mounted IMU (Y-axis and Z-axis). These axes capture rotational components of gait dynamics, with the Y-axis typically encoding stronger lateral movement patterns.
- **Non-EEG Stress dataset:** Accelerometer (AccY) and Heart Rate (HR) signals. Accelerometer data capture agitation-related movement bursts, whereas HR reflects slower physiological stress responses.
- **Parkinson dataset:** LeftFoot and RightFoot force-derived gait signals, which encode stride-to-stride irregularities associated with Parkinsonian motor impairment.

These modalities were selected because they represent the strongest unimodal predictors for their respective tasks, thereby providing a meaningful testbed for assessing SPT in isolation.

**Dataset-Specific Observations. CAMARGO (Gait – Multiclass).** The CAMARGO dataset exhibits periodic gait structure, which naturally aligns with the temporal modeling capacity of Transformers. From-scratch performance remains moderate, reflecting the multiclass complexity of locomotion tasks. SPT consistently improves performance, particularly

for the Y-axis gyroscope signal, where gains increase with model depth. The structured temporal nature of gait likely allows SPT to initialize attention layers in alignment with recurring movement motifs, explaining the substantial improvements observed at two and three layers.

**Stress (Binary Classification).** For the accelerometer-based stress dataset, baseline performance is already relatively strong due to clear motion-related stress patterns. SPT yields consistent but moderate improvements (typically 2–4%), particularly at shallow depths. This suggests that SPT primarily stabilizes early optimization in moderately structured signals.

In contrast, the HR-based stress dataset presents smoother and more weakly structured temporal dynamics. Here, improvements are smaller (1–2.5%) and become more apparent at higher depths. The limited temporal landmarks in HR signals constrain the representational benefit of SPT, though performance remains consistently non-degrading.

**Parkinson (Binary Classification).** The Parkinson datasets display a distinctive behavior. Unlike other datasets, models trained from scratch benefit substantially from increased depth. Accuracy improves markedly from one to three layers even without SPT initialization. This suggests that Parkinson gait signals contain sufficiently rich temporal irregularities that deeper Transformers can exploit even under random initialization.

Nevertheless, SPT still provides significant improvements at intermediate depth (two layers), particularly for the LeftFoot dataset, where gains exceed 6% with high statistical significance. At three layers, improvements diminish, indicating that increased depth alone already captures much of the relevant temporal structure. In this setting, SPT primarily accelerates convergence and stabilizes optimization rather than fundamentally altering the representational capacity.

**Depth-Dependent Behavior.** Figure 5.2 shows that the average improvement provided by SPT increases from one to two layers and stabilizes thereafter. This suggests that SPT interacts synergistically with moderate model depth: shallow models lack sufficient capacity to fully exploit structured initialization, while deeper models already capture temporal dependencies effectively, reducing marginal gains.

Importantly, SPT never degrades performance relative to training from scratch. Its benefit is most pronounced in signals with strong periodic or quasi-periodic temporal structure (e.g., gait), moderate in movement-based stress signals, and smaller in physiologically smoother signals such as heart rate. These findings indicate that SPT acts as a structured inductive prior whose impact scales with both signal regularity and model capacity.

Table 5.2: From-scratch vs. SPT performance per dataset and transformer depth, including percentage improvements. Statistical significance (computed on accuracy) is denoted by asterisks: * $p \leq 0.05$, ** $p \leq 0.01$, *** $p \leq 0.001$, **** $p \leq 0.0001$.

| 2*Dataset | 2*Layers | From Scratch | | | | SPT | | | | Improvement (%) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Acc. | F1 | Rec. | Spec. | Acc. | F1 | Rec. | Spec. | ΔAcc | ΔF1 | ΔRec | ΔSpec |
| 3*Camargo, Gyr Y (Foot) | 1 | 0.532±0.04 | 0.533 | 0.521 | 0.538 | 0.551 | 0.557 | 0.560 | 0.551 | 1.9* | 2.4 | 3.9 | 1.3 |
| | 2 | 0.525±0.05 | 0.532 | 0.522 | 0.527 | 0.578 | 0.582 | 0.579 | 0.572 | 5.3*** | 4.7 | 5.4 | 4.5 |
| | 3 | 0.524±0.03 | 0.527 | 0.524 | 0.523 | 0.623 | 0.629 | 0.617 | 0.620 | 9.9**** | 10.2 | 9.3 | 9.7 |
| 3*Camargo, Gyr Z (Foot) | 1 | 0.565±0.03 | 0.547 | 0.566 | 0.545 | 0.577 | 0.575 | 0.578 | 0.578 | 1.2 | 2.8 | 1.2 | 3.2 |
| | 2 | 0.575±0.03 | 0.574 | 0.578 | 0.570 | 0.603 | 0.598 | 0.596 | 0.607 | 2.8* | 2.4 | 1.8 | 3.7 |
| | 3 | 0.573±0.04 | 0.569 | 0.566 | 0.569 | 0.611 | 0.610 | 0.614 | 0.605 | 3.8** | 4.1 | 4.8 | 3.6 |
| 3*Stress AccY | 1 | 0.748±0.03 | 0.747 | 0.748 | 0.725 | 0.789 | 0.796 | 0.783 | 0.772 | 4.1** | 4.9 | 3.5 | 4.6 |
| | 2 | 0.772±0.02 | 0.781 | 0.775 | 0.754 | 0.796 | 0.800 | 0.795 | 0.784 | 2.4 | 1.9 | 2.3 | 3.0 |
| | 3 | 0.780±0.03 | 0.793 | 0.781 | 0.778 | 0.810 | 0.816 | 0.808 | 0.800 | 3.0* | 2.3 | 2.8 | 2.2 |
| 3*Stress HR | 1 | 0.713±0.03 | 0.730 | 0.718 | 0.708 | 0.720 | 0.738 | 0.723 | 0.713 | 0.7 | 0.8 | 0.5 | 0.5 |
| | 2 | 0.719±0.06 | 0.724 | 0.716 | 0.711 | 0.735 | 0.744 | 0.730 | 0.723 | 1.6 | 2.0 | 1.4 | 1.2 |
| | 3 | 0.697±0.05 | 0.698 | 0.683 | 0.690 | 0.722 | 0.728 | 0.718 | 0.717 | 2.5* | 3.0 | 3.6 | 2.8 |
| 3*Parkinson (Left Foot) | 1 | 0.715±0.04 | 0.657 | 0.705 | 0.713 | 0.728 | 0.700 | 0.712 | 0.729 | 1.3 | 4.3 | 0.7 | 1.6 |
| | 2 | 0.745±0.03 | 0.738 | 0.740 | 0.745 | 0.813 | 0.809 | 0.813 | 0.818 | 6.8**** | 7.1 | 7.3 | 7.3 |
| | 3 | 0.823±0.01 | 0.815 | 0.819 | 0.822 | 0.850 | 0.848 | 0.857 | 0.854 | 2.7 | 3.3 | 3.8 | 3.2 |
| 3*Parkinson (Right Foot) | 1 | 0.742±0.01 | 0.724 | 0.730 | 0.741 | 0.767 | 0.739 | 0.763 | 0.764 | 2.5 | 1.5 | 3.3 | 2.3 |
| | 2 | 0.753±0.03 | 0.731 | 0.751 | 0.757 | 0.806 | 0.789 | 0.805 | 0.801 | 5.3*** | 5.8 | 5.4 | 4.4 |
| | 3 | 0.834±0.02 | 0.831 | 0.837 | 0.831 | 0.842 | 0.839 | 0.841 | 0.842 | 0.8 | 0.8 | 0.4 | 1.1 |

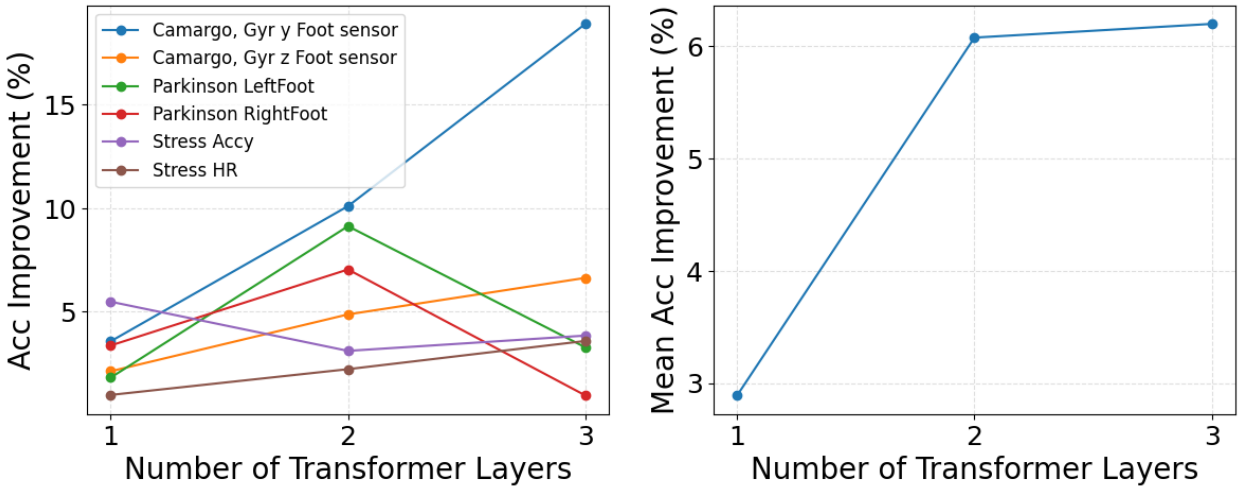


Figure 5.2: On the left the effect of SPT per dataset, on the right the average

Multivariate Analysis

**Experimental Design.** The multivariate experiments evaluate the Transformer when trained jointly on all available sensor channels, allowing the model to capture both temporal dynamics and cross-channel dependencies. For each dataset (Camargo (Entire Foot Sensor), Stress (Accy + HR), and Parkinson (Right Force + Left Force)), we compare training from scratch (Xavier initialization) with four SPT variants: *SPT-Point*, *SPT-Block*, *SPT-Column*, and *SPT-Mixed*. Models are evaluated at depths of 1, 2, and 3 encoder layers.

During pre-training, all SPT models are optimized using a **Masked Mean Squared Error (Masked MSE)** loss, computed exclusively over masked elements. This ensures that the model is explicitly trained to reconstruct missing values rather than copying visible inputs. Downstream fine-tuning uses supervised cross-entropy loss.

The quantitative results are reported in Table 5.3.

**Global Observations.** Across all three datasets, SPT variants consistently outperform training from scratch. However, the magnitude and scaling of improvements depend on: (i) the intrinsic temporal structure of the task, (ii) the redundancy and synchronization across channels, (iii) the structural bias imposed by the masking strategy.

Camargo and Parkinson exhibit strong periodic structure (gait cycles), while Stress contains heterogeneous physiological signals (EDA, HR, accelerometry) with weaker cross-channel alignment. These structural differences explain the differentiated impact of each masking policy.

#### SPT-Point

SPT-Point applies independent masking at the scalar level. It resembles multivariate denoising autoencoding and primarily reinforces robustness to local perturbations.

In Table 5.3, SPT-Point yields consistent but moderate improvements. The gains are particularly visible in Camargo (depth 2: $0.828 \rightarrow 0.848$, $p \leq 0.05$), where local periodic structure dominates but cross-channel alignment is imperfect.

However, improvements do not scale aggressively with depth, as shown in Fig. 5.3. This reflects the fact that pointwise masking does not explicitly enforce structured temporal or cross-channel reasoning.

#### SPT-Block

SPT-Block masks contiguous temporal segments, requiring the model to reconstruct extended time intervals. This directly encourages long-range temporal modeling.

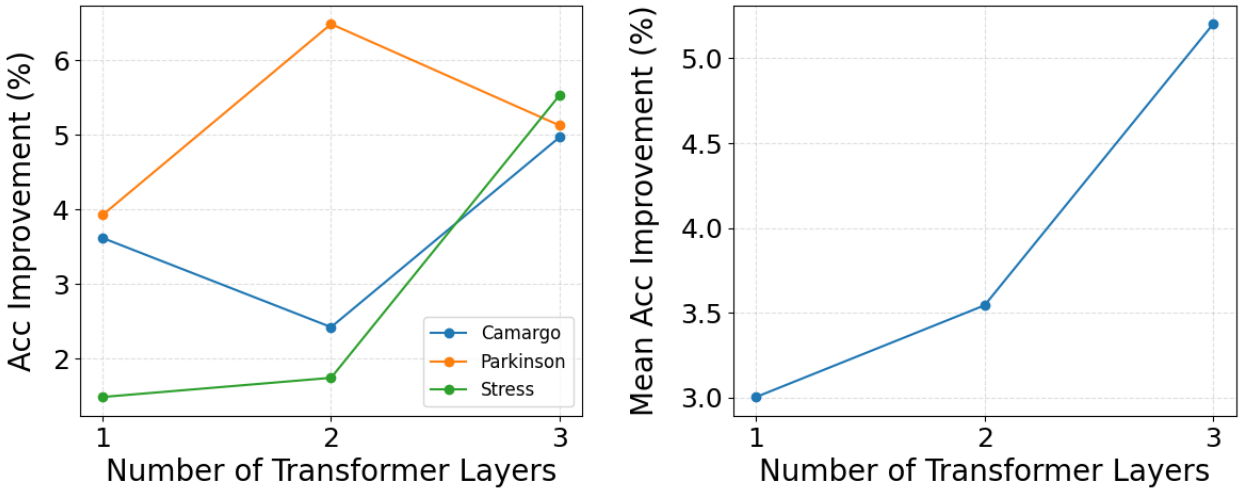


Figure 5.3: SPT-Point — improvement vs transformer depth. Left per dataset, right average. Improvements grow modestly with scale.

As shown in Table 5.3, SPT-Block performs particularly well in temporally structured datasets such as Parkinson. At depth 2, accuracy reaches 0.964 ($p \leq 0.001$), substantially exceeding from-scratch training.

The scaling behavior in Fig. 5.4 reveals that improvements peak at intermediate depth. This suggests that block masking efficiently encodes dominant temporal structure early, but additional layers do not proportionally increase the benefit.

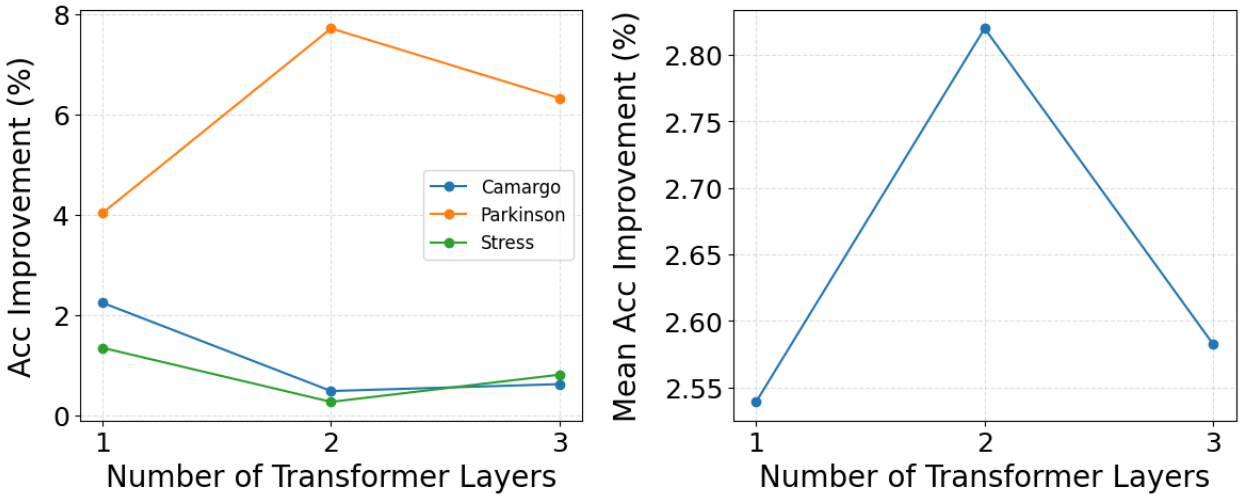


Figure 5.4: SPT-Block — strongest improvements at mid scale, limited benefit beyond 2 layers. Left per dataset, right average

### SPT-Column

SPT-Column removes entire feature channels during pre-training. This forces reconstruction via cross-channel inference and explicitly promotes sensor fusion.

In Table 5.3, SPT-Column demonstrates strong gains in Camargo and Parkinson, both of which contain synchronized multi-sensor measurements. At depth 3 in Camargo, accuracy reaches 0.869 ($p \leq 0.001$).

Figure 5.5 shows the strongest depth-dependent growth among all strategies. This indicates that deeper Transformers are particularly effective at exploiting structured cross-channel priors introduced by column masking.

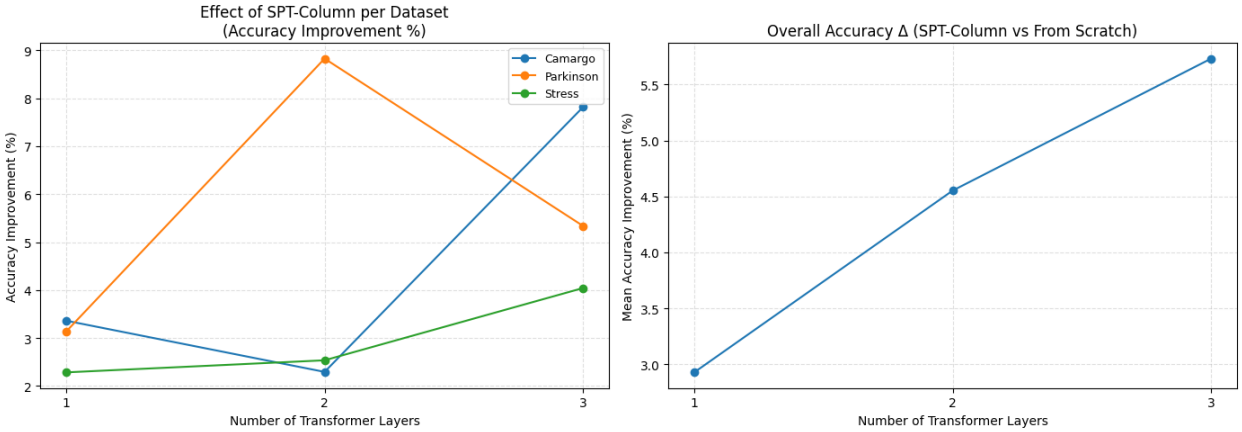


Figure 5.5: SPT-Column — the strongest and most consistent scaling with transformer depth. Left per dataset, right average

### SPT-Mixed

SPT-Mixed combines point, block, and column masking into a unified objective. This hybrid strategy introduces uncertainty across multiple structural axes simultaneously.

As shown in Table 5.3, SPT-Mixed provides the most stable improvements across heterogeneous datasets, particularly in Stress. At depth 3, accuracy reaches 0.788 ($p \leq 0.01$), outperforming individual masking strategies.

The scaling behavior in Fig. 5.6 varies by dataset. Parkinson peaks at intermediate depth (due to already strong baseline structure), while Stress continues improving with depth, reflecting its heterogeneous multimodal composition.

**Interpretation.** Together, these results demonstrate that the effectiveness of SPT depends on the alignment between masking structure and task structure.

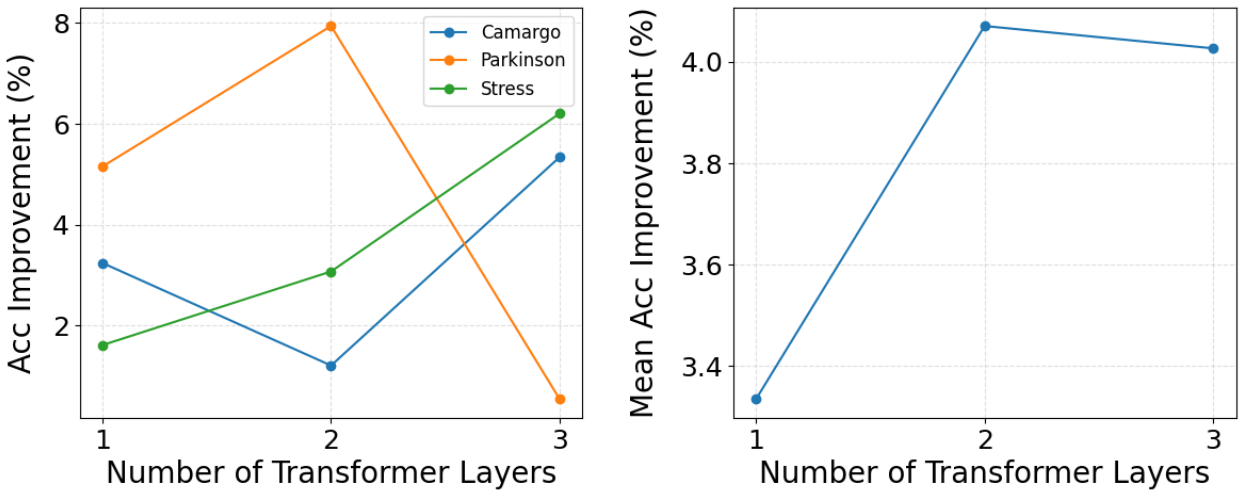


Figure 5.6: SPT-Mixed — strongest in heterogeneous settings, scaling depends on dataset structure. Left per dataset, right average

- In strongly periodic datasets (Parkinson, Camargo), block and column masking are particularly effective because they align with temporal and cross-channel regularities.
- In heterogeneous physiological datasets (Stress), mixed masking performs best because no single structural prior dominates.

As model depth increases, the performance gap between SPT and from-scratch models widens. This indicates that structured pre-training improves optimization stability and allows deeper Transformers to better exploit their representational capacity without overfitting.

### Optimization Perspective

From an optimization perspective, SPT operates as a structured warm-start mechanism. When transformers are initialized randomly, especially with small or noisy datasets, they must discover temporal structure from scratch, often becoming trapped in poor local minima. SPT mitigates this by introducing a reconstruction-based pre-training phase that aligns the parameters toward a useful manifold. This lowers gradient variance during fine-tuning and directs attention maps toward meaningful temporal and cross-modal patterns. Importantly, SPT reduces loss landscape sharpness, a property associated with better generalization. By pre-training on a reconstruction task without labels, models converge more smoothly during downstream optimization, particularly when limited labeled examples are available — a common condition in clinical applications.

Altogether, these results show that SPT provides a consistently positive impact that scales with model depth and remains robust across downstream tasks. It emerges as a competitive alternative to external self-supervised pre-training methods, with the key advantage that it requires no additional data beyond the target dataset itself. This has critical implications for clinical AI, where labeled data is scarce and generalization across domains remains challenging.

Despite its strengths, the approach is not without limits. The degree of improvement varies based on the inherent temporal structure of the data, and the masking strategies perform differently depending on the presence of cross-channel correlations. Furthermore, the study focuses on classification tasks using fixed-length windows. Future work should investigate how SPT performs on forecasting, detection, and segmentation tasks, especially with variable-length inputs or irregularly sampled data — common characteristics in medical contexts. Exploring the full architecture design space, including width and number of attention heads, could further deepen our understanding of how SPT shapes learning dynamics.

In conclusion, SPT stands as a powerful, dataset-aligned initialization method that brings significant and consistent improvements across real-world clinical signals. It decouples temporal representation learning from label dependency, yields scalable benefits as transformer depth increases, and adapts across modalities and tasks. By structuring model initialization in ways that reflect signal modality and temporal dynamics, SPT addresses key challenges in medical machine learning, offering a highly practical pathway toward more intelligent, robust, and data-efficient clinical models.

Table 5.3: Multivariate performance comparison between From-scratch training and SPT variants. Only Accuracy reports mean ± standard deviation. Statistical significance is denoted by asterisks (* $p \leq 0.05$, ** $p \leq 0.01$, *** $p \leq 0.001$, **** $p \leq 0.0001$).

| Dataset | Layers | From Scratch | | | | SPT-Point | | | | SPT-Block | | | | SPT-Column | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Acc. | F1 | Rec. | Spec. | Acc. | F1 | Rec. | Spec. | Acc. | F1 | Rec. | Spec. | Acc. | F1 | Rec. | Spec. |
| 3*Camargo | 1 | 0.803 ± 0.066 | 0.842 | 0.816 | 0.819 | 0.832 ± 0.056 | 0.850 | 0.832 | 0.835 | 0.821 ± 0.059 | 0.845 | 0.834 | 0.826 | 0.830 ± 0.050 | 0.844 | 0.826 | 0.838 |
| | 2 | 0.828 ± 0.072 | 0.857 | 0.821 | 0.836 | 0.848 ± 0.059 * | 0.865 | 0.855 | 0.851 | 0.832 ± 0.063 | 0.857 | 0.836 | 0.841 | 0.847 ± 0.065 * | 0.866 | 0.848 | 0.850 |
| | 3 | 0.806 ± 0.075 | 0.810 | 0.806 | 0.802 | 0.846 ± 0.054 ** | 0.862 | 0.843 | 0.851 | 0.811 ± 0.075 | 0.816 | 0.801 | 0.808 | 0.869 ± 0.056 *** | 0.865 | 0.849 | 0.851 |
| 3*Stress | 1 | 0.744 ± 0.063 | 0.732 | 0.746 | 0.766 | 0.755 ± 0.063 | 0.749 | 0.757 | 0.772 | 0.754 ± 0.063 | 0.745 | 0.754 | 0.765 | 0.761 ± 0.063 | 0.732 | 0.766 | 0.783 |
| | 2 | 0.749 ± 0.058 | 0.747 | 0.753 | 0.762 | 0.762 ± 0.062 | 0.757 | 0.762 | 0.780 | 0.751 ± 0.056 | 0.744 | 0.751 | 0.761 | 0.768 ± 0.054 * | 0.763 | 0.770 | 0.774 |
| | 3 | 0.742 ± 0.056 | 0.744 | 0.749 | 0.765 | 0.783 ± 0.051 * | 0.760 | 0.784 | 0.799 | 0.748 ± 0.062 | 0.737 | 0.748 | 0.759 | 0.772 ± 0.045 | 0.774 | 0.765 | 0.783 |
| 3*Parkinson | 1 | 0.893 ± 0.047 | 0.889 | 0.883 | 0.898 | 0.928 ± 0.025 * | 0.926 | 0.930 | 0.918 | 0.929 ± 0.032 * | 0.927 | 0.930 | 0.922 | 0.921 ± 0.044 * | 0.922 | 0.923 | 0.920 |
| | 2 | 0.895 ± 0.055 | 0.893 | 0.896 | 0.888 | 0.953 ± 0.027 ** | 0.954 | 0.947 | 0.954 | 0.964 ± 0.041 *** | 0.968 | 0.961 | 0.957 | 0.974 ± 0.043 **** | 0.968 | 0.971 | 0.961 |
| | 3 | 0.918 ± 0.032 | 0.909 | 0.914 | 0.916 | 0.965 ± 0.041 ** | 0.959 | 0.967 | 0.964 | 0.976 ± 0.038 *** | 0.969 | 0.977 | 0.973 | 0.967 ± 0.026 ** | 0.962 | 0.968 | 0.959 |

## 5.5 Conclusion

In this chapter, Self-PreTraining (SPT) has been introduced and evaluated as an initialization strategy for transformer-based models applied to medical time series classification. Unlike

supervised pretraining approaches or methods that rely on external datasets, SPT leverages only the data available in the target task through an unsupervised reconstruction objective. By masking and reconstructing segments of the input signals, SPT imposes a temporal and structural prior that shapes the learning dynamics of transformer models prior to task-specific fine tuning.

The experimental analysis conducted in this chapter spans heterogeneous medical datasets, including inertial gait signals, physiological stress markers, and multimodal Parkinsonian movement data. Across these diverse domains, SPT improves performance compared to training from scratch. These improvements are particularly evident in deeper transformer architectures, where SPT mitigates common optimization challenges such as unstable convergence and limited generalization. The results indicate that self-pretrained initialization enables deeper models to more effectively exploit their representational capacity, even in settings characterized by limited and noisy labeled data.

This chapter further highlights that the effectiveness of SPT depends on the structure of the self-supervised objective. The comparative analysis of different masking strategies shows that no single approach is universally optimal: point-wise masking promotes local robustness, block-wise masking captures temporal continuity, and column-wise masking exploits cross-sensor dependencies. The mixed masking strategy integrates these complementary properties, providing a flexible initialization scheme that adapts well to heterogeneous medical signals.

The findings presented in this chapter establish SPT as a practical and domain-aligned method for enhancing transformer-based models in clinical applications. Its ability to improve performance without requiring architectural modifications or additional data makes it particularly suitable for medical machine-learning scenarios, where labeled data is often scarce. Future research directions include extending this framework to other time series tasks such as forecasting, imputation, and segmentation, as well as investigating its effectiveness on irregularly sampled or asynchronous clinical signals. By enabling more robust and data-efficient learning, SPT represents a promising pathway toward the deployment of reliable transformer-based models in real world healthcare settings.

# Chapter 6

# Conclusion

In this thesis, we advanced the field of artificial intelligence applied to healthcare, with particular attention to deep learning methods for human locomotion analysis and to self-pretraining strategies aimed at improving diagnostic performance. The contributions presented in Chapters 2 and 3 address the research questions related to artificial intelligence for lower limb rehabilitation robotics, while the work presented in Chapters 4 and 5 addresses the research questions concerning Self-PreTraining for medical time series analysis.A concise synthesis of these contributions and their relationship to the research questions is provided in Box 6.1 and Box 6.2, respectively.

In Chapter 2, this thesis provided an extensive and systematic review of artificial intelligence–based methodologies applied to lower limb exoskeleton assisted rehabilitation. The chapter examined the state of the art in machine learning and deep learning techniques used across key rehabilitation robotics tasks, including gait analysis, locomotion classification, intention detection, trajectory prediction, and adaptive control. A particular emphasis was placed on how AI models are integrated within exoskeleton systems to improve human–robot interaction, personalization of therapy, and real time adaptability to patient needs. The review highlighted the progressive shift from traditional rule-based and model-driven approaches toward data-driven and learning-based solutions capable of capturing the intrinsic variability and nonlinearity of human locomotion. Furthermore, the chapter compared different classes of algorithms—ranging from classical machine learning to deep neural architectures—and discussed their respective advantages and limitations in clinical and real world settings. By critically analyzing current literature, the chapter identified key open challenges, such as limited data availability, poor generalization across subjects and pathologies, lack of interpretability, and difficulties in clinical translation. These findings form the foundation of the contributions summarized in Box 6.1.

In Chapter 3, the thesis presented an in-depth experimental study focused on deep learn-

ing techniques for human locomotion analysis in lower limb exoskeleton applications. The chapter systematically evaluated multiple neural network architectures and sensing configurations on both classification and regression tasks, including terrain recognition, slope inclination estimation, and stair height prediction. By leveraging multimodal inertial data and comparing different model families, the study provided a detailed assessment of how architectural choices and sensor placement affect performance, robustness, and computational efficiency. Beyond raw performance metrics, the chapter incorporated explainability analyses to better understand the internal decision-making processes of the models and to identify the most informative sensors and features for each task. These results provide experimental evidence supporting the conclusions summarized in Box 6.1.

In Chapter 4, the thesis investigated self-pretraining as a strategy to improve sequence classification performance, with the goal of understanding its underlying mechanisms and practical benefits. The chapter introduced and analyzed different training pipelines, comparing standard supervised learning with self-pretrained initialization under controlled experimental conditions. Through extensive evaluations across datasets, model depths, and training configurations, the study examined how self-pretraining influences convergence behavior, representation quality, and generalization performance. Particular attention was given to the impact of pretraining duration, parameter initialization, and layer-wise freezing strategies. The insights derived from this analysis directly inform the methodological conclusions summarized in Box 6.2.

Finally, in Chapter 5, the thesis extended the investigation of self-pretraining to the domain of medical time series using transformer-based architectures. This chapter addressed the question of whether self-pretraining remains beneficial when applied to more expressive and scalable models, as well as to heterogeneous and clinically relevant datasets. Different masking strategies were systematically explored to assess how the structure of the self-supervised objective interacts with the characteristics of medical signals. The experimental results demonstrated that self-pretraining can provide statistically significant performance improvements across multiple medical time series tasks, although its effectiveness depends on factors such as dataset size, signal structure, and masking policy. Together with the findings of Chapter 4, these results complete the set of contributions summarized in Box 6.2.

To clearly summarize how these contributions address the research questions introduced in Chapter 1, the main outcomes of the thesis are synthesized in Box 6.1 and Box 6.2. Each box provides a concise, bullet-point overview linking the research objectives to the corresponding methodological and experimental results, highlighting both theoretical insights and practical implications.

**Box 6.1: Answer to Research Question 1: AI for lower limb Rehabilitation Robotics**

**Research Question 1:**
*What artificial intelligence approaches have been developed for lower limb rehabilitation robotics, and how can deep learning architectures and sensor configurations be advanced to improve human locomotion understanding?*

**Related chapters:** Chapters 2 and 3

- Provided a comprehensive and systematic review of AI-based methodologies applied to lower limb rehabilitation robotics, covering gait analysis, locomotion classification, intention detection, trajectory prediction, and adaptive control.
- Identified a clear shift from traditional model-based approaches toward data-driven and deep learning solutions capable of handling the variability and non-linearity of human locomotion.
- Experimentally demonstrated that deep sequence models can accurately capture spatiotemporal locomotion patterns using wearable sensor data.
- Evaluated multiple neural architectures and sensing configurations, highlighting the trade-off between performance, robustness, and sensor complexity.
- Showed that explainability techniques can be used to identify the most informative sensors and features, providing practical guidelines for efficient and clinically viable exoskeleton design.

**Box 6.2: Answer to Research Question 2: Self-PreTraining for Diagnosis and Rehabilitation Robotics**

**Research Question 2:**
*How can Self-PreTraining be exploited to improve transformer-based models for diagnosis and rehabilitation robotics, and how well does it generalize across heterogeneous medical time series data?*

**Related chapters:** Chapters 4 and 5

- Investigated the theoretical and empirical foundations of Self-PreTraining, explaining why it enables transformer models to learn long range temporal depen-

dencies more effectively.

- Demonstrated that Self-PreTraining reshapes the optimization landscape, improving training stability and generalization compared to random initialization.
- Proposed and systematically evaluated multiple masking strategies (Point, Block, Column, and Mixed), highlighting how each exploits different temporal and cross-channel structures.
- Showed that Self-PreTraining improves transformer performance across diverse medical domains, including rehabilitation robotics, stress detection, and Parkinson's disease monitoring.
- Established Self-PreTraining as a practical, data-efficient initialization strategy that requires no external datasets or architectural modifications, making it well suited for real world clinical applications.

**Future Directions**

While this thesis has demonstrated the potential of artificial intelligence–based methodologies for lower limb rehabilitation robotics through extensive offline analysis and experimental evaluation, a fundamental next step lies in the online validation of these systems in real world robotic and clinical settings. The translation of AI models from controlled, offline experimental conditions to real time deployment on physical rehabilitation robots represents a critical challenge and an essential direction for future research. Only through online validation can the true clinical relevance, robustness, and safety of the proposed approaches be fully assessed.

A primary future direction concerns the integration of the developed AI models into real lower limb exoskeletons and rehabilitation robots operating in closed-loop control. Offline evaluations, while necessary for benchmarking performance and understanding model behavior, do not fully capture the complexities of real world human–robot interaction. In online scenarios, AI models must operate under strict real time constraints, processing high frequency sensor data, handling delays, and responding promptly to changes in user behavior or environmental conditions. Future work should therefore focus on embedding the proposed perception, classification, and prediction modules directly into robotic control architectures, ensuring compatibility with onboard hardware, communication protocols, and safety mechanisms.

Another crucial aspect of online validation is robustness to real-life variability. In labo-

ratory datasets, signals are often collected under standardized conditions, with limited noise and controlled experimental protocols. In contrast, real world rehabilitation settings introduce substantial sources of variability, including sensor drift, motion artifacts, changes in sensor placement, patient fatigue, and unexpected user behaviors. Future studies must evaluate how AI-driven locomotion analysis and self-pretrained models perform under such non-ideal conditions. This includes long-term testing across multiple sessions, days, and users, allowing models to be assessed not only for accuracy but also for stability, reliability, and failure modes over time.

Clinical validation represents an equally important future direction. For AI-based rehabilitation systems to be adopted in clinical practice, they must demonstrate measurable benefits for patients and clinicians. Future work should involve structured clinical trials in collaboration with rehabilitation centers, where AI-enabled exoskeletons are evaluated during real therapy sessions. These studies should assess outcomes such as functional recovery, gait quality, patient engagement, and therapist workload. Moreover, qualitative feedback from clinicians and patients should be incorporated to refine model behavior, interface design, and decision-making logic, ensuring that AI systems support rather than hinder the rehabilitation process.

Another promising avenue for future research lies in adaptive and personalized online learning. While this thesis explored self-pretraining and representation learning in offline settings, real world deployment opens the possibility of continual adaptation during use. Future systems could leverage online or semi-online learning strategies to adapt models to individual patients over time, accounting for changes in motor capabilities, rehabilitation progress, or fatigue levels. Such adaptive systems would require careful design to ensure safety and stability, but they could significantly enhance personalization and long-term effectiveness of robot-assisted therapy.

The validation of explainability and transparency in online settings also deserves attention. While explainability methods were used offline to analyze model behavior, future work should investigate how interpretable AI outputs can be incorporated into real time robotic systems. Providing clinicians with meaningful, real time insights into model decisions—such as why a certain gait mode or assistance level was selected—could improve trust, facilitate clinical acceptance, and support decision-making during therapy sessions. Evaluating the usefulness and clarity of such explanations in real clinical workflows is an important step toward human centered AI in rehabilitation robotics.

From a systems perspective, future research should also address the co-design of AI models and robotic hardware. Online validation may reveal trade-offs between model complexity, computational load, power consumption, and responsiveness. This may motivate

the development of lightweight or hardware aware AI architectures specifically optimized for embedded robotic platforms. Additionally, hybrid control strategies that combine AI-based perception with model-based or impedance control could improve safety and reliability, particularly during unexpected events or model uncertainty.

Finally, ethical and regulatory considerations will play an increasingly important role in future developments. Online deployment of AI-driven rehabilitation robots involves direct physical interaction with vulnerable users, making safety, accountability, and data privacy paramount. Future work should therefore include systematic evaluation of risk, fail-safe mechanisms, and compliance with medical device regulations. Establishing standardized protocols for online testing and validation will be essential to ensure reproducibility, comparability, and regulatory approval of AI-enabled rehabilitation technologies.

In summary, while this thesis has laid the methodological and experimental foundations for AI applications in lower limb rehabilitation robotics, the ultimate validation of these systems must occur online, on real robots, and in real-life clinical environments. Future research should prioritize the transition from offline benchmarks to real world deployment, focusing on robustness, adaptability, clinical impact, and user-centered design. Addressing these challenges will be essential for transforming AI-based rehabilitation robotics from promising research prototypes into reliable and effective clinical tools.

bibliography.bib