UQAC
Université du Québec
à Chicoutimi

HOW CAPABLE ARE LARGE LANGUAGE MODELS AT HUMAN ACTIVITY RECOGNITION WITHIN A SMART HOME?

BY JEAN-PHILIPPE LAROUCHE

MASTER'S THESIS PRESENTED TO UNIVERSITÉ DU QUÉBEC À CHICOUTIMI IN PARTIAL FULFILLMENT OF THE REQUIERMENTS FOR THE DEGREE OF MASTER OF SCIENCE IN THE SUBJECT OF COMPUTER SCIENCE

QUÉBEC, CANADA

RESUME

La population mondiale vieillit et cette tendance est prévue de se poursuivre au cours des prochaines décennies. Ce phénomène est d'autant plus marqué dans les sociétés occidentales, où les taux de naissance ont connu une forte baisse. Cela a entraîné une augmentation du nombre de personnes agées qui sont plus fréquement atteintes de problèmes de santé nécessitant un support dans l'accomplissement de leurs tâches quotidiennes, menaçant ainsi leur autonomie et accroissant la pression sur les services de santé et les familles. Pour tenter d'atténuer ce problème, les maisons intelligentes sont devenues une option de plus en plus intéressante par rapport à l'intervention humaine directe. Équipées d'un ensemble de capteurs et couplées à l'apprentissage automatique, les maisons intelligentes offrent une grande variété d'options d'assistance et contribuent même à la médecine préventive. Au cœur de cette approche se trouve la reconnaissance d'activités humaines (RAH), un concept qui s'avère être un problème complexe. La RAH est nécessaire pour être en mesure de soutenir adéquatement les personnes nécessitant de l'assistance dans un habitat intelligent. Malheureusement, non seulement elle implique une très longue liste d'activités possibles, mais elle souffre également d'une forte variabilité intra-sujet et inter-sujet. Cela complique considérablement l'entraînement de modèles capables d'accomplir avec précision les tâches de RAH. Les *Large Language Models* (LLM) ont dominé la recherche et l'attention médiatique ces dernières années. Ce n'était donc qu'une question de temps avant qu'ils ne fassent leur entrée dans le domaine de la RAH. En effet, plusieurs études récentes font état de résultats prometteurs, particulièrement sur les jeux de données basés sur les capteurs IMU. Motivés par ces résultats optimistes, nous avons entrepris d'appliquer les LLM au sein de notre laboratoire d'habitat intelligent et à notre propre jeu de données avec l'objectif d'obtenir une méthodologie plus simple et une approche plus généralisable aux tâches de RAH. Nous avons ensuite exploré diverses approches pour améliorer les performances des LLM, telles que l'ingénierie d'invite (prompt engineering) et des techniques de prétraitement des données. Contrairement au discours dominant, nous avons constaté que l'utilisation des LLM pour la RAH est bien plus complexe et moins efficace que ce qui était suggéré initialement, même en suivant de les méthodologies publiées. Dans ce mémoire, nous décortiquons nos tentatives infructueuses, partageons ce qui a réellement fonctionné, et nous nous demandons si toute cette puissance des LLM en vaut vraiment la peine. Spoiler : parfois, un bon vieux Random Forest remporte encore la mise.

ABSTRACT

The world population is getting older, and it is expected to continue in this direction over the next few decades. It is even more present in Western societies, where birth rates have observed a sharp decline. This phenomenon has led to an increase in elderly individuals who are more affected by health conditions requiring assistance to perform daily tasks, threatening their autonomy and placing further strain on the health services and family. In an attempt to mitigate this, smart homes have become an increasingly interesting option over direct human intervention. Equipped with a suite of sensors and coupled with machine learning, smart homes offer a great variety of support options and even contribute to preventive medicine solutions. A core concept of this approach is Human Activity Recognition (HAR), which has shown itself to be a complex problem. HAR is necessary in order to be in a position to adequately support individuals requiring assistance in a smart home. Unfortunately, not only does it involve a very long list of activities, it also suffers from high intra-subject and inter-subject variability. This significantly complicates the training of models capable of accurately performing HAR tasks. Large Language Models (LLMs) have dominated both research and media attention in recent years; it was only a matter of time before they found their way into the domain of HAR. Indeed, several recent studies report promising results, particularly on IMU-based datasets. Motivated by these optimistic claims, we set out to apply LLMs within our smart home laboratory and to our own smart home dataset, hoping for a simpler methodology and more generalizable approach to HAR tasks. We then investigated various approaches to increase the performance of LLMs. Contrary to the prevailing narrative, we found that using LLMs for HAR is far more complex and less effective than initially suggested, even when following published methodologies. In this master's thesis, we unpack our experiments, share what actually worked, and reflect on whether all that LLM power is really worth the hassle. Spoiler: sometimes a good old Random Forest still wins the day.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACRONYMS

AoT Arrow of time

API Application Programming Interface

BiGRU Bidirectional gated recurrent unit

CNN Convolutional Neural Networks

CoT Chain of Thought

CRDV Centre de recherche sur le vieillissement

DL Deep Learning

GNAFCC Global Network for Age-friendly Cities and Communities

GPT Generative Pre-trained Transformers

HAR Human Activity Recognition

IMU Inertial Measurement Unit

INTER Ingénierie de technologies interactives en réadaptation

IT Information Technology

LIARA Laboratoire de recherche sur l'intelligence ambiante pour la reconnaissance d'activités

LLM Large Language Model

LSTM Long Short-Term Memory

MADA Municipalité amie des aînés

NLP Natural Language Processing

RGBD Red-Green-Blue-Depth

RMS Root mean square

SSH Secure Shell Protocol

SSL Self-supervised learning

SVM Support vector machine

UN United Nations

UQAC Université du Québec à Chicoutimi

UWB Ultra-wideband

WHO World Health Organization

FOREWORD

This thesis, titled "How capable are large language models at human activity recognition within a smart home?", is submitted in partial fulfillment of the requirements for the Master's degree in Computer Science at University of Québec à Chicoutimi.

The journey toward this research project began with a role as a software developer within a research center for the Thales Group. It is here that I had the amazing opportunity to work along side a brilliant team of scientists and engineers on research topics for the defense sector. This greatly contributed to my interest to pursue my own research and further develop my abilities. The topic for this research project came from a multitude of factors. Assisting Pascal E. Fortin as a research assistant with his work, continuous late night discussions with Kévin Bouchard, researcher at the LIARA laboratory, and the increasing adoption of large language models by the general public. Could large language models allow us to more easily perform human activity recognition tasks within our laboratory? Over the past year, exploring this question has been both a rigorous academic challenge and a deeply rewarding personal experience. Not only have I tremendously learned about what it takes to succeed as a researcher but I have also learned a lot about myself.

I would first like to thank my supervisors, Kévin Bouchard Ph.D, Sébastien Gaboury Ph.D and Pascal E. Fortin Ph.D, for their invaluable guidance, patience, support and for constantly pushing me to refine my methodology and think further. I would also like to thank the Thales Group and my team for their support and flexibility which allowed me to pursue this research project while retaining my role.

Finally, a special thanks to my partner, Dr. Charlotte Lepage-Pérusse for her continuous support throughout the late nights, complex scheduling arrangements and listening to more talk about models and human activity sensor data than she ever bargained for.

CHAPITRE I

INTRODUCTION

1.1 CONTEXT OF THE RESEARCH

It is a well-known phenomenon that the world population is ageing. According to the United Nations (UN), the proportion of the world's population over 60 will nearly double from 12% to 22% [65] between 2015 and 2050. The UN, through the World Health Organization (WHO), has declared this decade (2021-2030) the *The Decade of Healthy Ageing* [66] . The objective of this initiative is the creation of age-friendly environments and aligns with the UN 2030 agenda for Sustainable Development, a commitment by all member states to improve the lives of current and future generations of older people. One of the main action areas is the development of age-friendly cities and communities, of which eight domains have been identified, one of which is housing [67]. To assist in this mandate, the WHO has established the Global Network for Age-friendly Cities and Communities (GNAFCC) [68] with various branches throughout the world to inspire, connect, and support communities in this mission.

Within the province of Quebec, Canada, the issues associated with ageing are very present with 20% of the population over the age of 65 since 2021 and is expected to climb past 27% by

2041 [22]. In Canada, most old age statistics are calculated with the age of 65 as the lower number since this number is directly correlated to the retirement age. To better compare with the UN's data, we can use the information for each Canadian province from Statistics Canada [55]. With this information, we are able to calculate the current population 60 years of age and older in Quebec to be 28.6%, already well above the world average prediction of 22% by 2050 and higher than the current Canada-wide average of 26%.

In Quebec, more than 90% of the population over 65 years of age lives in a private home [23, Table 1.2.1, p. 28] and more than 75% suffer from a chronic illness [23, Table 5.1.3, p. 168]. Ageing comes with its share of conditions and illnesses that affect an individual's ability to perform daily living activities, such as cooking, maintaining personal hygiene, cleaning, etc. The loss of capacity to perform daily activities associated with illnesses directly affects these individuals autonomy and quality of life, even leading to the loss of their homes if their health degrades significantly. To help mitigate these issues, Quebec has its own chapter of WHO's GNAFCC called Municipalité amie des aînés (MADA) [69] which collaborates with Centre de recherche sur le vieillissement (CRDV), a research center within the Unviersité de Sherbrooke.

1.1.1 INTER AND THE LIARA LABORATORY

Computer science has had an impact on most fields through technological transfers, innovation in hardware and software, and wide-ranging applicability in various domains. This impact has become even more noticeable in the last decade with advancements in machine learning. The Ingénierie de technologies interactives en réadaptation (INTER) interdisciplinary strategic cluster was created within the University of Sherbrooke to increase collaboration between scientists in formal sciences, social sciences, and health sciences. Its focus is on habitats and

mobility, with each prioritizing two axes: 1. The conception and prototyping of new systems, and 2. The evaluation and transfer of new technologies. The objectives of INTER include: mobilizing scientists, supporting initiatives, providing tools and science-positive environments, and positioning research from Quebec universities on the international stage. This strategic cluster grew to include the majority of Quebec universities, including the Université du Québec à Chicoutimi (UQAC), which saw the creation of the Laboratoire de recherche sur l'intelligence ambiante pour la reconnaissance d'activités (LIARA) laboratory.

The LIARA is a computer science laboratory with a focus on developing smart habitats to support individuals with conditions that affect autonomy and who have difficulty performing daily living activities. The laboratory consists of an apartment-style habitat with a suite of sensors and technology to capture the data relevant to daily activities performed by the individual who inhabits it. The LIARA has contributed to multiple advances in Human Activity Recognition (HAR), especially in the development of deep learning models [34]. One of the objectives is to recognize daily activities performed by inhabitants in order to detect health changes, develop assisted living solutions [48] and improve preventive medicine through technological transfer.

1.1.2 HUMAN ACTIVITY RECOGNITION

Human activity recognition remains a formidable challenge in the domain of smart homes and assisted living [25]. The primary objective of HAR in these environments is to infer the activities of a habitant from sensor data [45]. The complexity of this task stems from the vast and heterogeneous nature of daily activities, which range from simple actions, such as walking or sleeping, to complex, multi-step activities like meal preparation or personal hygiene.

Furthermore, HAR systems must account for significant inter-subject variability, as different individuals perform the same activity in distinct ways, a challenge often exacerbated in populations with diverse physical or cognitive conditions. Additionally, models must accommodate intra-subject variability, where the same individual performs an activity differently depending on the context. For example, boiling pasta versus baking a cake yields drastically different data, despite both falling under the broad classification of *cooking*.

Data acquisition approaches for HAR within smart homes are generally divided into two primary modalities: wearable sensors (e.g. accelerometers and gyroscopes) [18] and ambient sensors (e.g. pressure sensors) [35]. Wearable sensors are frequently favored in experimental setups due to their cost-effectiveness, commercial availability, and ease of deployment. However, they are limited in the depth of spatial and contextual information they can capture. In contrast, ambient sensors mitigate these limitations by providing high-fidelity contextual data and precise spatial localization (e.g. specific positioning within a room), albeit at the expense of higher costs and more complex installations. Ultimately, optimal HAR architectures employ a combination of both approaches, integrating both wearable and ambient modalities to construct a robust and highly accurate representation of the resident's environment and actions.

1.1.3 LARGE LANGUAGE MODELS

Large Language Model (LLM)s, a transformer-based architecture, began demonstrating interesting potential in the late 2010s with the arrival of BERT in 2018 [13]. They gained popularity within the scientific community with the arrival of GPT-3 by OpenAI in mi-2020 [3], which was available through an Application Programming Interface (API). At the end of

2022, OpenAI decided to offer access to GPT-3 to the general public through a web interface called ChatGPT. With this release, LLMs became the face of artificial intelligence with the general public and saw a significant increase of interest in transformer-based models. With this popularity and promising generalization capability, many research projects have been launched across most scientific domains with the objective of determining their capabilities.

Being trained on a large text corpus of most of the world's public knowledge, sometimes claimed to be the *entire* public-facing internet, LLMs show impressive capabilities at correctly answering introductory and intermediate questions from most domains. This includes writing code, writing text from documents to complete stories, answering questions about daily life, etc. This enthusiasm and capability have led to changes in the habits of information search, with LLMs tending to replace traditional search engines [32] [5]. In the context of HAR, early works have demonstrated potential for LLMs to dynamically recognize human activities from sensor data without the need for finetuning or heavy data pre-processing [11] [24].

1.2 RESEARCH PROBLEM

Although deep learning models have advanced HAR in smart homes [28], these models have limitations. Training a deep learning model to identify a small subset of daily activities is a rigid, time-consuming process that is dependent on extensive, environment-specific training datasets. Furthermore, these architectures typically require complex data pre-processing pipelines and struggle to generalize across different residents or novel activities without substantial retraining. As a result, deploying and scaling these traditional models in real-world smart homes remains highly complex and a barrier to accessibility.

LLMs offer a promising alternative to overcome these barriers. By leveraging their extensive pre-trained reasoning, LLMs have the potential to significantly reduce the need for extensive data pre-processing and feature engineering. In the context of a smart home, they offer exceptional ease of use and lower the knowledge threshold required for deployment. Rather than requiring specialized machine learning expertise to train bespoke models from scratch, researchers and practitioners could potentially use natural language and sensor data to define and recognize a wide variety of human activities within a smart home dynamically and in real time.

However, despite recent studies demonstrating the capability of LLMs to recognize human activities, integrating them into practical smart home ecosystems introduces important challenges. The fundamental architecture of an LLM is designed to process natural language, whereas smart home sensors generate high-frequency, numerical raw data. Currently, attempting to bridge this gap by processing vast amounts of precise, raw numerical data through LLMs is computationally prohibitive. It necessitates ultra-high-performance computing infrastructure, leading to unaffordable costs and unrealistic deployment scenarios for a standard home environment. Therefore, a significant gap remains to find an efficient, cost-effective and user-friendly methodology to harness the semantic power of LLMs for activity recognition without incurring debilitating computational overhead.

Furthermore, the initial works analyzing the capabilities of LLMs have only analyzed public datasets with a small number of data-distinct activities. For these models to perform in real-world smart home scenarios, it is important to validate the capabilities of LLMs on datasets from a smart home with a wide variety of activities, with multiple representations for each participant. In adition, these datasets must contain a strong variety of participants.

These conditions will better evaluate the capabilities of the LLMs when faced with significant intra-variability and inter-variability, as this will be more representative of a real-world environment where the data is not *clean*.

1.3 CONTRIBUTION OF THE RESEARCH

In the context of this master's research project, two primary scientific contributions are brought forward. The first contribution brought forth is the validation of current approaches to HAR tasks using LLMs seen in the literature on a complex, previously unseen dataset. While existing literature has demonstrated theoretical and initial feasibility for LLMs to perform well as zero-shot human activity recognizers, these are often limited to online datasets [6] with limited distinct activities and with minimal noise. By evaluating these approaches against a previously unseen dataset characterized by a significantly greater volume of distinct activities with inter-subject and intra-subject variability, this research bridges the gap between initial capability demonstration and real-world applicability. This contribution provides a critical benchmark for the generalization and robustness of LLM-based HAR tasks in a dynamic, real-world smart home environment.

The second contribution lies in the identification and evaluation of techniques designed to enhance the performance of LLMs on HAR tasks when using Inertial Measurement Unit (IMU)-based data while maintaining a simple deployment strategy. Given the complexities of communicating high-frequency numerical sensor data in a manner suitable for natural language processing, this research investigates optimal strategies to maximize HAR performance while mitigating computational overhead. By analyzing various methods, such as prompt

engineering [42] or signal processing techniques [1], this work establishes a baseline on IMU data processing techniques for HAR tasks. Ultimately, this contribution provides guidelines for researchers and practitioners on how to effectively configure and deploy LLMs in the context of HAR to achieve optimal performance in smart home ecosystems.

1.4 RESEARCH METHODOLOGY

This research project is conducted following a scientific approach divided into four primary phases. The initial phase of this research involves the acquisition of knowledge of the following subjects: human activity recognition, large language models, prompt-engineering techniques and large language models on HAR tasks. This phase is achieved through a comprehensive review of the literature pertaining to all the aforementioned subjects. This is a crucial phase since it grants a better understanding of the domain through the identification of state-of-the-art approaches, limitations and previous research done on these topics.

The second phase consists of applying the most promising approaches identified in the literature review of phase one (1) on the LIARA dataset. The selection of the LIARA dataset provides a complex, highly variable dataset with more than a dozen activities and dozens of participants obtained from a realistic smart home environment. The objective of this phase is to establish a baseline performance metric on LLM approaches when subjected to high-variability data that is not found online, providing a quantitative starting point for the rest of this research project.

The third phase builds on the previous phase by evaluating optimization techniques aimed at increasing LLM performance on HAR tasks. This stage goes through an iterative refinement

process to address specific challenges of the use of LLMs for HAR while maintaining simplicity and numerical data. This includes prompt-engineering, IMU data pre-processing and curation methods and dimensionality reduction techniques, such as principal component analysis, to determine the best approaches to maximize the LLMs contextual reasoning capabilities. The objective is to determine the optimal performance scenario of a LLM on HAR classification activities when using numerical IMU sensor data in order to accurately compare with deep learning techniques.

The final phase of this research project is the comprehensive analysis of LLMs performance to determine the efficacy of using LLMs for HAR tasks and their potential for future work. The performance of the models on HAR tasks is evaluated between themselves, previous works and against established techniques in the field of HAR. This includes a comparison to the classical machine learning approach, the random forest. This comparative evaluation is not limited to classification accuracy. It assesses the trade-offs regarding data pre-processing requirements, model deployment complexity, and computational efficiency to evaluate the proposed advantages of leveraging LLMs in a smart home context and attempt to provide a global picture on their use.

1.5 ORGANIZATION OF THE THESIS

This masters thesis is composed of seven (7) chapters. The first chapter is an introduction to the research subjects, the context and the research problem. In adition, this chapter contains other relevant information about the project such as the methodology and anticipated scientific contributions.

The second chapter is a review of the literature. It covers multiple topics in order to clarify the different concepts this work is built upon. It begins with covering the extensive work done on human activity recognition, specifically the state-of-the-art deep learning approaches which showed great results. Following this is a section on the use LLMs for classification tasks and another section on the different prompt engineering techniques as this is the main form of communication with the LLMs. The final section pertains to the initial works done on using LLMs for HAR tasks, which heavily inspired this project.

The third chapter covers the experimental setup of this research project. This includes details of the physical and technical environment in which the research is conducted. Here, we include all the details about the LIARA laboratory, such as the layout and the specific sensing infrastructure used for data collection. This chapter also details the two (2) datasets used for the experiments. Finally, it enumerates the different models used and the decision behind their selection.

The fourth and fifth chapters cover the experiments carried out in this research project. The former covers the first half of the experiments done, which attempts to reproduce results seen in the literature review with the LIARA dataset, while the latter covers the second half of the experiments done where we attempt to increase the performance through various techniques. Each chapter details the objective, the experiments themselves, and the results before ending with a discussion around them.

The sixth chapter is a comprehensive discussion of the whole research project. Here, we discuss the different challenges that were encountered, new research that has emerged

throughout the project, our analysis of the current state of things and, importantly, what happened when we introduced a traditional machine learning approach.

The seventh and final chapter is the conclusion of the thesis. This is where we close our thoughts, identify the contributions and outline directions for future work on this topic.

CHAPITRE II

LITERATURE REVIEW

The second chapter of this master's thesis consists in the exploration of the various scientific works that are related to and inspired the research project. To begin, we present the state-of-the-art approaches to human activity recognition found in the literature. Following this, we look at the different techniques and use cases of large language models. In the last section, we take a closer look at the most recent works applying LLMs to HAR tasks. Most of the articles listed in this chapter were identified through the *Sofia* academic search tool, a search tool offered by the library of UQAC and shared between all universities in Quebec.

2.1 HUMAN ACTIVITY RECOGNITION

The field of HAR has undergone a significant paradigm shift from traditional machine learning models, which relied heavily on handcrafted feature engineering, to Deep Learning (DL) architectures capable of automated feature extraction [14]. Although DL models such as Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks have demonstrated state-of-the-art performance on lab-controlled datasets [59] [72], they often struggle with generalization in real-world free-living environments and are constrained by

the scarcity of large-scale labeled datasets. This results in models trained to perform well in laboratory scenarios on a small set of well-defined activities. They generally must be re-trained and adapted in order to be applied in a new environment or when new activities are introduced.

One of the most significant recent advancements in addressing the dataset bottleneck in HAR is the work of Yuan et al. (2024), titled "Self-supervised learning for human activity recognition using 700,000 person-days of wearable data" [71]. This study marks a transition from small-scale supervised learning to the development of a model that performs HAR tasks using large scale wearable sensor data and is able to generalize. The authors leveraged the UK Biobank dataset, encompassing over 700 000 person-days of unlabeled accelerometer data, one of the largest of its kind to date. To gather data, over 100 000 participants wore a wrist-based accelerometer for more than seven (7) days in their usual environment.

To utilize this massive unlabeled corpus, the authors implemented a Self-supervised learning (SSL) framework. They begin by performing weighted sampling in proportion to the standard deviation to give more weight to high movement periods. Additionally, they apply a multi-task SSL as a data augmentation technique. The SSL strategy involves three simple but effective transformation-recognition tasks: Arrow of time (AoT), permutation and time warping. AoT flips the signal along the time axis, permutation breaks the signal into chunks and shuffles them, and time warping stretches and compresses random segments of the signal, resulting in random slowdowns or speeding up of the signal. Following this comes the core architecture, the training of a feature extractor utilizing a ResNet-16 V2 with 18 layers of 1D convolutions, designed to process random 10-second windows of raw tri-axial acceleration signals down-sampled to 30Hz. This deep CNN has more than ten (10) million parameters.

To finish, they added a final fully connected layer, which outputs a learned feature vector of size 1024.

Once this model was trained, they performed human activity recognition using transferred learning in order to fine-tune the model on eight (8) labeled datasets. They used a 7:1:2 split ratio for train/validation/test sets and early-stopping via patience to avoid overfitting. The fine-tuning was done with two approaches: the first was fine-tuning all the layers, the second consisted of freezing the feature extractor and only fine-tuning the final fully connected layer. Within the eight (8) datasets, they selected datasets of various sizes and types. From very large datasets, free-living datasets like Capture24 [6], small laboratory datasets like PAMAP2 [46] and mixed lab-free living datasets like Opportunity [49]. Then they compared the results of their approach with a random forest. This serves as a crucial benchmark due to its simplicity and is often overlooked when evaluating machine learning models.

The resulting model demonstrated exceptional robustness. When fine-tuned for downstream tasks, it consistently outperformed strong supervised baselines across the eight (8) benchmark datasets, showing a median relative improvement in the F1-score of 24.4%. Crucially, the study proved that self-supervised pre-training allows the model to generalize across external datasets, different sensor devices, and varied environmental cohorts, effectively bridging the gap between laboratory training and real-world deployment. This work serves as a critical precursor to zero-shot HAR using LLMs, as it demonstrates the feasibility of learning universal motion representations without explicit activity labels.

While the previous work focuses on the scale of data and self-supervision, other contemporary works focus on addressing architectural nuances required to capture complex

human movements. Addressing the trade-off between recognition accuracy and computational efficiency, the work Cross-Attention Enhanced Pyramid Multi-Scale Network introduces a symmetric Pyramid Multi-scale Convolutional Network (PMCN) with a Cross-Attention mechanism [41]. This architecture is designed to obtain a rich receptive field at finer levels, allowing the model to capture both micro-movements (e.g. typing) and macro-movements (e.g. walking). By integrating a triple attention mechanism, the model establishes interrelationships among sensor dimensions, temporal windows, and feature channels, effectively suppressing irrelevant noise while amplifying salient motion information. They evaluate their novel approach on four (4) benchmark laboratory datasets, including PAMAP2 [46] and Opportunity[49]. The results they obtained were 0.18% lower than the best performing state-of-the-art related work, this was achieved with less than half of the parameters. Compared to benchmark models, such as Support vector machine (SVM) and a Cross-Net, they also achieved superior F1-score while needing less GPU performance, measured in FLOPs, to achieve it. Their results demonstrate equivalent HAR F1-score performance to the related and state-of-the-art works while maintaining low computational overhead.

The work titled "Multi-STMT: Multi-Level Network for Human Activity Recognition Based on Wearable Sensors" proposes a novel deep learning framework designed to address the challenges of insufficient feature extraction and inadequate attention to critical signals in wearable sensors for HAR tasks [72]. The authors introduce Multi-SMT, a multi-level network that integrates a CNN module, Bidirectional gated recurrent unit (BiGRU) modules, and an attention mechanism. This architecture is specifically created to tackle multimodal time-series data from sensors (e.g. accelerometers) by leveraging a spatiotemporal attention mechanism and multiscale temporal embedding. By incorporating attention layers within the CNN and BiGRU modules, the model can adaptively focus on significant spatial patterns and

long-term temporal dependencies while suppressing noise. The effectiveness of Multi-STMT was validated through experiments on four major public datasets. The results demonstrate an accuracy improvement of 5.39% over the state-of-the-art method with a negligible increase in spatiotemporal complexity. These findings indicate that the multi-level approach significantly improves the ability of HAR systems to distinguish between subtle activity differences, making it a robust solution for real-world applications in healthcare.

To complement the large-scale adaptation of models to HAR tasks, recent literature has focused on the practicalities of 24-hour monitoring and the distinction between static and dynamic behaviors. A person's movement is typically a mixture of static (e.g. sitting) and dynamic (e.g. walking) states. FusionActNet [52], introduces a two-stage training methodology within a multi-structured architecture. The first stage pre-trains two dedicated residual networks to capture the static and dynamic behavior of an activity. The second stage uses a guidance module, constructed with depth-wise separable CNN blocks, to facilitate the final decision-making between similar activities. The proposed framework was rigorously evaluated using two benchmark datasets, UCI HAR and Motion-Sense. The results demonstrate significant improvements over existing methods, with FusionActNet achieving a state-of-the-art accuracy of 97.35% on the UCI HAR dataset and 95.35% on the Motion-Sense dataset. This underscores the model's stability and ability to extract meaningful features from the complex, high-frequency signals of wearable devices. The authors highlight the practical utility of this system in real-world applications, including robot-based assistance for the elderly and personalized healthcare monitoring.

Finally, AccNet24 [14] introduces a novel deep learning framework designed to automate the classification of daily activities on large datasets. The authors developed and evaluated

a multi-stage deep learning framework for classifying 24-hour activities from wrist-worn accelerometers. To begin, they segment the acceleration data into thirty (30) second windows and apply signal-to-image conversion on each segment. Using transfer learning, the authors then extract deep features from the signal images. The extracted features are used to classify the activities as one of the following four labels: sleeping, sedentary, light intensity or moderate to vigorous intensity. This classification is done using a bidirectional long short-term memory recurrent neural network, which captures temporal dependencies to accurately distinguish between the activities. The results of the study demonstrate that AccNet24 significantly outperforms traditional machine learning approaches that rely on hand-crafted features. To evaluate their approach, the authors use a dataset of 151 participants, split between 101/25/25 for train/validation/test. Continuously achieving a high accuracy greater than 95%, the framework surpassed other common classifiers, such as Random Forest and Support Vector Machines, by a margin of 16% to 30% on previously unseen data. With the new paradigm shift to 24-hour activity cycles [50], AccNet24 represents a shift toward more precise, 24-hour health monitoring by successfully integrating complex signal processing.

2.2 LARGE LANGUAGE MODELS

Given that LLMs are language-based models, early works focused on Natural Language Processing (NLP) classification tasks. They are widely used in a great variety of sub-fields of NLP and are well known as excellent few-shot learners with task-specific exemplars. The following section on LLMs focuses on the transition from standard prompting to instruction-tuned architectures and the emergence of Chain of Thought (CoT) reasoning, which provides the theoretical and empirical foundation for zero-shot classification when using LLMs. The

foundational breakthrough in enabling LLMs to perform complex tasks without task-specific training is detailed in the seminal work "Large Language Models are Zero-Shot Reasoners" by Kojima et al. [26]. This work challenges the then-dominant paradigm that required few-shot learning to elicit multi-step reasoning from models like GPT-3.

This work introduced a simple yet breakthrough prompting strategy known as Zero-shot-CoT. While previous works like "Finetuned Language Models Are Zero-Shot Learners" by Wei et al. demonstrated that models could solve complex problems if fine-tuned with instruction tuning [63], referred to as Few-shot-CoT, the work by Kojima et al. discovered that LLMs possess an inherent ability to reason that can be triggered by a single template: *"Let's think step by step."*

The mechanism of Zero-shot-CoT functions in a two stage process. First, the model is prompted with the question followed by the trigger phrase. This induces the model to generate a reasoning sequence of intermediate steps rather than jumping directly to an answer. Second, the generated reasoning and the original prompt are then sent to the model via a second prompt where the answer is demanded to the model (e.g. "Therefore, the answer is...") in order to isolate the final answer.

> **1st prompt: reasoning extraction** In this step we first modify the input question x into a prompt x' using a simple template "Q: [X]. A: [T]", where [X] is an input slot for x and [T] is an slot for hand-crafted trigger sentence t that would extract chain of though to answer the question x. For example, if we use "Let's think step by step" as a trigger sentence, the prompt x' would be "Q: [X]. A: Let's think

step by step.". Prompted text x' is then fed into a language model and generate subsequent sentence z.

2nd prompt: answer extraction In the second step, we use generated sentence z along with prompted sentence x' to extract the final answer from the language model. To be concrete, we simply concatenate three elements as with "[X'] [Z] [A]": [X'] for 1st prompt x', [Z] for sentence z generated at the first step, and [A] for a trigger sentence to extract answer. The prompt for this step is self-augmented, since the prompt contains the sentence z generated by the same language model. In experiment, we use slightly different answer trigger depending on the answer format. For example, we use "Therefore, among A through E, the answer is" for multi-choice QA, and "Therefore, the answer (arabic numerals) is" for math problem requiring numerical answer. Finally, the language model is fed the prompted text as input to generate sentences ^y and parse the final answer

The Zero-shot-CoT approach was evaluated on twelve (12) datasets from four categories of NLP tasks: arithmetic, common sense, symbolic and other logical reasoning. For the experiments, seventeen (17) models were used in total, ranging from small models with 0.3B parameters to very large models of 540B parameters. GPT-3 was used in a few different variations, as it was the most promising and state-of-the-art LLM at the time, while a few other models were used such as PaLM and OPT. Their main point of comparison for this new approach was zero-shot prompting without CoT.

Their results demonstrate significant performance improvements of the Zero-shot-CoT method. In arithmetic tasks, Zero-shot-CoT demonstrates gains of up to 61% on the MultiArith

dataset and up to 30.3% on the GSM8K dataset. However, in common sense reasoning, the authors did not notice a noticeable improvement which was expected since Few-shot-CoT also did not demonstrate an improvement [63]. Even though the answer is wrong on common sense reasoning tasks, the authors notice that the answers are logically correct or simply contain a minor mistake that is understandable by a human. The results also indicate a strong correlation between performance increase and model size. Smaller models showed a more subtle increase in performance when employing Zero-shot-CoT versus larger models that saw a significant gain in performance. The research also evaluated the impact of prompt selection for CoT. The prompt that showed the greatest gain in performance was *"Let's think step by step."* at 78.7% accuracy. However, adding a small distinction saw a reduction of 8.4% when using the prompt *"Let's think like a detective step by step."* indicating possible added confusion to the model when adding superfluous information that is not necessary.

The ability of the LLMs to adequately process the prompts described by the work of Kojima et al. is mostly a result of the work of the Google Research team of Wei et al. [63] in which they introduce instruction tuning to fine tune LLMs. This has the impact of significantly improving the zero-shot ability of LLMs on unseen tasks. In this work, the team introduced instruction-tuned model called FLAN (Finetuned Language Net), a 137B parameter pre-trtained LLM fine tuned with instruction tuning on a collection of more than sixty (60) NLP tasks via natural language instructions. One of the key factors was the use of instruction templates that varied for each task, forcing the model to learn the intent behind the instruction rather than just the statistical patterns of the dataset. Their results indicated that instruction tuning substantially improves the model's performance on unseen tasks. For instance, a model tuned on translation and common sense reasoning tasks could perform zero-shot sentiment classification with higher accuracy than the much larger GPT-3 at 175B parameters.

The authors evaluate their model FLAN against the zero-shot performance of LLMs that were not fine tuned on natural language inference (NLI), reading comprehension, closed book QA, translation, common sense, coreference resolution and struct-to-text. This gives a great variety of NLP tasks to compare against and provides a clear and wide ranging picture of the performance of their approach. They evaluate the models on unseen tasks by grouping datasets into task clusters. The demonstrated results are impressive, compared to GPT-3 175B, FLAN 137B saw an increase of up to 17.4% on zero-sho translation tasks, up to a 37.5% gain on NLI, an increase of up to 22.4% on reading comprehension and of up to 23% on sentiment tasks. Certain categories, such as common sense, saw no noticeable performance improvement. This suggests that certain NLP tasks are more limited by model architecture than exposure to information. Overall, this work established that zero-shot performance is not just about model size, but about exposure to diverse task descriptions during a secondary, fine tuning phase.

Building on the reasoning capabilities identified by the two previous works of Kojima et al. and Wei et al., subsequent research has focused on optimizing LLMs specifically for classification accuracy and domain-specific recognition. The approach proposed in the work "Text Classification via Large Language Models" by Sun et al. seeks to address the limitations of LLMs when faced with text classification [58]. In this work, the authors notice that LLMs tend to struggle compared with fine tuned models. They indicate that this is caused by the regular models lack of reasoning ability with linguistic nuances, such as irony and subtle sentiment cues, and caused by the limited amount of tokens permitted in context learning. In order to address these limitations, they propose CARP: Clue and Reasoning Prompting.

CARP adopts a two phase progressive reasoning strategy specifically developed to address the limitations in linguistic text classification. In the first phase, the model identifies superficial

cues. These are keywords, tones, semantic relations, references, etc. found within the text. The second phase involves using the clues to induce a diagnostic reasoning process that concludes with a final label. This approach led to a new state-of-the-art performance on four (4) out of five (5) benchmarks. The results obtained are 97.39% (+1.24) for SST-2, 96.40% (+0.72) for AGNews, 98.78% (+0.25) for R8 and 96.95% (+0.6) for R52. The work presented in this paper demonstrates that forcing a LLM to explicitly state the clues before performing classification significantly reduces errors.

Another interesting work building on the CoT principal in the field of LLM and NLP comes from Saeed et al. in their work "Large Language Models Are Zero-Shot Text Classifiers" [62]. In this work, the authors focus on the issues of expensive computational costs, time requirements and performance to unseen classes observed on text classification problems in NLP while using LLMs. Their work focuses on effectively using prompt strategies and comparing LLMs with other state-of-the-art classification methods, including traditional machine learning techniques. This work highlights that while LLMs may have higher latency and cost than a trained SVM or random forest, their generalization capability makes them superior for tasks where labels are scarce or defined in real time. They emphasize the role of prompt engineering as the new feature engineering where the success of a classification task depends on how well the semantic boundaries of the classes are defined in the prompt. They also state the advantages of LLMs as especially advantageous for NLP classification asks where the operator may not have extensive knowledge of NLP classification.

A critical challenge in zero-shot classification tasks with LLM is domain mismatch. This is when the model struggles on performing tasks pertaining to a highly specific domain (e.g. financial earnings) since it was trained on a generalized dataset. The work of Li et al. in

"Prompting Large Language Models for Zero-Shot Domain Adaptation in Speech Recognition" [31] explores a novel approach to addressing the "domain mismatch" problem in Automatic Speech Recognition (ASR). This work proposes using LLaMA 7B from Meta [60] to perform zero-shot domain adaptation, where the model is adapted to a new domain using only a descriptive text prompt rather than intensive retraining. The authors introduce two primary methods for this adaptation. The first is shallow fusion, this approach consists of the model providing external linguistic scores to guide the ASR decoding process during a second pass. The second approach is deep LLM-fusion. This method integrates the model into the speech recognition architecture which effectively adapts the model to the domain. By providing a contextual prompt (e.g. "The following text is the transcription of company earnings calls"), the model can leverage its vast internal knowledge to better predict domain specific terminology and context without task-specific fine tuning.

The demonstrated experimental results for the TedLium-2 [40] and SPGISpeech [51] datasets demonstrate that this prompting strategy significantly reduces word error rates across different domains. The deep LLM-fusion approach was particularly effective for improving the recall of entities and out-of-vocabulary words that the ASR system might not have encountered during its initial training. Ultimately, the study highlights that the in-context learning capabilities of LLMs can be a cost-effective and flexible alternative to traditional domain adaptation methods, making NLP classification tasks more versatile across diverse specialized fields.

2.3 LLMS AND HAR

The integration of LLMs into the domain of HAR represents a paradigm shift from traditional supervised learning architectures, such as CNNs and LSTM networks, to generative and reasoning-based frameworks. This section evaluates the transition from data-intensive discriminative models to zero-shot, prompt-based recognition, focusing on the architectural innovations and cross-modal challenges associated with interpreting inertial sensor data through the lens of natural language.

A foundational contribution to this field is the work titled "HARGPT: Are LLMs Zero-Shot Human Activity Recognizers?" [24]. This study serves as an important benchmark as it explores whether the internal knowledge of pre-trained LLMs can be leveraged to interpret raw IMU sensor data for HAR tasks without the need for task specific training or domain fine tuning. The authors propose HARGPT, a framework that leverages the pre-trained world kn.owledge of LLMs, specifically GPT-4, to perform zero-shot classification on HAR tasks. The core hypothesis is that LLMs, through their vast exposure to textual descriptions of physical movements and physics-based reasoning, can understand and categorize activity patterns directly from raw sensor data. Crucially, HARGPT makes use of CoT prompting as seen in the previous section.

The framework proposed by the HARGPT team involves two (2) prompting strategies to elicit the LLMs reasoning capabilities. The first strategy is role-play prompting. This is done by instructing the LLM to assume the role of a human activity recognition expert through the instruction "You are an expert of IMU-based human activity analysis". This is crucial as it indicates to the LLM to adopt a domain specific "mindset" when treating the following prompt,

which in the case is the domain is human activity recognition. As seen in the previous section, this has been shown to reduce errors as the model focuses on domain-specific details. The second strategy is chain-of-thought prompting. By adding the instruction "think step-by-step" in the prompt, the authors trigger sequential reasoning in the model which breaks down the problem into smaller, logical steps before providing a final answer, essentially "showing its work".

To perform their experiments, the authors had to tackle the complexity of communicating raw IMU data to the LLM. To do so, they prepared a structured prompt. The first step was an instruction to the model telling it was an "expert" in human activity recognition, triggering the role-play prompting mechanism. This was followed by the main portion of the prompt, the question. The question begins by stating that it was IMU data collected from a wrist-worn device at a specific frequency (sampling rate). The acceleration and gyroscope data was then inserted in the question in the following format:

> Accelerations: x-axis: ..., y-axis: ..., z-axis: ...
>
> Gyroscopes: x-axis: ..., y-axis: ..., z-axis: ...

The second portion of the question included the finite list of possible activities the data could represent, followed by asking the model to "tell us" what action the person was doing based on the data. The final part of this prompt asks the model to make a step by step analysis to trigger the CoT strategy within the model.

To evaluate this approach, the authors selected GPT-4 [39] from OpenAI as the LLM to be tested since it was the most promising state-of-the-art model at the time. Furthermore,

they selected two (2) public HAR datasets for experimentation: Capture24[6] and HHAR (Heterogeneity Activity Recognition)[2]. Capture24 contains only acceleration data but consists of free-living data captured from participants over a 24-hour period. Four (4) activities are used from this dataset: Bicycling, Sit-stand, Sleep and Walking. The HHAR dataset contains both accelerometer and gyroscope data. From this dataset the authors selected two similar activities to evaluate: Going downstairs and going upstairs. The data from both datasets was down sampled to a frequency of 10Hz to limit the amount of tokens sent to the LLMs. In order to better understand the HARGPT framework, the authors compared their approach with traditional machine algorithms, random forest and SVM, as well as standard deep learning methods, a deep CNN and a LIMU-LSTM. Finally, they also performed the same experiments with GPT-4 but without triggering CoT. This was done by replacing the final phrase of the prompt asking the model to make a step by step analysis by asking it to "please give your answer directly without analysis."

The results demonstrate strong performance of GPT-4 on zero-shot HAR tasks. For Capture24, GPT-4 with CoT achieved a F1-score of 79.5% while both random forest and SVM showed F1-scores similar to each other, at 55.5% and 54.5% respectively. Additionally, the deep learning approaches of deep CNN and LIMU-LSTM achieved a similar F1-score as well, at 58.8% and 58.5% respectively. When looking at the results for the HHAR dataset, GPT-4 with CoT performs well with an F1-score of 79.0%. However, the results are more diverse when looking at the other approaches. For traditional algorithms, random forest achieved a score of only 32.5% while SVM maintained a similar score of 57.0%. On the deep learning side of things, deep CNN got a score of 38.5% while LIMU-LSTM achieved an impressive score of 70.0%. Finally, the results also demonstrate the importance of CoT observed in the previous sections. GPT-4 without CoT achieved an F1-score of 56.5%, compared to 79.5%

with it for the HHAR dataset. For Capture24, the difference is even more significant with a F1-score of 46.5% without CoT versus 79.0% with it.

The results achieved in this work demonstrate a strong potential of LLMs performing well on zero-shot HAR tasks. They also validate the importance of using CoT when using LLMs across different domains as it significantly improves the performance of the models on a variety of tasks. Specifically, it demonstrates the models ability for logical reasoning based on their world-knowledge and suggests initial capability in handling raw numerical sensor data. Although, the HARGPT team also mention that on some occasions the model did return perfunctory answers, meaning the answer provided was more of a constructive opinion on how to handle the task and did not provide an answer. Some primarily experiments done by the team on smaller models, mentioned in the discussion section of the paper, suggests a strong correlation between the model size and performance as smaller models demonstrated a lower robustness and higher tendency to provide perfunctory answers.

Published in the same time frame as HARGPT, the work titled "Unsupervised Human Activity Recognition through Two-stage Prompting with ChatGPT" by Xia et al. from the university of Osaka proposes a two-stage prompt engineering approach that leverages the pre-trained embedded general knowledge of LLMs to interpret sequences of activities that interact with objects, captured via wearable sensors, without requiring any prior training on labeled or domain specific datasets [70]. The authors noted that ChatGPT, the web interface version of GPT-3, has a limited generalization ability when dealing with a series of activities that involve overlapping objects if presented as a simple list of objects (words) due to similar weights being assigned to the different objects (words) in the list. This approach proposed in this work addresses this limitation.

The core of the methodology is a two-stage prompt engineering approach. The first stage is called knowledge generation and is structured in two prompts. In this stage, ChatGPT is prompted to generate detailed descriptions of various human activities based on the objects typically associated with them. The first prompt is structured to identify which activities are difficult to distinguish from the objects used to perform the activity. This is achievable due to the implicit knowledge embedded within the LLM. The second prompt is designed to generate descriptions of the activities identified in the previous prompt with the knowledge obtained from the first prompt. The second stage is called answer generation, it uses the knowledge gained in the previous stage and applies the new descriptions to the activities to perform HAR tasks. This enables the model to better focus on the important objects and the differentiating elements to achieve better performance. At the end of this stage, the authors also ask the model to explain its answer which is a similar mechanism to CoT mentioned in the works of the previous sections.

To evaluate their approach, the authors use three (3) public HAR datasets: Opportunity[49], 50Salads[56] and CMU-MMAC[29]. They also compare the proposed strategy with zero-shot and few-shot strategies. The results demonstrate significant performance gains brought with the proposed strategy. The proposed solution achieved an F1-score of 91.15% compared to 53.61% with zero-shot for the Opportunity dataset, 100% versus 64.47% for 50Salads and 100% vs 76.32% for CMU-MMAC. These results demonstrate the importance of having clearly defined labels when using LLM for HAR tasks as a clear understanding of the underlying activity can greatly assist the model in attributing the correct classification.

In this stage, ChatGPT is prompted to generate detailed descriptions of various human activities based on the objects typically associated with them. Crucially, the model is guided to

identify discriminative objects that help distinguish between activities which generate similar sensor data. For example, distinguish *making a sandwich* from *cleaning the kitchen* by focusing on the specific use of bread or a sponge. This allows the system to build a semantic bridge between raw sensor data and higher level human actions.

The paper "Large Language Models are Zero-Shot Recognizers for Activities of Daily Living" builds on the previous works with the introduction ADL-LLM, a framework that transforms raw, environmental sensor data (e.g. pressure sensors) into natural language descriptions [11]. Instead of feeding a LLM raw numerical data, the framework generates textual event logs (e.g. "The subject entered the kitchen and turned on the stove."). These descriptions are then passed to the model through a carefully designed prompting strategy. The LLM acts as a zero-shot recognizer of human activities, meaning it can accurately identify the activity (e.g. cooking) without ever having been explicitly trained on the domain specific dataset.

The authors performed the evaluation of the framework on two public datasets which demonstrates remarkable performance from ADL-LLM, often matching or exceeding the accuracy of supervised deep learning approaches. The work also explores a few-shot variation, where providing the LLM with just a few labeled examples of activities demonstrates a further gain in performance. This flexibility allows the framework to outperform standard methods, particularly in data scarce scenarios, where only a handful of labeled activities are available for a new individual.

Ultimately, this work further highlights the potential for LLMs to serve as scalable, out-of-the-box solutions for HAR tasks. By bypassing the need for labor intensive data

labeling and environment specific fine tuning, the ADL-LLM approach offers a more practical method for deploying unobtrusive monitoring systems in the homes of older adults or patients with disabilities requiring care. The authors suggest that this common sense reasoning capacity makes LLMs more adaptable to the varied and relatively unpredictable ways different individuals perform activities.

A significant issue in HAR tasks is the variations in user behavior, different sensor types, and new environments, known as distribution shift. The paper "LLM4HAR: Generalizable On-device Human Activity Recognition with Pretrained LLMs" introduces a new LLM based model to mitigate this issue by introducing LLM4HAR [20]. LLM4HAR begins by using GPT-2 as the backbone LLM model for their approach [44]. The core of the model consists of three specialized modules designed to bridge the gap between continuous sensor data and the natural language functioning of LLMs. First, the sensor data adaptation module aligns IMU data with LLMs using sensor embedding techniques. Second, the sensor knowledge learning module injects domain specific sensor information into the LLM, allowing the model to interpret sensor data through the lens of its pretrained semantic understanding. This enables the model to recognize activities even in cross domain scenarios where it has not seen specific user-sensor combinations before. The third module is the efficiency enhancement module which uses a fine tuning strategy to reduce the model size which enables the solution to be used on a mobile device.

To evaluate LLM4HAR, the authors begin by evaluating their approach on four public HAR datasets: HHAR[2], Shoaib[53], Motion[38] and UCI[47]. The evaluation metrics they use are average accuracy and F1-score. Furthermore, they compare their work against represented learning models (Gated Recurrent Unit, Transformer and TSMixer), transfer learning models

(SDMix, UniHAR and CrossHAR) and against the HARGPT[24] approach. The results demonstrate that LLM4HAR achieves the best performance across all datasets, with an average F1 score of 71.42% and an average accuracy of 79.86%. Over all, the model proposed by this work outperforms all other methods by 7.02% in average accuracy and 13.82% in F1-score.

Beyond academic metrics, the model was deployed in a real world use case at a company called JD Logistics, one of the largest E-commerce logistics companies in China. The model was deployed on a smartphone device for courier activity recognition. The application in this real world use case had the objective of identifying the presence of elevators based on movement and on detecting the workload of the courier based on time spent walking, number of stairs climbed, etc. In the use case of identifying elevators, the system achieves an accuracy of 90.8% while in the case of workload detection the accuracy achieved is 90.9%. These results demonstrate that the generalized reasoning of LLMs on HAR tasks can be successfully transferred to the physical world, providing a robust solution for scalable and adaptable HAR.

2.4 DISCUSSION

To conclude, the fields of research presented in the above sections of the literature review comprise the intersection of domains that have been subject of research for very different lapses of time. HAR has been a field of interest for decades and the subject of much research with strong results when using deep learning methods. With the limitations of deep learning approaches concerning the necessity of large labeled datasets and requiring re-training when faced with new environments, activities or individuals, the latest approaches focus on complex, large architectures with self-supervised techniques on unlabeled datasets to mitigate these

limitations [71]. Other works attempted to mitigate the issues of inter-subject and intra-subject variability by capturing subtle movements in the data [41] or extracting the critical points in the data [72]. Yet, the limitation of needing to be re-trained when faced with new environments or sensors remained, and the complexity of these approaches render their implementation difficult.

In contrast, LLMs are a very recent technology generating a lot of interest and many research projects with wide ranging applicability. Much effort has been put into determining if they are effective few-shot [63] or zero-shot [26] solutions in a variety of tasks, although initial works have focused on NLP tasks. This has revealed the importance of the architecture of the prompt sent to the model, specifically the importance of the CoT technique [64]. These works were crucial in permitting the most recent works merging LLMs and HAR [24] [11], which demonstrate promising initial capability of these models being zero-shot human activity recognizers. This would greatly facilitate the deployment of HAR based solutions in smart homes that do not require a complex solution, re-training or fine-tuning for different sensors, environments or individuals.

CHAPITRE III

EXPERIMENTAL SETUP

This chapter details the physical and technical environment in which the research is conducted. It begins by describing the LIARA smart home laboratory and the specific sensing infrastructure used for data collection. Subsequently, the two datasets, the custom laboratory dataset and the Capture24 public dataset, are presented and compared. Finally, this chapter outlines the computational models and hardware specifications utilized to execute the experiments, providing the necessary context for the data processing techniques discussed in the following chapters.

3.1 LIARA SMART HOME ENVIRONMENT

The LIARA laboratory, as seen in Figure 3.1, is a replication of an apartment style smart home found within the UQAC and is one of the laboratories part of the INTER strategic cluster. The LIARA laboratory represents a one (1) bedroom apartment with a kitchen, bathroom, living room and dining area. It includes all the furniture and equipment that you would expect to find in a standard north American apartment. The laboratory contains a sensor suite which includes: Ultra-wideband (UWB) radar, Red-Green-Blue-Depth (RGBD) cameras, an IMU sensor worn

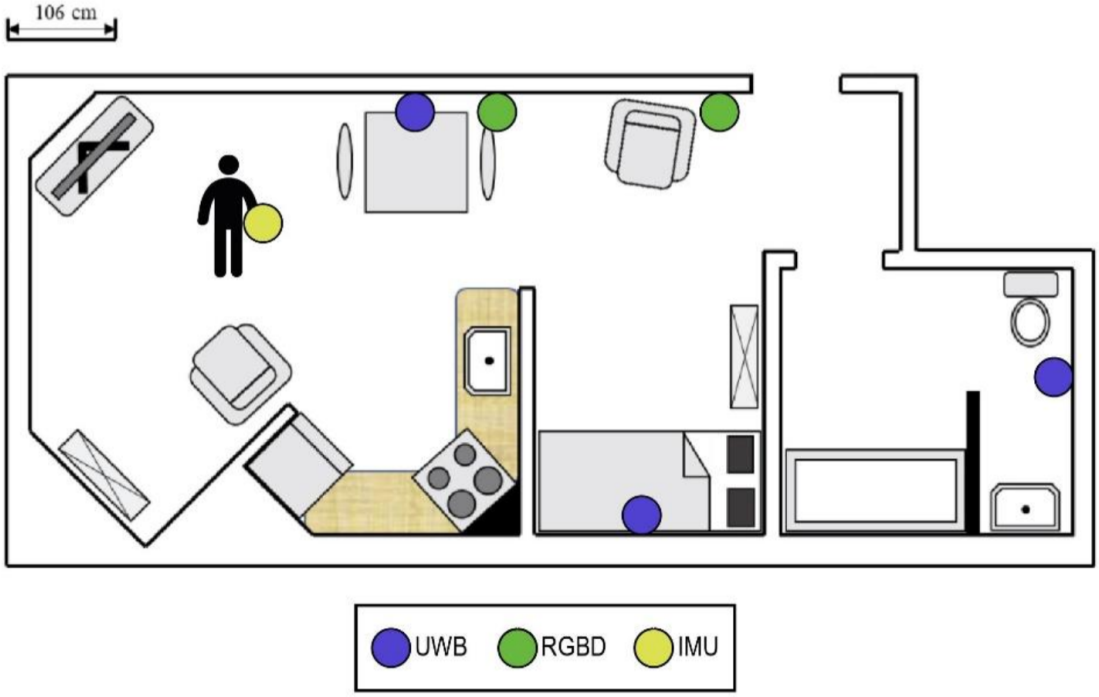

Figure 3.1 : LIARA laboratory layout

by participants and ambient sensors. Sensors can be used individually or in a grouping to collect data while participants go about simulating a variety of household daily living activities. Table 3.1 lists the family of sensors found in the laboratory and their particularities.

Sensors	Frequency	Information
Ambient sensors	Event based	-
Ultra-Wideband Sensors	50 Hz	-
IMU Bracelets	50 Hz	Bluetooth
RGBD Sensors	60 FPS	720p

Table 3.1 : Sensors found in LIARA Smarhome Laboratory

3.2 DATASETS

This section introduces and describes the datasets that we selected to perform the experiments of the research project. In machine learning, the data chosen to train and evaluate models is of critical importance as choosing the wrong dataset will lead to erroneous results even if the rest of the methodology is correct. For our use case, we selected two datasets: one of our laboratory's internal datasets and one public dataset. The internal LIARA dataset we selected was specifically created to replicate real world living conditions with a smart home from multiple participants. This means it purposely includes inter and intra-variability within the data. We also know for a fact that this dataset was not seen by LLMs in training. The public dataset we selected is the Capture24 [6] dataset. This dataset was selected for two primary reasons. One, it is a relatively large free-living dataset which was manually annotated from video worn by the participants. This makes it a reliable dataset in which participants did not follow a defined activity path. Second, it is a dataset frequently used in HAR works, including the work HARGPT [24] mentioned in the literature review.

3.2.1 LIARA

The LIARA dataset we selected for this experiment is one of our internal datasets that was collected within our laboratory in 2022 and is stored privately on our servers, thus unavailable to the general public. As mentioned previously, this dataset was specifically selected for our experiments since it replicates a real world scenario with inter and intra-variability. This means that for the same activity (label), we observe data variations on how that activity is performed by different individuals and variations by one individual performing the same

activity at different points in time. In order to achieve this, a participant was given the name of an activity to perform but it was left up to them on how they performed it. For example, when simulating the activity of taking a shower, a participant could wash their hair or just their body. They could start by adjusting water temperature or skip this step entirely. This created variations in the data between participants. Other activities, such as brushing their teeth, would be performed at different times and different areas; kitchen vs bathroom. This meant that even for the same individual, the data could have variations for the same activity label.

The dataset was collected from twenty six (26) participants within the smart home laboratory. The structure of the dataset consists of folders for each participant, each with a folder for each activity. Within the activity folder, each sensor is represented by a dedicated JSON file containing its recorded activity data. The dataset consists of over 1600 files at a total size of 55GB. The dataset consists of fourteen (14) distinct daily activities per participant. Activities are as follows:

- Brushing your teeth
- Taking a shower
- Going to the Bathroom
- Sleeping
- Getting dressed/undressed
- Doing housework
- Cleaning dishes

- Putting away dishes
- Resting
- Using a computer/mobile phone
- Putting away clothes
- Eating
- Drinking
- Reading

Some activities have more than one simulation acquisition and some are performed in multiple locations. This is indicated in the name of the JSON file (e.g. 3ca7e943_IMU_BrushingTeeth_2_Bathroom.j
Importantly, for each acquisition, the recording of the data begins ten (10) seconds before the participant begins simulating the activity and the recording continues for five (5) seconds after the participant has stopped the simulation. Figures 3.2 and 3.3 represent an example of acceleration and gyroscope data for a participant who is drinking, captured by the LIARA team.

In the context of this research project, we only used data from the custom IMU bracelet as seen in Figure 3.4. This bracelet was created by the LIARA team in a previous project as an open-source bracelet for the medical and healthcare context [7]. This bracelet captures data at 50Hz, as seen in Table 3.1. We strictly use the IMU data in this research project for two primary reasons. The first is that many related works mentioned in the literature review use

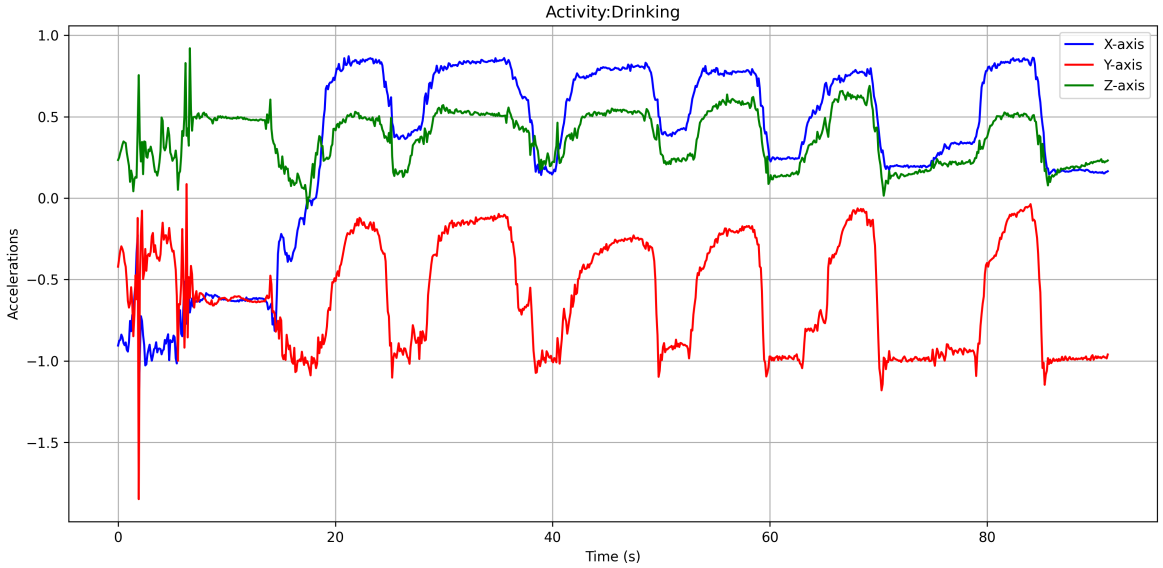

Figure 3.2 : Participant f87e2ddf Drinking Full Acceleration Data

Figure 3.3 : Participant f87e2ddf Drinking Full Gyroscope Data

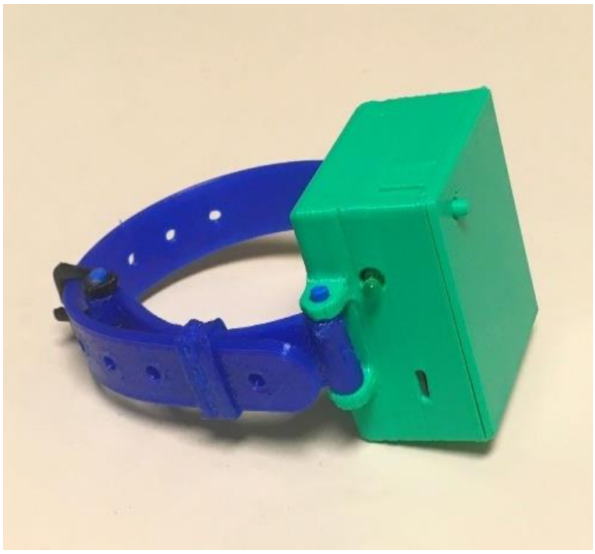

Figure 3.4 : IMU bracelet

IMU data to perform HAR tasks with LLMs [24] [11] [20]. The second is that the public dataset with which we compare our dataset is a IMU only dataset.

3.2.2 CAPTURE24

The Capture24 dataset is a large scale, free living, publicly available repository for HAR developed by researchers at the University of Oxford [6]. It is frequently utilized in the literature as a benchmark for evaluating HAR algorithms due to its focus on real world data, which captures the natural, unscripted variability of human behavior in real world settings.

The dataset consists of three (3) axis acceleration data recorded at a sampling frequency of 50 Hz. The primary hardware used for data collection was the Axivity AX3, a medical-grade wrist-worn accelerometer. Data was collected from over 200 participants over continuous

24-hour periods, providing a massive longitudinal record of physical activity across various demographics.

A significant advantage of the Capture24 dataset is its objective ground truth methodology. Rather than relying on participant self-reporting, the study employed a wearable camera worn by participants. This camera captured images every 20 to 30 seconds and were subsequently manually annotated by researchers to provide precise activity labels. This labeling allows for training and validation of a variety of classification models.

In the context of this research project, Capture24 serves as an important initial point of comparison for our internal laboratory dataset. Similar to our internal dataset, Capture24 captures the complexities of real world living since the participants perform the activities in their natural setting. Furthermore, this dataset has been used by other works for HAR tasks which allows us to better compare our results with previous research. By applying the same data processing techniques for both datasets, we can evaluate the proposed approaches on a public and private dataset to get a better sense of performance and generalization. Using both of these datasets ensures that the results are not limited to a specific smart home or outdoor environment but are robust when applied to the complexities of daily living in various contexts.

3.3 HARDWARE ENVIRONMENT

One of the main objectives of this research project is to evaluate the possibility of running models locally for HAR tasks. The LIARA laboratory possesses a high performance machine which enables us to run large models, such as Llama 3.3 70B, locally. This machine is composed of the following specifications:

- Ubuntu 20.04.6 LTS
- 256GB of DDR4 ECC memory
- Two (2) AMD EPYC 7513 processor
- Eight (8) NVIDIA RTX A5000 24GB graphic cards

Access to this machine is achieved entirely through Secure Shell Protocol (SSH) via the university network as no user interface is provided and no physical access to the machine is possible. The machine has limited internet access, primarily allowing pip to install Python packages or other Linux commands (e.g. git).

Additionally, a mobile workstation is used to perform experiments with online models (e.g. GPT-4o) and perform other necessary tasks such as developing proofs of concepts and code validation before running experiments on the LIARA lab machine. This machine consisted of:

- Ubuntu 22.04.5 LTS
- 32GB of DDR4 non-ECC memory
- Intel Core Ultra 9 185H
- Nvidia RTX 4080 12GB graphic card

3.4 LLMS USED

To satisfy the requirements for local execution within a smart habitat and to provide a comprehensive evaluation of LLM capabilities, a dual-deployment strategy is adopted. The

first approach leverages proprietary, state-of-the-art models provided by OpenAI via API integration. In parallel, the second strategy utilizes locally hosted Llama models. At the time of the study, the Llama architecture was one of the models with the best performance on public benchmarks and represented the most prevalent open-source model within the literature.

3.4.1 OPENAI GPT MODELS

The experimental framework for the online model portion of this research primarily utilizes models from the OpenAI Generative Pre-trained Transformers (GPT) series, specifically leveraging the architectural advancements of the GPT-4 [39] and GPT-5 [54] iterations. These models are used for experimentation via OpenAI API calls from a Python program.

Throughout the first half of the developmental and evaluative phases, GPT-4o [39] serves as the primary model from OpenAI. This selection is motivated by two primary factors: its status as the most advanced commercially available model at the project's inception and its widespread adoption within the contemporary works identified in the literature review. By using GP-4o [39] for much of the experimentation, a consistent baseline is maintained, allowing direct comparison with existing benchmarks and ensuring the reproducibility of the initial findings.

To ensure the research remains aligned with the rapidly evolving state-of-the-art, the final experimental trials incorporate GPT-5.2 [54]. Released toward the conclusion of the project timeline, GPT-5.2 [54] represents a significant generational leap in reasoning capabilities and contextual understanding. Given the observable performance improvements over the GPT-4o [39] baseline, it is integrated into the final evaluation phase to provide a comparative analysis

```
cmake -B build -DGGML_CUDA=ON
```

Figure 3.5 : Command to build Llama.cpp

against the most sophisticated frontier models available. This dual-model approach allows the results to reflect both the established standards of the current literature and the projected capabilities of the next generation of artificial intelligence.

3.4.2 LLAMA MODELS

Complementing the OpenAI online approach, several models from the Meta Llama 3 [17] family are deployed on a local laboratory machine to assess the viability of on-device, private inference. These models are run using Llama.cpp as a server [16]. When compiling Llama.cpp, we built it for use with CUDA to leverage the performance of the Nvidia GPUs found in our laboratories system, as seen in Figure 3.5. For running on the machine, we set the CUDA max batch size to 512 and enable unified memory via environment variables. The exact variables can be seen in Figure 3.6. For the command that runs the server, we set 81 layers to be offloaded to the Nvidia GPU, set it as background task and use the *nohup* command so it would stay running until we killed it. The exact command we use is displayed in Figure 3.7. The Python program used for local inference is the same used for the OpenAI models, albeit with a different API call configuration. The Python program runs on the same machine as the model for the local inference approach.

```
GGML_CUDA_PEER_MAX_BATCH_SIZE=512
GGML_CUDA_ENABLE_UNIFIED_MEMORY=1
```

Figure 3.6 : Environment variables used for Llama.cpp

```
nohup ./llama-server -m /data/models/Meta-Llama33-70B_Q8/Llama-3.3-70B-Instruct-Q8_0-00001-of-00002.gguf -ngl 81 --port 8080 &
```

Figure 3.7 : Command used for running the Llama.cpp server

Llama 3.1 8B [17] and Llama 3.2 3B [17] variants are specifically selected for deployment on mobile workstations to investigate the performance thresholds of lightweight models in highly portable and resource-constrained environments. To maintain high-fidelity output while adhering to the memory limitations of these workstations, both models are implemented using 8-bit quantization (Q8). This enables a direct evaluation of the trade-offs between model size and functional utility in edge-computing scenarios.

For a more robust comparison against proprietary cloud solutions, a Llama 3.3 70B [17] model is deployed on a centralized local server, also using a 8-bit quantization version (Q8). This configuration represents the maximum model scale feasible for self-hosting within the available infrastructure and provides an essential open-source counterpoint to OpenAI models. By spanning this range of model sizes, the experimental setup provides a comprehensive overview of how localized LLM integration performs across varying tiers of hardware availability.

CHAPITRE IV

REPLICATION OF EARLIER WORKS

4.1 OBJECTIVE

The primary goal of the first set of experiments is to reproduce the results observed in earlier works using our previously unseen laboratory's IMU dataset. As seen in the literature review, earlier works demonstrated that LLMs are effective zero-shot recognizers for sensor-based HAR tasks [24] [30]. The reproduction of these results serves two objectives: 1. Validate the approach used by the projects works in our context and with our private dataset and 2. Obtain a solid foundation for the next series of experiments within this master's projects and for future works.

To achieve this goal, we begin by evaluating the public dataset of Capture24 [6] that was used in the work of the work of HARGPT. As seen in the literature review, the paper HARGPT demonstrated promising zero-shot performance on HAR tasks with LLMs using this dataset [24]. Since this is a public dataset, there is a risk of it having been part of the training data of the LLM used in this work. To validate the capability of LLMs as zero-shot human activity recognizers, we apply the same methodology to our private laboratory's dataset. This will

provide us with a strong point of comparison and will further inform us on the capability of LLMs for HAR tasks.

4.2 PREPARING THE DATA

4.2.1 CAPTURE24

The objective of our first experiment is to benchmark the Capture24 dataset. The results we obtain from this dataset will enable us to compare more accurately the results we obtain on our private LIARA dataset since we will apply the same process and techniques to evaluate them. Unfortunately, we do not have all the details on how the dataset was prepared or how the experiments were conducted in the case of HARGPT so we will attempt to replicate the results to the best of our ability.

Since the Capture24 dataset contains thousands of hours of data and that feeding such data directly to the LLM would cause a memory error on the LIARA machine and incur significant costs for an online platform, it is necessary to start by obtaining workable windows of data. Once this is achieved, we will be able to feed this data in a structured prompt to the models for experimentation. The Capture24 dataset contains only three (3) axis of accelerometer data and no gyroscope data. This data was prepared in a csv format for each participant with the timestamp, label and x, y, z raw IMU data.

Inspiring ourselves from the paper HARGPT [24], we opted to select the same activities for experimentation: Bicycling, Sit-Stand, Sleep and Walking. We also opted to reduce the

frequency to 10Hz as they had done, thus maintaining a better comparison. Finally, sample random windows of data based on the desired activity labels.

4.2.2 LIARA

Similar to the Capture24 dataset, the first step of our LIARA experiments is to prepare the dataset. To begin, each activity is encoded in base64 for optimal storing and had to be decoded. Once decoded, each activity is between 60 seconds and 135 seconds with the sampling rate at 50Hz. This means that each activity has between 3 000 and 6 750 data points for each axis. With three (3) axis for gyroscope and three (3) for the accelerometer, this would result in a minimum of 18 000 data points per activity, this is a far too great amount of tokens to be sent to an LLM, even with our hardware. Out of curiosity, we tried and this leads to the a very slow inference time and OpenAI out right refusing our calls as being too large.

We began by down-sampling the frequency to 10Hz using the decimate function from the *scipy* library, which is the same approach taken by HARGPT [24] mentioned in the literature review. Once completed, we still have large windows up to 135 seconds for 1 350 IMU data points per axis. To further reduce this we implemented a function to randomly select a fifteen (15) second window for each activity. Since the recording started ten (10) seconds before the participant started performing the activity and kept going for 5 seconds after activity had completed, we removed the first and last ten (10) seconds from each activity before selecting the random window. This processed dataset is then saved separately in order to be re-used and for archival purposes.

Finally, since many of the reference datasets are benchmarked using only three (3) to five (5) activities, we chose to select four (4) activities from our dataset for our baseline. Considering that our dataset contains fourteen (14) distinct activities and some of them can be ambiguous (e.g. *DoingHousework*), we chose four (4) activities that are sufficiently distinct from another in terms of the data they generated while also challenging the model slightly with less common activities. For these experiments we chose the following activities: Eating, Reading a book, Sleeping, Taking a Shower.

Figures 4.1 through 4.8 provide visual examples of the IMU data for each activity. Here we can visualize the three (3) axis of accelerometer and three (3) axis of gyroscope data at 10Hz over fifteen (15) seconds randomly selected for a participant. In some instances, it is possible for us to directly understand an activity from the visual data. For example, Figure 4.5 shows the accelerometer data for a sleeping participant, which is showing no movement. Figure 4.4 shows the gyroscope data for a participant reading a book, we can clearly see the initial data shows the participant settling in a chair followed by the lack of significant movement while reading. We can also see when participant turns a page at roughly the ten (10) second mark from the spike in movement.

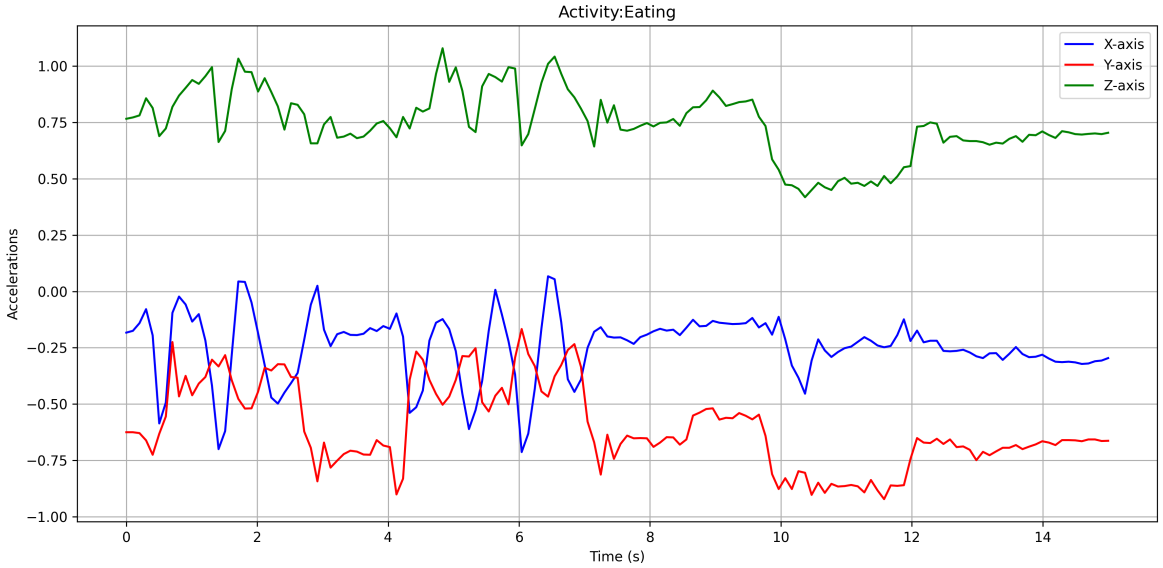

Figure 4.1 : Participant 1e1a8f23 Eating Acc

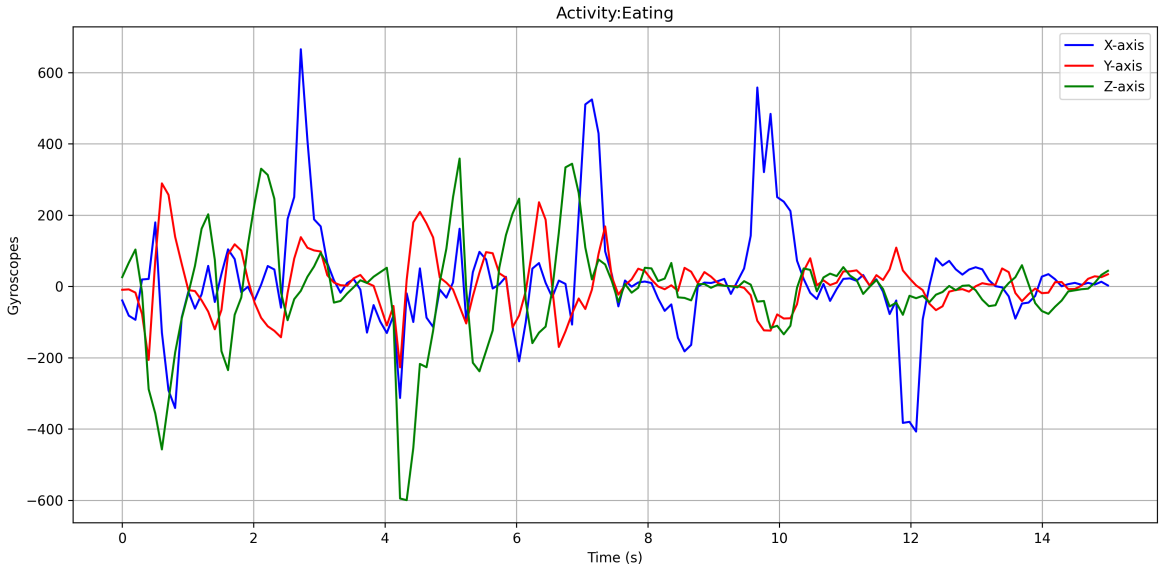

Figure 4.2 : Participant 1e1a8f23 Eating Gyro

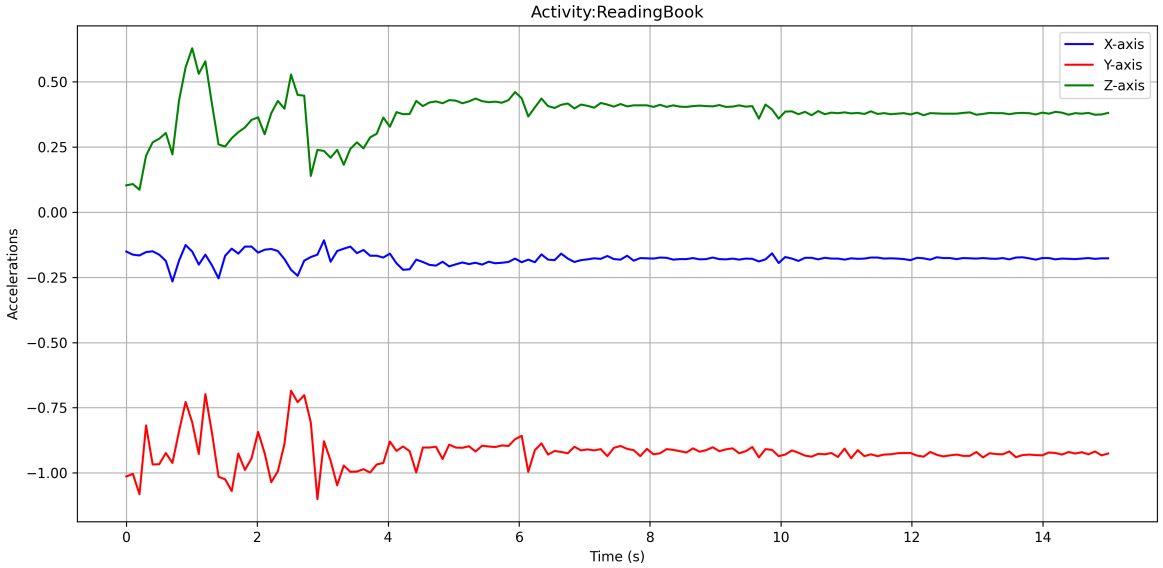

Figure 4.3 : Participant 4e36eb16 Reading Acc

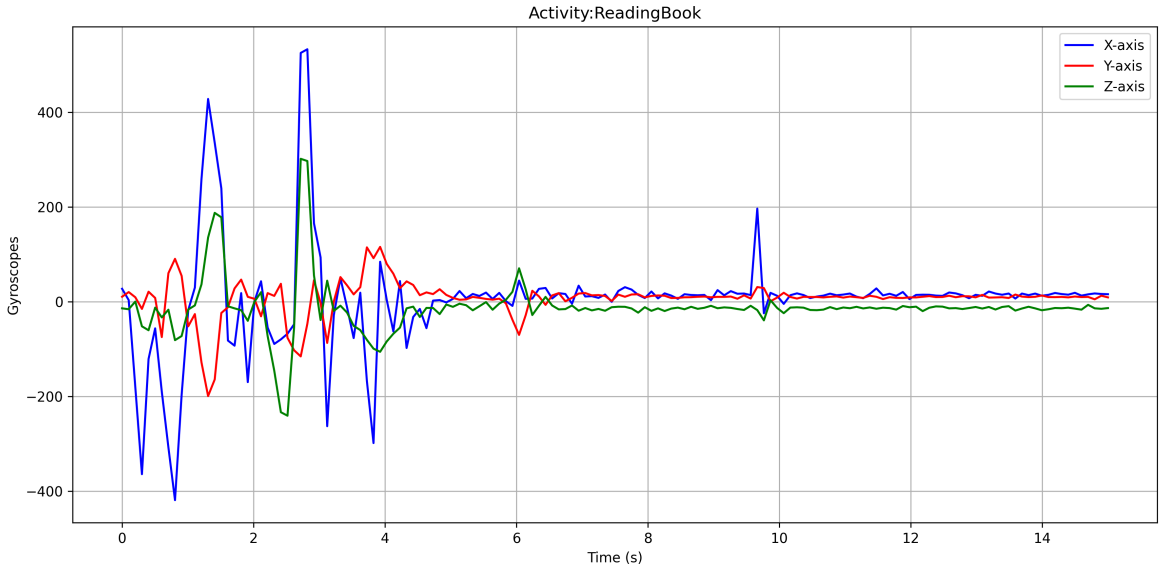

Figure 4.4 : Participant 4e36eb16 Reading Gyro

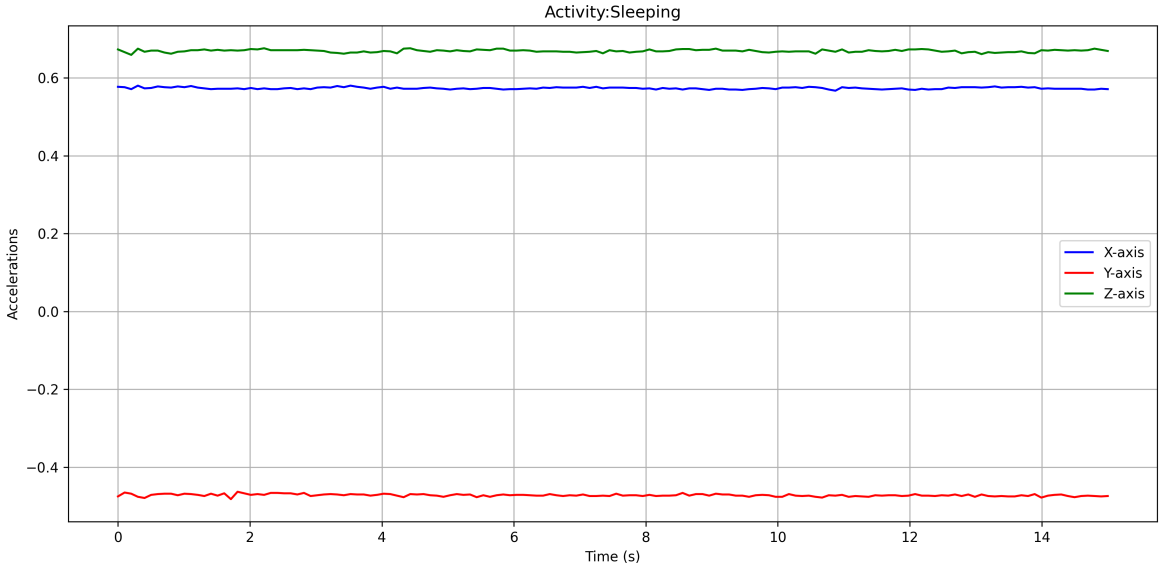

Figure 4.5 : Participant 21bf37e6 Sleeping Acc

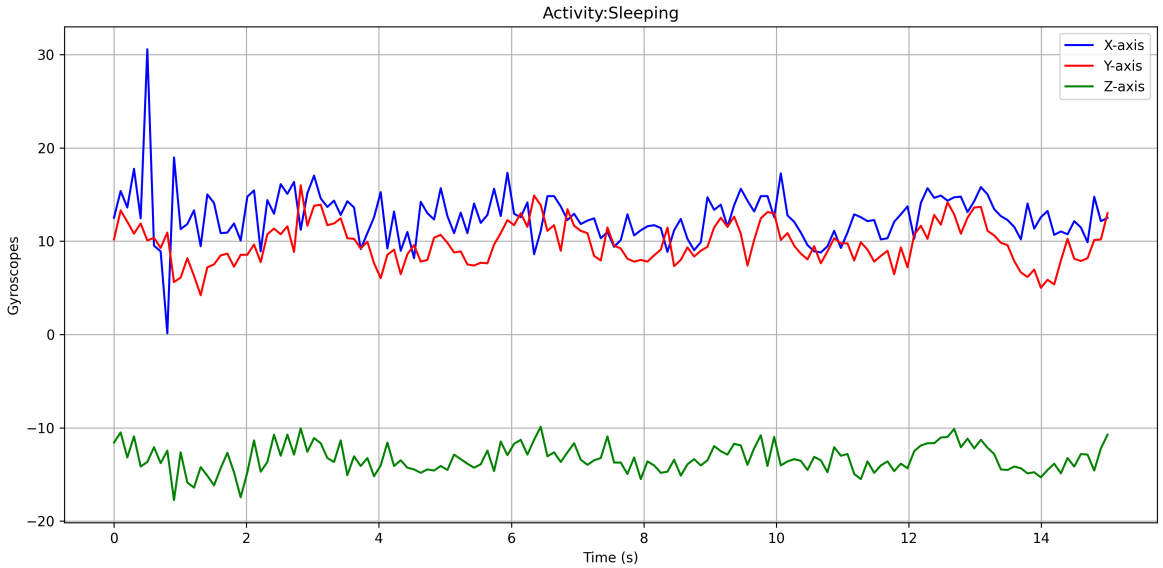

Figure 4.6 : Participant 21bf37e6 Sleeping Gyro

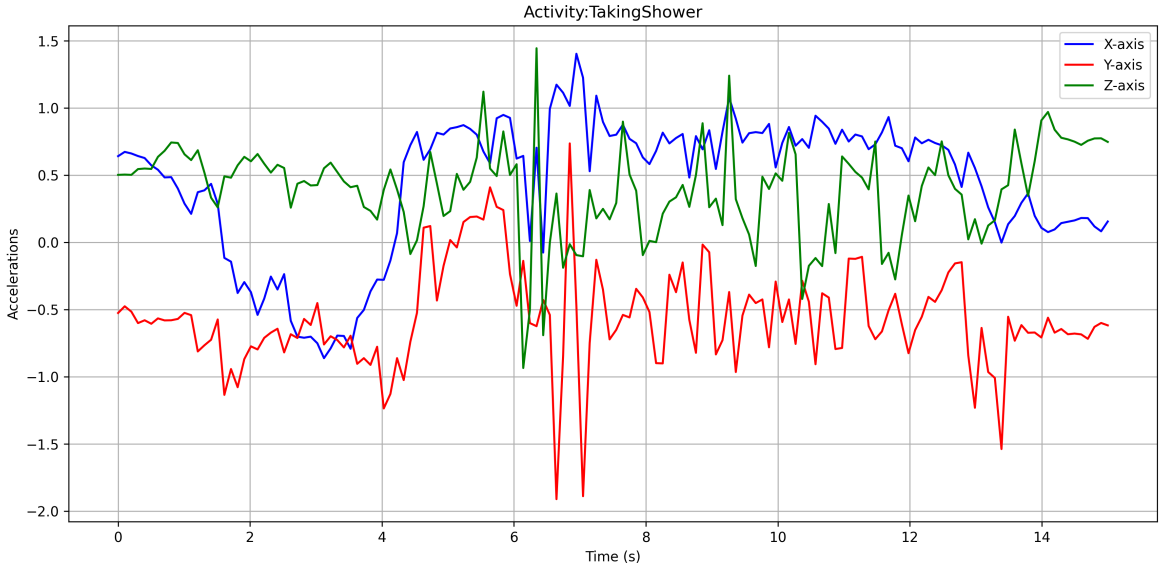

Figure 4.7 : Participant a6f74a8b Taking a shower Acc

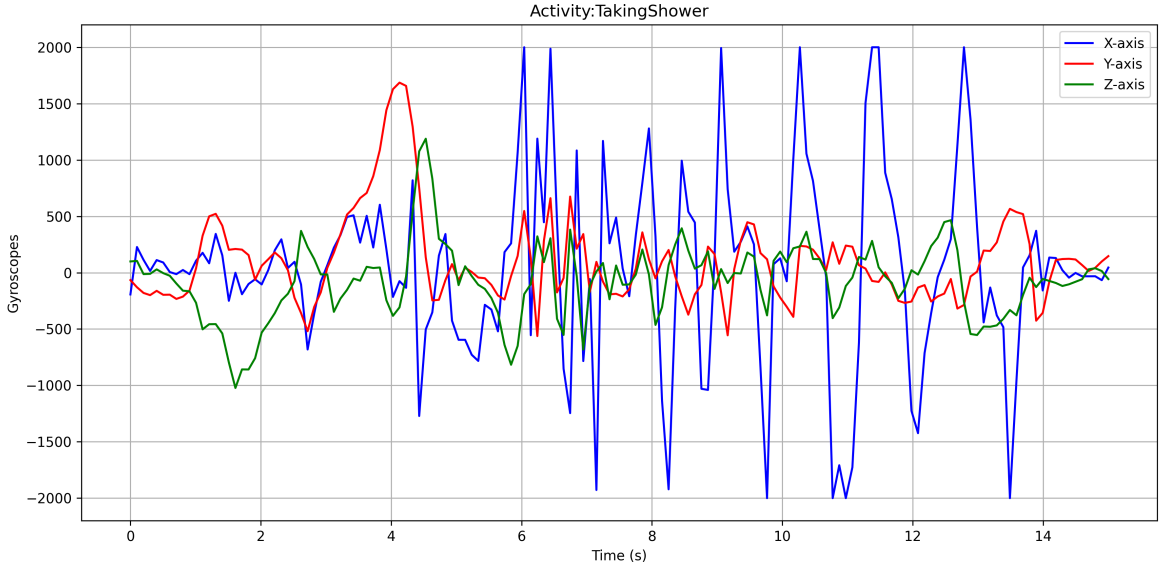

Figure 4.8 : Participant a6f74a8b Taking a shower Gyro

4.2.3 PROMPT DESIGN

Once our datasets are ready, we prepare the prompt that will be used for the experiments. For this part, we have to take advantage of CoT prompting since it usually increases the performance of LLMs as it elicits deeper reasoning from the models [64]. In the context of LLMs, CoT is prompting in a manner that the model goes through a series of intermediate and coherent reasoning steps to reach the final answer. This is demonstrated in Figure 4.12 where the model describes each step of its thought process before giving its final answer. The gain in performance of CoT is demonstrated through experiments in the HARGPT paper [24] and by the work done at EveryWare lab [64]. The HARGPT team saw an increase of up to 33% to the F1 score when using CoT on GPT-4 [39] while the EveryWare lab team documented up to 11% increase when using very large models such as PaLM 540B [10]. One of the best ways to achieve CoT reasoning is to ask the model to *Please make an analysis step by step* within our prompt.

As shown in Figure 4.9, our prompt structure starts with giving the role of an expert of IMU-based activity analysis to the LLM. Following this, we tell the model what the data is, where it comes from and other specifications. Specifically, we tell it that the IMU data consists of three (3) axis of accelerometer data and three (3) axis of gyroscope data, the frequency is at 10Hz, that it comes from a wrist-worn bracelet and the activity is over a period of fifteen (15) seconds.

Following this, we input the main section of the prompt which is shown in Figure 4.10. This consisted of the acceleration data, followed by the gyroscope data. After this, we gave it

```
{"role": "system", "content": "You are an expert of IMU-based human activity analysis."}
{"role": "user", "content": "The IMU data is collected from a bluetooth IMU bracelet containing 3 acceleration axes and 3
gyroscope axes attached to the user's arm with a sampling rate of 10Hz for a periode of 15 seconds. The IMU data is given in
the IMU coordinate frame. The three-axis accelerations and gyroscopes are given below."}
```

Figure 4.9 : Initial messages given at the beginning

```
{"role": "user", "content": "Accelerations:"}
{"role": "user", "content": str(selected_activity['Accelerations'])}
{"role": "user", "content": "Gyroscopes:"}
{"role": "user", "content": str(selected_activity['Gyroscopes'])}
{"role": "user", "content": "The person's action belongs to one of the following categories: " + str(activities_list) +".
Could you please tell me what action the person was doing based on the given information and IMU readings? Please make an
analysis step by step. Please state only one activity you believe the person is doing as the final line in the following
format : \"The activity is : X\" "}
```

Figure 4.10 : Instruction messages

the list of possible activities and asked it to tell us what the individual was doing based on the IMU data while induction CoT reasoning. We also instructed the LLM to give its final answer in a specific format at the end to facilitate the gathering of answers, which can be seen on Figure 4.11 as asking for the format *The activity is: X*.

4.3 EXPERIMENTS

This section details the experiments and presents the results we obtain from each dataset. As mentioned in the previous chapter, we use a Python program to load the activities from the

```
### Instruction: You are an expert of IMU-based human activity analysis.
### Question: The IMU data is collected from a bluetooth IMU bracelet containing 3 acceleration axes and 3 gyroscope axes
attached to the user's arm with a sampling rate of 10Hz for a periode of 15 seconds. The IMU data is given in the IMU
coordinate frame. The three-axis accelerations and gyroscopes are given below.
Accelerations:
x-axis: {...}, y-axis: {...}, z-axis: {...}
Gyroscopes:
x-axis: {...}, y-axis: {...}, z-axis: {...}
The person's action belongs to one of the following categories: <categories_list>. Could you please tell me what action the
person was doing based on the given information and IMU readings? Please make an analysis step by step. Please state only one
activity you believe the person is doing as the final line in the following format : \"The activity is : X\"
```

Figure 4.11 : Full prompt given to LLM

	Precision	Recall	F1-Score
Meta-Llama-3.3-70B-Q8	**0.45**	**0.46**	**0.43**
GPT-4o	0.42	0.43	0.41

Table 4.1 : Capture24 - Initial Results

prepared dataset and insert the data into a prepared prompt which can be seen in Figure 4.11. This prompt is then sent to the model via an API call, either to OpenAI for the GPT-4o model or to our locally hosted Llama model using Llama C++. Our prompt triggers the CoT of the models, as seen in the response from a model in the Figure 4.12, and each response is logged into a JSON file with the correct activity tag. This file is then used to compute the results. Specifically, we calculate the confusion matrix for the experiment with the associated precision, recall and F1-score for each label as well as the compiled results.

4.3.1 CAPTURE24

The first series of experiments are performed on the Capture24 dataset. As mentioned above, this is done to have an initial benchmark and point of comparison. Since this is a public dataset, it allows us to better compare with other works who used this dataset and will grants some initial insights on how the models perform with our methodology. This is essential in order to reduce environmental and methodology variations when comparing with the unseen LIARA dataset.

The first experiment gave poor results. Both Llama 3.3 and GPT-4o gave similar results. As shown in Table 4.1, Llama 3.3 has an overall F1-score of 0.43 while GPT-4o ended with 0.41. Looking at the classification reports for both models, Table 4.2 for Llama 3.3 and Table 4.3 for

```
To determine the activity, we need to analyze the IMU readings, focusing on the nature of activities typically
associated with specific movements and orientations detectable through acceleration and gyroscope data.

### Step-by-Step Analysis:

1. **Data Overview**:
    - **Accelerometer Data**:
        - The data are given in terms of acceleration along three axes (x, y, z).
        - The acceleration values show variations indicating arm movements or changes in posture.
    - **Gyroscope Data**:
        - The data represent angular velocity around three axes (x, y, z).
        - Significant variations in angular velocity suggest rotational movements.

2. **Data Characteristics**:
    - **Accelerometer Observations**:
        - The x-axis data shows noticeable fluctuations, with values ranging from approximately -1.2 to 0.3 g,
        suggesting periodic movements or changes in arm position.
        - The y-axis data ranges from around -1.4 to 0.3 g, which may indicate lateral (side-to-side) movements or
        changes in orientation.
        - The z-axis data is mostly positive and relatively stable, suggesting the arm might be in a steady position
        with some slight movements.
    - **Gyroscope Observations**:
        - The x-axis gyroscope data shows rapid changes, with peaks up to 1999.694 deg/s, indicative of quick
        rotations.
        - The y-axis data shows moderate movement with less extreme peaks, suggesting some rotations but not as
        vigorous as the x-axis.
        - The z-axis data has a range with some peaks and valleys, indicating some rotations along this axis as
        well, but less extreme than x-axis movements.

3. **Activity Analysis**:
    - **Eating**:
        - Likely involves repetitive hand-to-mouth movements, which can cause moderate x and y-axis accelerations
        with periodic gyroscope peaks.
    - **ReadingBook**:
        - Generally involves minimal hand movement with possible slight adjustments, leading to low, stable
        accelerometer readings and minimal gyroscope activity.
    - **Sleeping**:
        - Expected to show very low accelerometer and gyroscope activity due to minimal movement.
    - **TakingShower**:
        - May involve varied arm movements and rotations as the person reaches for soap, shampoo, etc., causing
        higher and more varied accelerometer and gyroscope readings.

4. **Matching Observations to Activities**:
    - The significant variations and rapid changes in gyroscope readings, particularly on the x and z axes, suggest
    active rotational movements.
    - The accelerometer data shows variability, particularly on the x and y axes, suggesting lateral and vertical
    arm movement consistent with active hand use.
    - The overall activity pattern suggests dynamic and varied movement, aligning more closely with activities
    involving frequent arm use and changes in posture.

5. **Conclusion**:
- Given the observed variability and the nature of movements, the data best matches with an activity involving
active use of arms and rotations, such as reaching for objects or adjusting position frequently.

The activity is: TakingShower"
```

Figure 4.12 : Example of model answer

	Precision	Recall	F1-Score
bicycling	0.00	0.00	0.00
sit-stand	0.33	0.28	0.30
sleep	**0.69**	0.39	0.50
walking	0.47	**0.83**	**0.60**

Table 4.2 : Capture24 - Meta Llama 3.3 - Results

	Precision	Recall	F1-Score
bicycling	0.07	0.12	0.09
sit-stand	0.33	0.17	0.23
sleep	**0.56**	**0.74**	**0.63**
walking	0.48	0.45	0.47

Table 4.3 : Capture24 - GPT-4o - Results

GPT-4o, we can see that they performed differently on each activity. Llama 3.3 performed best on the *walking* activity while GPT-4o performed better on the *sleep* activity. Both did terribly for the *bicycling* activity and achieved not-so-different poor results on the *sit-stand* label.

In an attempt to bring the results closer to those reported in the literature, specifically the work of HARGPT [24], we evaluated different approaches to improve the performance of the models on both datasets. The method that results in the most significant performance boost is augmenting the number of samples for each activity. To do this, we randomly sampled three (3) windows of data from each instance of an activity performed by an individual, effectively tripling our number of activities for the experiments. This means that a *bad* random sample of data has a reduced impact on the final results. As seen in Table 4.4, the best performance we obtained with the Capture24 dataset is a F1 score of 0.71 on Llama 3.3 and 0.68 for GPT-4o. The label *sleep* remained the top performer for GPT-4o with a F1-score of 0.85 while both *sleep* and *walking* finished with a F1-score of 0.79 for Llama 3.3. The activity *bicycling* remains the worst performer by far for both models.

	Precision	Recall	F1-Score
Meta-Llama-3.3-70B-Q8	0.74	0.70	0.71
GPT-4o	0.72	0.66	0.68
HARGPT	**0.81**	**0.79**	**0.79**

Table 4.4 : Capture24 - Best Results

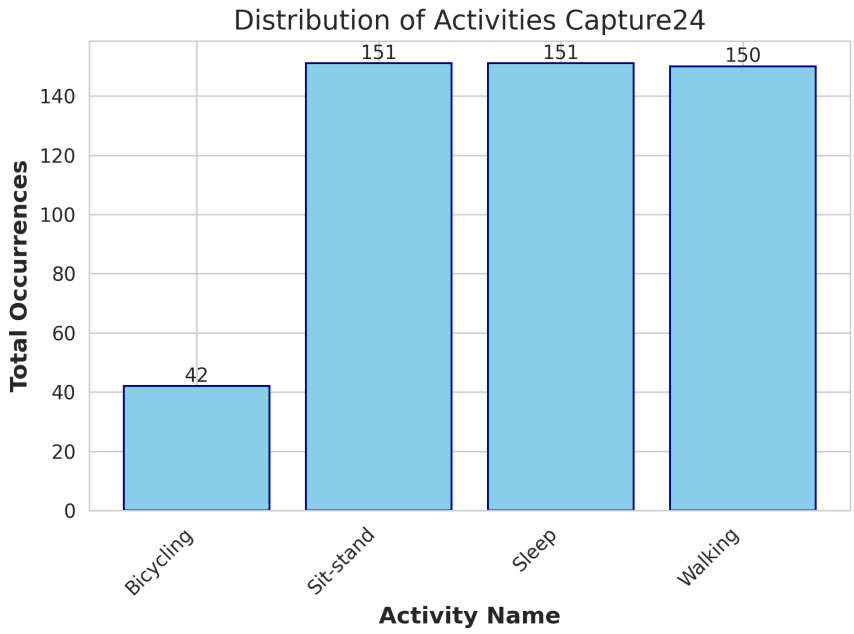

Figure 4.13 : Distribution of Activities - Capture24

It is important to note that in the dataset, the activity labels are not balanced, as can be seen in Figure 4.13. Specifically, the label *bicycling* is more than three times less represented than the other three labels. We chose to keep it this way in the optic that this best represents a real-world scenario and that the dataset Capture24, like ours, attempts to reflect real-world conditions.

	Precision	Recall	F1 Score
Meta-Llama-3.3-70B-Q8	**0.41**	0.25	0.20
GPT-4o	0.36	**0.37**	**0.32**

Table 4.5 : LIARA - Initial Results - 4 Activities

4.3.2 LIARA DATASET

With the experiments of the Capture24 dataset completed, it is time to proceed with the experiments for our in-house and unseen LIARA dataset. Displayed in Table 4.5, the F1-score for Llama 3.3 is 0.20 while it is slightly higher at 0.32 for GPT-4o. Furthermore, we can see that the precision and recall for GPT-4o, at 0.36 and 0.37 respectively, are not too far apart as was the case in the Capture24 experiments for this model. We observe the opposite for Llama 3.3 with a significant gap between the two, 0.41 for the precision and 0.25 for the recall.

Looking at the classification report for the two models, shown in Table 4.6 and 4.7, we can see that both performed very poorly on the sleeping activity with a F1-score of 0.07 for both. Both performed best on the *takingshower* label, with Llama 3.3 achieving an F1-score of 0.41 while GPT-4o achieved 0.45. A significant difference in performance emerged between the two models for the label *readingbook*. Llama 3.3 achieved a very low F1-score of 0.06 while GPT-4o matched the *takingshower* label with 0.45. For both models, the detection of eating activity yielded moderate F1-scores of 0.26 for Llama 3.3 and 0.31 for GPT-4o.

Following these results, we conducted an experiment comprising of three (3) activities instead of four (4). This establishes an additional baseline for comparison prior to proceeding. The objective is to determine if reducing the number of activities by one (1), which represents 25% of all activities included in this scenario, gives a noticeable boost in performance, as it

	precision	recall	F1-score
eating	0.20	0.38	0.26
readingbook	0.11	0.04	0.06
sleeping	**1.00**	0.04	0.07
takingshower	0.33	**0.54**	**0.41**

Table 4.6 : LIARA - Meta Llama 3.3 - Classification Report - 4 Activities

	precision	recall	F1-score
eating	0.31	0.31	0.31
readingbook	**0.56**	0.38	**0.45**
sleeping	0.25	0.04	0.07
takingshower	0.33	**0.69**	**0.45**

Table 4.7 : LIARA - GPT-4o - Classification Report - 4 Activities

should. We remove the activity *Reading* since the biggest gap in performance between the models occurred on this activity and we consider the other three (3) activities to be more *classical* with more distinctive features to discriminate them.

The obtained results yielded no statistically significant improvement in performance. As it can be seen in Table 4.8, the F1-score increased from 0.20 to 0.28 for Llama 3.3. On GPT-4o, we see a more prominent increase in performance for the F1-score, from 0.32 to 0.48. Looking at the classification report for both models, we can see in Table 4.9 and Table 4.10 an increase in F1-score for the label *sleeping* on both models, with the most significant increase on GPT-4o that jumps from 0.07 to 0.51. On the other hand, eating sees a sharp drop on Llama 3.3 while

	Precision	Recall	F1 Score
Meta-Llama-3.3-70B-Q8	0.41	0.35	0.28
GPT-4o	**0.65**	**0.51**	**0.48**

Table 4.8 : LIARA - Initial Results - 3 Activities

	precision	recall	F1-score
eating	0.12	0.12	0.12
sleeping	**0.67**	0.08	0.14
takingshower	0.43	**0.85**	**0.57**

Table 4.9 : LIARA - Meta Llama 3.3 - Classification Report - 3 Activities

	precision	recall	F1-score
eating	0.50	0.23	0.32
sleeping	**1.00**	0.35	0.51
takingshower	0.44	**0.96**	**0.60**

Table 4.10 : LIARA - GPT-4o - Classification Report - 3 Activities

staying relatively the same for GPT-4o. For both Llama 3.3 and GPT-4o, the label *takingshower* remains the strongest performer by far with an F1-score of 0.57 and 0.60, respectively.

Finally, we made the costly decision to attempt an experiment using the whole window size for each activity. The goal is to determine if the small window size we have been providing is a significant limiting factor in the performance of the models. This leads to a significant cost increase per call for the OpenAI API. Running on a locally hosted model required more resources and time. Unfortunately, after a few variations settings in Llama C++ and also trying to use a quantize four (Q4) version, we dropped the locally hosted Meta Llama model for this test since we could not get it to run consistently. This increase in cost and time did not correlate to an increase in performance of the GPT-4o model. As seen in Table 4.11, the performance decreased significantly. The F1-score dropped by half, from 0.48 to 0.24. Looking at the classification report in Table 4.12, we can see that *sleeping* fell back down with a F1-score of 0.00. The activity of *eating* stays around the same with a F1-score of 0.33 while reading came down to 0.23 from 0.45 in the first series of experiments. Finally, the *takingshower*

	Precision	Recall	F1 Score
GPT-4o	0.27	0.31	0.24

Table 4.11 : Results - 4 Activities - Full Data Window

	precision	recall	F1-score
eating	0.36	0.31	0.33
readingbook	**0.44**	0.15	0.23
sleeping	0.00	0.00	0.00
takingshower	0.27	**0.77**	**0.40**

Table 4.12 : GPT-4o Classification Report - 4 Activities - Full Data Window

label remains dominant with a F1-score of 0.40. An insignificant difference from the initial experiment's number of 0.45.

4.4 DISCUSSION

The results were surprising to us and did not confirm our initial hypothesis. We expected to have similar results within a margin of a couple percentage points when compared to results obtained in the work of HARGPT [24]. We also noticed a significant difference between GPT-4o and the Llama 3.3 model when using the LIARA dataset, which did not occur in the case of the Capture24 dataset. We expected GPT-4o to outperform Llama 3.3 considering the former is a state-of-the-art model while the latter is an open-source model and locally hosted, yet it only did in the case of the LIARA dataset. This section encompasses a complete and comprehensive investigation of what exactly is happening and why it is happening.

4.4.1 ACTIVITY LABEL INTERPRETATION

One of the first points to investigate is the data from our laboratory. Previous work has successfully used this data to train deep learning models, so why was the LLM unable to accurately use it to determine the activity being performed by the participant? One of the first clues comes straight from the description of the activity. Let us take a look at the description for the activity *sleeping*, translated from French:

> This activity takes place only once in the room, and the participant may not leave. Before the recording begins, the participant must stand in front of the bed. During the recording, the participant may simulate falling asleep by moving in bed or simulate being in a deep sleep, meaning they will remain completely still. The researcher responsible for data collection must take notes on the participant's activity throughout the recording.

The sleep activity has ambiguity. The participant can be simulating deep sleep or falling asleep; these two acts generate very distinct data. It also begins outside of the bed, so the act of getting in bed generates significant movement. We can see this with the entire acceleration data for a participant going to sleep in Figure 4.14. The first half shows a lot of movement on all axes as the participant gets into bed and, probably, simulates falling asleep which includes a lot of movement. The second half shows the lack of movement we can expect from *sleeping*, with a few spikes from when the participant rotates.

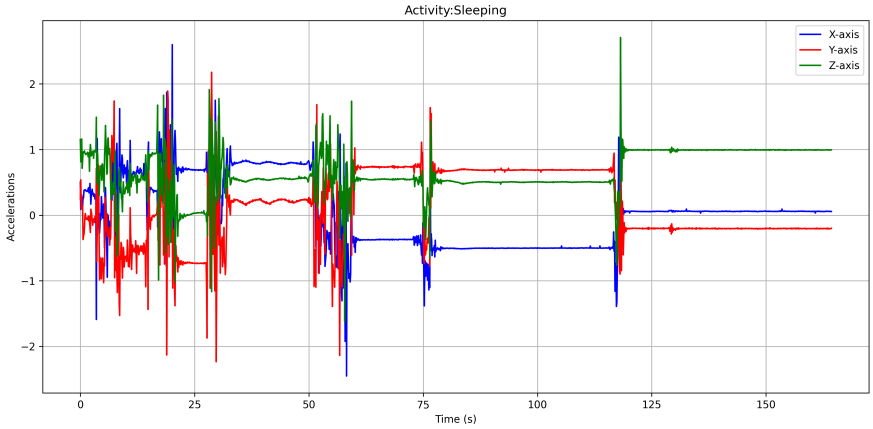

Figure 4.14 : Acceleration data for participant 1b111579 sleeping

By analyzing the responses where the LLM mislabels the sleeping activity, we notice that the model often interprets spikes in movement of the IMU data as being an activity other than sleeping, as seen when it identifies sleeping as taking a shower in one instance:

> Given the dynamic nature of the recorded IMU data with significant movements captured across all axes, indicating wide-ranging repetitive movement[...]

Due to the CoT, the LLM provides a description for each activity for each experiment. This allows us to understand how the model perceives each activity. In the context of sleeping, here are a few descriptions provided by GPT-4o:

> **Sleeping**: This would involve very minimal movement, with the IMU data showing little to no changes except for very slight fluctuations due to minor body adjustments.

Sleeping: This activity involves very little movement, if any. The accelerometer and gyroscope readings should be close to static, with occasional minor adjustments.

Sleeping: The arm is mostly stationary, possibly with small involuntary movements or rotations, but generally low activity.

This explains the poor performance for the activity *sleeping* as the model seems to understand the activity to be something different versus how it is defined as in our dataset. It further demonstrates the importance of having common definitions of labels to reduce such errors. This could be mitigated by providing the description for each activity to the model ahead of time in order to make sure the label is adequately defined.

4.4.2 DATA HAS BEEN SEEN IN TRAINING

A critical factor is whether the LLMs were exposed to the evaluation datasets during their training phase. Considering public datasets, such as Capture24 [6] and HHAR [57], are online and heavily used in HAR research, it is highly likely that they have been seen by the model when it was trained. This has also been suggested by other research projects [19]. This would mean that the impressive results obtained in other works could be, in part, explained by the LLM memorizing that data and its associated label and not that it is capable of analyzing IMU data to infer the activity an individual is performing. It could also explain why GPT-4o and Llama 3.3 performed very similarly on the Capture24 dataset and not on the LIARA dataset. If both models saw it in training then it is reasonable to assume that they would perform similarly

when presented the same prompt and data. Since the LIARA dataset is unseen, it forces the model to try and actually understand the data to classify it. It is at this stage that we can observe the expected superior performance of GPT-4o over Llama 3.3.

4.4.3 ACTIVITIES USED AS A FALLBACK

Analyzing the results, we observed that the activity *takingshower* is heavily selected by both the GPT-4o and LLama 3.3 models compared to other possible labels, even when reducing from four (4) to three (3) activities. Looking at the confusion matrix for both models for all the experiments, shown in Tables 4.13 through 4.17, we can see that both models select the label *takingshower* very heavily over any other label by a wide margin. This activity is quite vague, as in it represents a great variety of possible actions from rinsing, washing your feet, adjusting water temperature, etc. as it can be seen by its description in Figure 4.15, translated from French. Our hypothesis is that the models use this activity as a fallback activity, as it can always *fit* the data we provide. We believe that when there is uncertainty or sections of data that does not fit the model's understanding of the possible labels, it falls back to this activity as an explanation. Additionally, in the case of the LIARA dataset, the activity includes actions before and after entering the shower. This likely compounds the effect and leads to further confusion for the models.

On one hand, it is preferable to avoid including such fallback activities in order to force the model to deal with uncertainty and make a decision on probable activities. On the other hand, such activities are part of the real world and occur in every day scenarios. Omitting such activities in order to boost model performance could serve in certain controlled environments

	eating	readingbook	sleeping	takingshower
eating	**10**	0	0	16
readingbook	19	**1**	0	6
sleeping	12	6	**1**	7
takingshower	10	2	0	**14**

Table 4.13 : LIARA - Meta Llama 3.3 - Confusion Matrix - 4 Activities

	eating	readingbook	sleeping	takingshower
eating	**9**	1	2	14
readingbook	6	**10**	1	9
sleeping	5	7	**1**	13
takingshower	7	0	0	**18**

Table 4.14 : LIARA - GPT-4o - Confusion Matrix - 4 Activities

	eating	sleeping	takingshower
eating	**3**	1	22
sleeping	17	**2**	7
takingshower	4	0	**22**

Table 4.15 : LIARA - Meta Llama 3.3 - Confusion Matrix - 3 Activities

	eating	sleeping	takingshower
eating	**6**	0	20
sleeping	5	**9**	12
takingshower	1	0	**25**

Table 4.16 : LIARA - GPT-4o - Confusion Matrix - 3 Activities

	eating	readingbook	sleeping	takingshower
eating	**8**	2	0	16
readingbook	3	**4**	0	19
sleeping	6	2	**0**	18
takingshower	5	1	0	**20**

Table 4.17 : GPT-4o Confusion Matrix - 4 Acitivities - Full Data Window

```
This activity takes place twice in the bathroom, and the participant cannot
leave this room. Before the recording begins, the participant must have
removed their shoes and be standing on the mat in front of the bathtub.
During the recording, the participant may, for example, simulate turning
the faucet on and off, washing their hair and body, rinsing, etc. The
researcher responsible for data collection must notify the participant
between 20 and 30 seconds before the end of the recording so that they can
perform the necessary actions to complete the activity.
```

Figure 4.15 : *takingshower* activity description, translated from french

and scenarios but would not apply well to real world, dynamic conditions. This is a noticeable weak point of the models when only provided with IMU data. Addressing this shortfall could include providing context to the model (e.g. what room the individual is in) or including data from other modalities (e.g. UWB-radars).

4.4.4 RANDOM WINDOWS CONUNDRUM

Using random windows from the whole data allows us to be able to run experiments since the whole data is too large to be efficiently used with a LLM or would be too costly when using an online platform, if they even allow the use of such large context windows. Furthermore, using random windows allows us to better simulate dynamic and live HAR tasks that use a much smaller amount of data from the last few seconds. It also brings to light another noticeable weak point: small data anomalies. For instance, utilizing a 10-second window may capture transient movements, such as minor positional shifts or dream-enacted motor activity during sleep. If the model captures this data, it will be led to believe that the individual is performing another activity due to the spike in acceleration and gyroscope data. By selecting a random window, we also expose ourselves to the same risk as we run the chance of randomly selecting

a window in which there is abnormal data for the labeled activity. One could argue that this is a real world scenario and that such data is never clean.

CHAPITRE V

REFINING FOR BEST RESULTS

5.1 OBJECTIVE

Following the results obtained with the initial experiments and the analysis performed, we implement a series of experiments to evaluate how to increase the performance of the models with our laboratory dataset. Considering that our dataset has not been made public and was not available to the models when they were trained, we believe that we have a unique opportunity to research various approaches and techniques on how they impact LLM performance on HAR tasks in a real world scenario.

5.2 MODIFICATIONS

The analysis of the results from the initial experiments allowed us to deepen our understanding of how the LLMs work in the context HAR with our dataset. As discussed in the previous chapter, we identified multiple limitations and possible causes which could, in part, explain the poor performance we observed. In order to better understand what is happening and validate

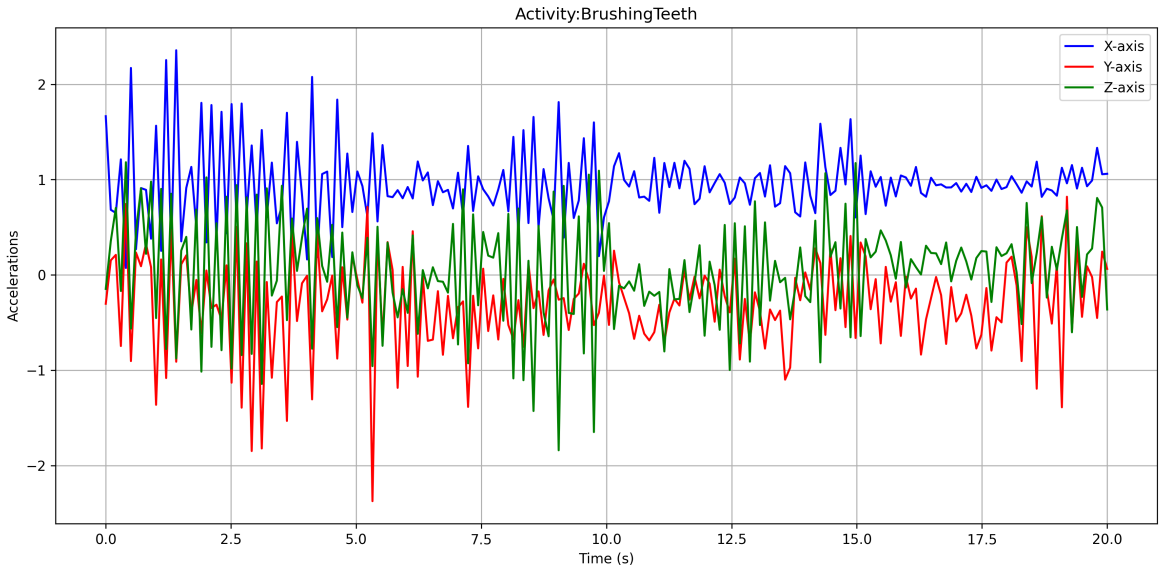

Figure 5.1 : Participant 4153eaad brushingteeth

our observations, we proceed with a a few modifications to how we pre-process our data for the following experiments with the hypothesis that these changes will yield better results.

5.2.1 CHOOSING OUR ACTIVITIES

The first step was the selection of activities characterized by high inter-participant consistency, simplicity, and distinct, highly discriminative data profiles. The goal for this was to make it easier for a LLM to correctly identify which activity the participant was performing. With this in mind, we attempted various combinations in order to determine which activities offered the best chance of success without making it too easy for the model to perform well. The goal is to keep it as close as possible to a real world setting. After considerations, we settled for the following combination of activities: *brushingteeth*, *drinking*, *readingbook*, *sleeping*.

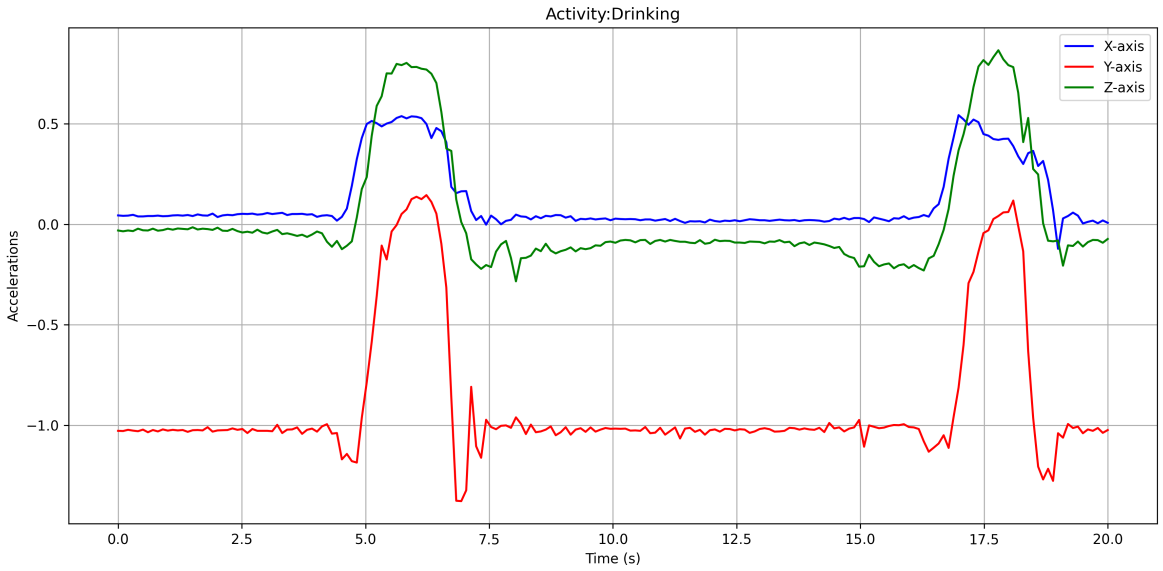

Figure 5.2 : Participant 8c4bad9c drinking

We chose these activities based on the distinctive data they generated on the accelerometer. Looking at Figures 5.1 through 5.4, we see how the data they generated are very different from one another and, to an extent, even a human could interpret the activity from them. The *sleeping* activity is very flat, indicating no movement, while we see the continuous up and down motion associated with brushing your teeth for *brushingteeth*. For drinking, we see elongated spikes that correlate to slow the slow up-down movement of taking a sip with resting periods in between. Finally, for *readingbook*, we can see a long resting period followed by activity from shuffling the book and turning a page.

5.2.2 INCREASING WINDOW LENGTH

The last of our initial experiments demonstrated that significantly increasing the size of the window did not bring a significant performance increase. However, the random window conundrum we discussed in the discussion section of the previous chapter leads us to believe

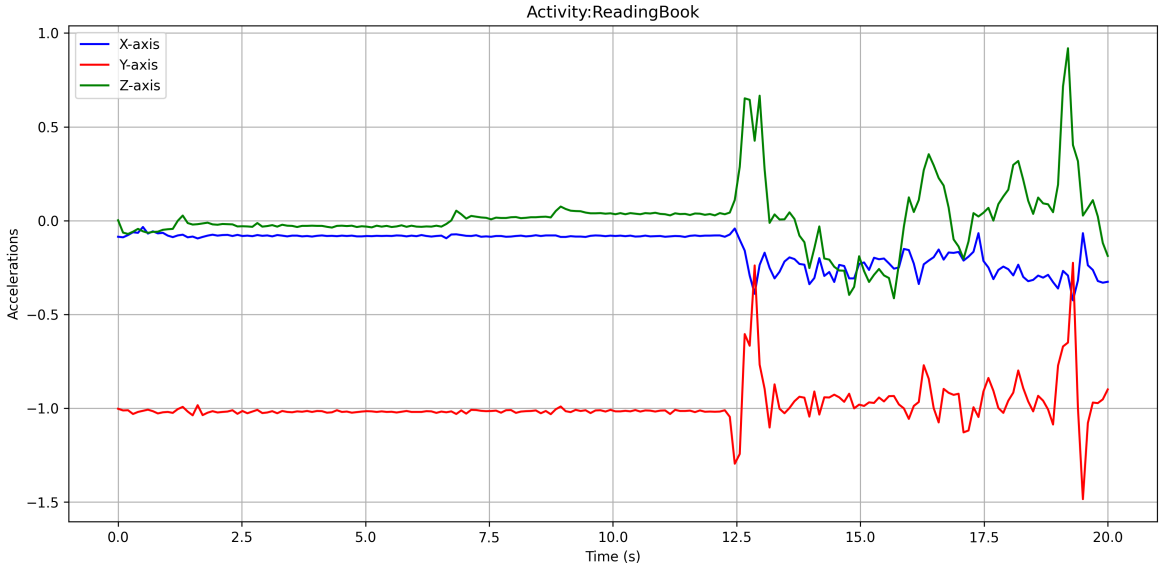

Figure 5.3 : Participant cd8abd36 readingbook

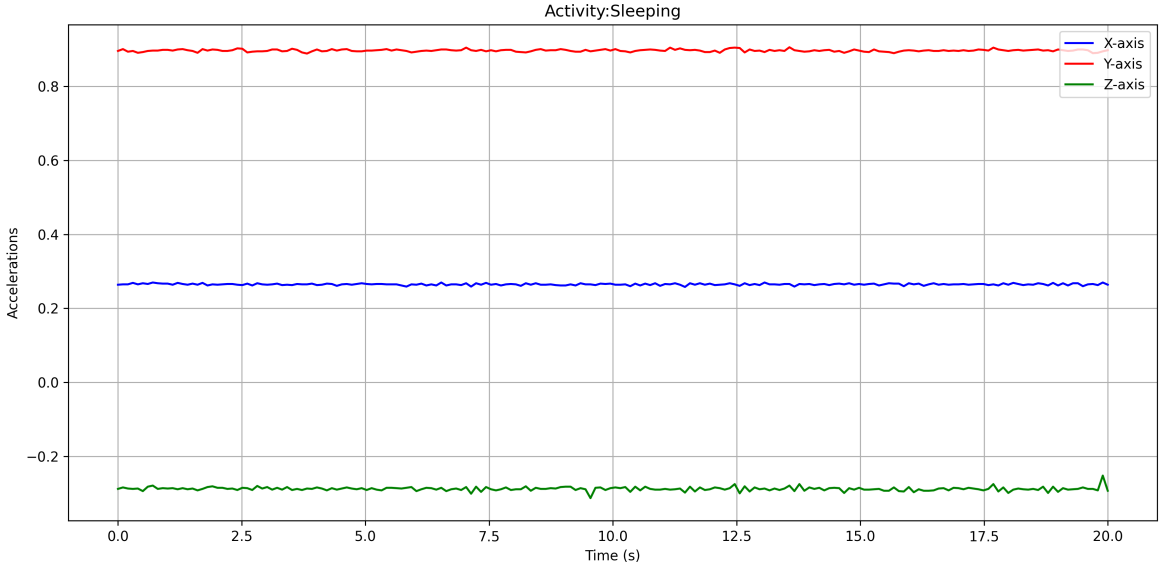

Figure 5.4 : Participant 80a2ce12 sleeping

that a small increase in window size to help overcome data anomalies would be beneficial. This is especially the case for the activity *sleeping* which, as described in the previous chapter, starts out of bed and can include more movement due to our definition than what the model believes. We choose to increase the random window length for the activities from fifteen (15) to twenty (20) seconds for the following experiments.

5.3 EXPERIMENTS

For the next series of experiments, we do not pursue with the locally hosted Llama model. Considering that it resulted in poor performance compared to the GPT series from OpenAI, we concentrate our efforts on the GPT models since it better reflects the performance we can expect from using LLMs for HAR tasks. As a reminder, we only evaluate the LIARA dataset in these experiments as we know it is unseen by the models during their training and offers the best real-world scenario.

5.3.1 REMOVING GYROSCOPE DATA

The Capture24 dataset does not include any gyroscope data, it only has the acceleration data. This leads us to the hypothesis that the gyroscope data confuses the model instead of helping it. Our hypothesis is that the gyroscope data contradicts the acceleration data in terms of what label the model believes it belongs to. For instance, the model might classify the acceleration data as belonging to *sleeping* yet, when it sees the gyroscope data it interprets it as not belonging to *sleeping*. We decided to run an experiment in which we include the gyroscope

	precision	recall	F1-score
GPT-4o Acc + Gyro	0.43	0.40	0.32
GPT-4o Acc	**0.44**	**0.44**	**0.43**

Table 5.1 : Gyro vs Acc - Total Results

	precision	recall	F1-score
brushingteeth	0.39	**0.90**	**0.54**
drinking	0.48	0.21	0.29
readingbook	0.33	0.12	0.17
sleeping	**0.50**	0.04	0.07

Table 5.2 : GPT-4o Acc + Gyro - Classification Report

data in one run and excluded it in the other. This approach isolates the impact of gyroscopic inputs on LLM accuracy, allowing for a direct assessment of their contribution to HAR tasks.

As we see in the Table 5.1, removing the gyroscope data led to an increase in performance. This indicates that the extra data only confuses the model as it could, in part, contradict its perception of the acceleration data. It can also indicate that the model prioritizes certain elements due to the ordering of the information in the prompt, as some works seen in the literature indicate that the ordering has an affect on performance [37] [9]. Looking at the classification report in Tables 5.2 and 5.3, we observe that all labels had an increase in F1-score when removing the gyroscope data. All except *brushingteeth* saw a significant increase. Looking at the confusion matrix for both experiments in Tables 5.4 and 5.5, we see that when the gyroscope data was included, the *brushingteeth* label was heavily selected compared to the other activities, leading it to be a fallback activity as *takingshower* was in the previous chapter. But when we remove the gyroscope data, it is far less heavily selected compared to the other activities.

	precision	recall	F1-score
brushingteeth	0.49	**0.65**	**0.56**
drinking	0.43	0.40	0.42
readingbook	0.23	0.23	0.23
sleeping	**0.58**	0.27	0.37

Table 5.3 : GPT-4o Acc - Classification Report

	brushingteeth	drinking	readingbook	sleeping
brushingteeth	**47**	5	0	0
drinking	40	**11**	0	1
readingbook	17	6	**3**	0
sleeping	18	1	6	**1**

Table 5.4 : GPT-4o Acc + Gyro - Confusion Matrix

	brushingteeth	drinking	readingbook	sleeping
brushingteeth	**34**	14	4	0
drinking	20	**21**	10	1
readingbook	6	10	**6**	4
sleeping	9	4	6	**7**

Table 5.5 : GPT-4o Acc - Confusion Matrix

	precision	recall	F1-score
GPT-5.2 Acc	0.67	0.71	0.68

Table 5.6 : GPT-5.2 Acc - Total Results

	precision	recall	F1-score
brushingteeth	**0.84**	**0.92**	**0.88**
drinking	0.69	**0.92**	0.79
readingbook	0.53	0.31	0.39
sleeping	0.43	0.23	0.30

Table 5.7 : GPT-5.2 Acc - Classification Report

5.3.2 UPGRADING THE MODEL

At this stage in our experimentation, new state-of-the-art LLMs have been released by OpenAI in the form of the GPT-5 series. We decided to run an experiment in order to evaluate GPT-5.2 from OpenAI and determine if these new models bring an increase in performance. We run the same experiment as we did on GPT-4o above with just the acceleration data on GPT-5.2.

Looking at the final results in Table 5.6, we see a massive jump in performance compared to the GPT-4o with acceleration only in Table 5.1. The F1-score increased from 0.43 for GPT-4o to 0.68 for GPT-5.2. With the classification report in Table 5.7, we observe that the model performs well for the labels *brushingteeth* and *drinking*, with a F1-score of 0.88 and 0.79, respectively. Yet, it performs poorly for *readingbook* and *sleeping*, with F1-scores of 0.39 and 0.30, respectively. With the confusion matrix in Table 5.8 compared to the previous experiment in Table 5.5, we notice that GPT-5.2 was far more capable in differentiating between the *drinking* and *brushingteeth* activities. It also performed slightly better for *readingbook* but was somewhat worse for *sleeping*.

	brushingteeth	drinking	readingbook	sleeping
brushingteeth	**48**	3	0	1
drinking	2	**48**	0	2
readingbook	3	10	**8**	5
sleeping	4	9	7	**6**

Table 5.8 : GPT-5.2 Acc - Confusion Matrix

	precision	recall	F1-score
GPT-5.2 Acc	0.52	0.42	0.39

Table 5.9 : GPT-5.2 Acc with sleeping description - Total Results

5.3.3 ADDING A DESCRIPTION OF THE ACTIVITY

In the previous experiment, we notice that the activity *sleeping* still performs very poorly compared to the other activities. Considering that the *sleeping* activity we defined in our dataset includes getting in bed and the act of moving when trying to fall asleep, which is different than how the LLMs define sleeping (i.e. almost no movement), we are interested in providing our description of *sleeping* in the prompt to determine if this can assist the model.

Looking at the total results found in Table 5.9, we notice a significant drop in performance compared to the previous experiment. The final F1-score fell from 0.69 to 0.39. When looking at the results in more detail through the classification report and confusion matrix, Tables 5.10 and 5.11, we see that *sleeping* essentially becomes a fallback activity. Since this activity can now represent a great variety of different acceleration data, the model selects it a majority of the time. Way more than any other label. This leads to a sharp decrease in overall model performance for this experiment. Although *sleeping* gets a small increase in

	precision	recall	F1-score
brushingteeth	**0.84**	0.62	**0.71**
drinking	0.50	0.13	0.21
readingbook	0.20	0.04	0.06
sleeping	0.25	**0.96**	0.40

Table 5.10 : GPT-5.2 Acc with sleeping description - Classification Report

	brushingteeth	drinking	readingbook	sleeping
brushingteeth	**36**	4	1	15
drinking	4	**7**	3	38
readingbook	1	3	**1**	21
sleeping	1	0	0	**25**

Table 5.11 : GPT-5.2 Acc with sleeping description - Confusion Matrix

F1-score performance from 0.30 to 0.40, *drinking*, *readingbook* and *brushingteeth* see a drop in F1-score performance.

5.3.4 EXTRACTING FEATURES

For another experiment, we are interested to see if the LLMs perform better on features instead of the raw IMU data. For this, we extract common features for each axis such as the mean, max, min, standard, etc. as well as skewness, kurtosis, Root mean square (RMS) and more for a total of sixty (60) different features. We use GPT-5.2 for this experiment and use features for both the accelerometer and gyroscope data.

- **Accelerometer & Gyroscope Features for each axis (42)**
  - Mean: Average acceleration / angular velocity
  - Std: Standard deviation

	precision	recall	F1-score
GPT-5.2 Acc	0.45	0.42	0.33

Table 5.12 : GPT-5.2 Features - Total Results

  - Min / Max: Extreme values
  - RMS: Root Mean Square
  - Skewness: Asymmetry distribution
  - Kurtosis: Peakedness of distribution

- **Magnitude & Signal Energy Features (5)**
  - AccMag (Mean + Std): Statistical metrics for the total acceleration vector magnitude
  - AccMag Energy: The total energy of the accelerometer signal
  - GyroMag (Mean, Std): Statistical metrics for the total angular velocity magnitude

- **Frequency-Domain Features for each axis (12)**
  - DomFre: Dominant Frequency
  - MaxFFTMag: Magnitude of the dominant frequency

- **Orientation Features (1)**
  - Orientation Tilt Cos: The cosine of the tilt angle, representing the device's orientation relative to gravity.

Looking at the results, Table 5.12, the model performs very poorly. The F1-score falls to a low of 0.33, lower than the previous experiment and far lower than the best case of 0.68 seen so far. Taking a look at the classification report, Table 5.13, and the confusion matrix,

	precision	recall	F1-score
brushingteeth	0.40	**0.98**	**0.57**
drinking	**0.75**	0.17	0.28
readingbook	0.38	0.19	0.26
sleeping	0.00	0.00	0.00

Table 5.13 : GPT-5.2 Features - Classification Report

	brushingteeth	drinking	readingbook	sleeping
brushingteeth	**51**	0	0	1
drinking	36	**9**	5	2
readingbook	17	3	**5**	1
sleeping	23	0	3	**0**

Table 5.14 : GPT-5.2 Features - Confusion Matrix

Table 5.14, we observe the model attributing the label of *brushingteeth* to the great majority of activities. This performs similarly to the activity *takingshower* in the first series of experiment as in, it is being used as a fallback activity that can always be reasoned to fit the data.

5.3.5 ALL FOURTEEN ACTIVITIES

As a final experiment, we included the full LIARA activities list of fourteen (14) distinct labels. The objective is to determine how well the methodology and model can scale with a much greater variety of activities that can overlap in terms of what their data looks like. This also more closely represents a real-world scenario and gets closer to the results that we would obtain in our laboratory running the model for real-time inference. For this experiment, we do not explain any of the labels to the model beforehand. Hence, the model must understand the label and determine if the data it receives fits. We also bump the random window to thirty (30) seconds so the model has a better picture of what is happening.

	precision	recall	F1-score
GPT-5.2 Acc	0.13	0.21	0.15

Table 5.15 : GPT-5.2 - All Activities - Total Results

	precision	recall	F1-score
brushingteeth	0.26	**0.63**	0.37
doinghousework	0.00	0.00	0.00
drinking	0.27	0.48	0.34
eating	0.09	0.15	0.12
gettingdressedundressed	0.00	0.00	0.00
goingtoilet	0.00	0.00	0.00
puttingawaycleanlaundry	0.00	0.00	0.00
puttingawaydishes	0.08	0.02	0.03
readingbook	0.05	0.04	0.04
resting	**0.44**	0.54	**0.48**
sleeping	0.17	0.12	0.14
takingshower	0.00	0.00	0.00
usingcompsphone	0.12	0.42	0.18
washingdishes	0.12	0.31	0.17

Table 5.16 : GPT-5.2 - All Activities - Classification Report

With the results displayed in Table 5.15, we see the F1-score has come down to 0.15. To better understand these results, we take a look at the classification report in Table 5.16. Here, we observe that the best performing label is *resting* at 0.48 for its F1-score and with *brushingteeth* and *drinking* doing well compared with the rest. Interestingly, we notice that many activities have a F1-score of 0.00: *doinghousework*, *gettingdressedundressed*, *goingtoilet*, *puttingawaycleanlaundry*, *takingshower*. The remainder of the labels perform poorly with single digit or in the teens F1-score.

	precision	recall	F1-score
GPT-5.4 Acc	0.71	0.72	0.72

Table 5.17 : GPT-5.4 Acc - Total Results

	precision	recall	F1-score
brushingteeth	**0.81**	**0.92**	**0.86**
drinking	0.69	0.77	0.73
eating	0.45	0.35	0.39
sleeping	0.79	0.58	0.67

Table 5.18 : GPT-5.4 Acc - Classification Report

5.3.6 COMBINING WHAT WE LEARNED

Following the experiments, we ran an experiment grouping what we learned so far to see if it would improve performances. As such, we increased the random window size to thirty (30) seconds of acceleration data, which matches the work of AccNet24 [14]. We also replaced the label *readingbook* by *eating* since GPT-5.2 seemed to have difficulties differentiating *readingbook* and *sleeping*, as seen in Table 5.8. Additionally, a new version of GPT-5 has come out and we will use this version, GPT-5.4.

As shown in Table 5.17, the performance saw a small increase to an F1-score of 0.72, up from 0.68 of the best experiment so far in Section 5.3.2. Looking at the classification report and confusion matrix, Tables 5.18 and 5.19 respectively, we can see we have a small decrease in performance for *brushingteeth* and *drinking*, where both seem to get confused with *eating* more than they did with *readingbook*. *Eating* performs the same as *readingbook*. But we observe that *sleeping* has seen a huge bump in performance, from an F1-score of 0.30 up to 0.67. This is what results in the better overall score.

	brushingteeth	drinking	eating	sleeping
brushingteeth	**48**	1	3	0
drinking	2	**40**	8	2
eating	8	7	**9**	2
sleeping	1	10	0	**15**

Table 5.19 : GPT-5.4 Acc - Confusion Matrix

	precision	recall	F1-score
RF Acc	0.89	0.88	0.88

Table 5.20 : Random Forest - Total Results

5.3.7 THE RANDOM FOREST

It remains important to benchmark models against traditional algorithms. If we cannot outperform them, we are not really solving a real problem. In that spirit, we trained a very simple random forest on our dataset. For this, we kept the full window of data to capture as much information as possible and since we are not limited by the size of the data. The only pre-processing we implemented is down-sampling from 50Hz to 10Hz. For the random forest, we selected seven (7) features to train on: Mean, Std, Min, Max, Median, 25% percentile and 75% percentile. We split the data into a train and test dataset with a 50-50 split to evaluate the random forest on as much data as possible while having a model that performs well. We also set the random state to forty-two (42).

As shown in Table 5.20, the random forest achieves a F1-score of 0.88. This outcome outperforms all tested LLM configurations, despite extensive activity selection, continuous optimization of the data pre-processing pipeline, and rigorous post-hoc analysis regarding LLM dataset interactions. A simple random forest trained in seconds outperforms everything so far.

	precision	recall	F1-score
brushingteeth	0.92	0.92	**0.92**
drinking	0.87	**0.93**	0.90
readingbook	**1.00**	0.69	0.82
sleeping	0.75	0.90	0.82

Table 5.21 : Random Forest - Classification Report

	brushingteeth	drinking	readingbook	sleeping
brushingteeth	**24**	0	0	2
drinking	1	**27**	0	1
readingbook	0	4	**9**	0
sleeping	1	0	0	**9**

Table 5.22 : Random Forest - Confusion Matrix

Looking at the classification report in Table 5.21, we observe that every label has a F1-score of 0.82 or higher. The highest performing activity remains *brushingteeth* with a F1-score of 0.92, but the previously struggling *sleeping* activity is up to 0.82. With the confusion matrix in Table 5.22, we notice that most errors are between *drinking* and *readingbook*. We also observe a few *sleeping* labels being placed with *drinking* and *brushingteeth*, probably due to nature of the activity our participants performed as discussed previously.

5.4 DISCUSSION

Through these experiments, we are able to further our understanding of the models and how they function. We validated some of our hypothesis, such as that certain activities can be used as a fallback activity when the data *fits* most labels. This illustrates the need to carefully select the activities used and also heavily suggests that LLMs might not be able to perform adequately in a real world scenario where inhabitants perform a dozens if not over a hundred

various activities in daily living. This last point is demonstrated in the final experiment in which all fourteen (14) labels are included and the model performance drops significantly.

Analyzing the final experiment where all fourteen (14) activity labels are used, it is very interesting that five (5) of the activities have an F1-score of 0.00. This indicates that these activity labels are practically never selected by the model across the near five hundred and fifty (550) activities that we tested during the experiment. Curiously, *takingshower*, which was observed being a fallback activity in chapter four (4), is among the labels not selected throughout the experiment. This leads to the hypothesis that, in the current context and methodology, the model eliminates activities that are vague by its definition of the label. An analysis of the five (5) activity labels yielding an F1-score of 0.00 suggests high statistical variance or class overlap. Consequently, the model failed to learn distinct patterns for these classes. Greater number of activities could possibly lead to the mitigation of the fallback activity issue we have observed so far.

Furthermore, our experiments indicate that LLMs do not perform well when dealing with raw, numeric data. Gyroscope data, something that we thought would result in better performance, combined with acceleration data, actually confuses the models and leads them to be less accurate. This is demonstrated by the gain in performance when removing it from the prompt. Also, the bad performance when given only features confirms the weakness of LLMs when dealing with numeric data exclusively. Although, one could argue that too many features were provided to the models and that lead to confusion in the same manner as when the gyroscope data was included.

Additionally, the fact that a simple random forest with a small set of features was able to vastly outperform state-of-the-art models really stands out. It further validates that LLMs are not efficient when being given numerical data. It also reminds us about the need to validate the work that we do as scientists against simple solutions and traditional algorithms. On the one hand, it is a blow to our prospect of using LLMs in their current state for HAR tasks. On the other hand, it is a powerful reminder and encouragement that traditional machine learning algorithms still have their place among the resource-intensive models.

Ultimately, we are reminded about the rapid advancements of LLMs. From one generation, GPT-4o -> GPT-5.2, we see a significant jump in performance. Such generational improvements in foundation models indicates the possibility that they could become far more efficient and useful for HAR tasks in the near future. We strongly believe that with the ideal activities selected and further data pre-processing refinement, we would have been able to match the performance demonstrated in the previous works with our dataset. On the other hand, this would not represent a real-world scenario and goes against the objective of this master's project.

Finally, combining all that we learned in an experiment did show a small bump in F1-score, although this was mostly due to the bump in performance from the label *sleeping* more than doubling its F1-score. This can be due to a multitude of factors. Maybe we got lucky in the random windows we generated, or the thirty (30) seconds window actually helps the model not get confused by abnormal data. It can also be the new model version that is better or simply the fact that the model did not have to take into consideration *readingbook*, which could generate similar data. In our opinion, it is a combination of all these factors. This further validates our point that achieving a good result with LLMs for HAR tasks requires serious thought, consideration and work of the data and pipeline to work well.

CHAPITRE VI

DISCUSSION

This chapter focuses on all the different aspects encountered during the research project. What went right, what went wrong, the different limitations we encountered, how the state-of-the-art has evolved, etc. The work covered in this master's thesis spans multiple iterations, technical challenges and advancements in the field of artificial intelligence. We learned a lot concerning the functioning of LLMs, differentiating factors between HAR datasets and the real world constraints around HAR tasks. Here, we share the insights we gained throughout the research project.

6.1 CAPTURE24 - WHAT HAPPENED?

A normal question to ask is why did we not achieve the same level of performance with the Capture24 dataset as the work done in HARGPT [24]? Our hypothesis comes down to how the data was pre-processed before being provided to the LLM. Unfortunately, we do not have all the information on how the data was prepared in their original project. We assume they must have selected windows of data, as the whole data is too large of a context for the model. We do not know how it was done. Judging from the significant differences in performances, it was

obviously done using a different method than how we proceed. We also do not know the size of the dataset they opted to use for testing. We opted not to spend too much time attempting to replicate their exact results as our goal was to have an initial benchmark to compare our unseen dataset and then work from there. Additionally, all attempts to perfectly match their data would be pure conjecture on our part.

6.2 HOSTING A LLM LOCALLY

One of the objectives of this research project is to determine the feasibility of hosting our models locally for HAR tasks. Self-hosting in the context of a smart home for HAR has numerous advantages, with the two primary being increased privacy and cost reduction. As seen in the research project, making continuous API calls to online models, provided by the likes of OpenAI, Google, Anthropic, etc., rapidly incurs significant costs. We completed a small set of controlled experiments using OpenAI's online models in this project and incurred costs over 150$ USD. If this was to be a solution that runs continuously in a smart home, costs could reach well over 1000$ USD monthly. Such a cost would be prohibitive for many potential users and limit adoption. Furthermore, continuously providing sensitive health-related information to an online platform presents privacy and security implications. This information could be used to the disadvantage of the inhabitant by associated third-parties (e.g. insurance providers) or lead to a major vulnerability in the case of a data breach. With security concerns increasing among the general public [43], such a solution would likely be a major irritant for most scenarios.

6.2.1 SHARED, OFFLINE MACHINE RUN BY A SMALL INSTITUTION

As mentioned in the experimental setup chapter, our laboratory possesses a machine with high-end consumer performance hardware. This hardware, combined with knowledge at the time, had us confident in our ability to run state-of-the-art open LLMs with large numbers of active embeddings. Unfortunately, we came to the realization that maintaining hardware for such tasks requires considerable time and technical know-how. In our laboratory's context, the hardware is kept in a practically offline state due to security considerations from the university's Information Technology (IT) staff. This means that we only have internet access in the context of installing packages from reputable package managers such as APT and pip (Python) and are not able to download or install packages freely from the web. This greatly limits the solutions we can use for self-hosting models and also means we find ourselves generally unable to update various software on the machine that is a few years old. As LLMs are a recent technology and evolve rapidly, this is a highly limiting constraint for our use case.

Furthermore, this machine is shared among all the scientists of the LIARA laboratory. This results in resource constraints on the machine which is particularly limiting in our use case, as we needed all of the hardware resources to run the Meta Llama 70B models, restricting the times we can run experiments. Additionally, the machine is configured with a shared user space meaning that all scientists share the same versions of drivers and software installed. This results in us being unable to increase the version of certain critical software components such as Python and, crucially, CUDA drivers. Ultimately, the compounding effect of the combination of factors leads to significant difficulties in being able to run the desired models locally with the optimal configuration.

6.2.2 COMPLEXITY OF FINDING THE RIGHT PARAMETERS

Locally hosting an LLM is more complex than simply having the correct hardware and software to run the model. Optimizing the parameters so it runs as efficiently as possible is the real challenge we faced. This requires further knowledge about low end hardware-software optimization than we anticipated. In our laboratory, we used Llama.cpp as it is the best solution we can use given our constraints. There are hundreds of different parameters that can be used when configuring the hosting server. Some can be combined together and some cannot. Exactly what they do requires specific knowledge about how LLMs work, hardware specifications and even driver code. This results in a significant time investment in order to efficiently run a specific model on precise hardware and is subject to change as the technology evolves. In Figure 6.1, we display a section of the Llama.cpp project page that enumerates a subsection of possible options that can be configured. We absolutely believe that this is achievable; entire communities are devoted to self-hosting LLMs. Our realization comes from the fact that it is not something that is done quickly or simply. Rather, it requires a significant time investment and, ideally, involves an individual with the experience and knowledge of running IT infrastructure. The platforms of OpenAI, Anthropic and such hide this complexity within their refined user interface and rapid answers. It is not to be negated when running a model locally for research purposes, as it has a significant impact on model performance and time necessary for implementation.

`--perf, --no-perf`	whether to enable internal libllama performance timings (default: false) (env: LLAMA_ARG_PERF)
`-e, --escape, --no-escape`	whether to process escapes sequences (\n, \r, \t, ', ", \) (default: true)
`--rope-scaling {none,linear,yarn}`	RoPE frequency scaling method, defaults to linear unless specified by the model (env: LLAMA_ARG_ROPE_SCALING_TYPE)
`--rope-scale N`	RoPE context scaling factor, expands context by a factor of N (env: LLAMA_ARG_ROPE_SCALE)
`--rope-freq-base N`	RoPE base frequency, used by NTK-aware scaling (default: loaded from model) (env: LLAMA_ARG_ROPE_FREQ_BASE)
`--rope-freq-scale N`	RoPE frequency scaling factor, expands context by a factor of 1/N (env: LLAMA_ARG_ROPE_FREQ_SCALE)
`--yarn-orig-ctx N`	YaRN: original context size of model (default: 0 = model training context size) (env: LLAMA_ARG_YARN_ORIG_CTX)
`--yarn-ext-factor N`	YaRN: extrapolation mix factor (default: -1.00, 0.0 = full interpolation) (env: LLAMA_ARG_YARN_EXT_FACTOR)
`--yarn-attn-factor N`	YaRN: scale sqrt(t) or attention magnitude (default: -1.00) (env: LLAMA_ARG_YARN_ATTN_FACTOR)
`--yarn-beta-slow N`	YaRN: high correction dim or alpha (default: -1.00) (env: LLAMA_ARG_YARN_BETA_SLOW)
`--yarn-beta-fast N`	YaRN: low correction dim or beta (default: -1.00) (env: LLAMA_ARG_YARN_BETA_FAST)
`-kvo, --kv-offload, -nkvo, --no-kv-offload`	whether to enable KV cache offloading (default: enabled) (env: LLAMA_ARG_KV_OFFLOAD)
`--repack, -nr, --no-repack`	whether to enable weight repacking (default: enabled) (env: LLAMA_ARG_REPACK)
`--no-host`	bypass host buffer allowing extra buffers to be used (env: LLAMA_ARG_NO_HOST)
`-ctk, --cache-type-k TYPE`	KV cache data type for K allowed values: f32, f16, bf16, q8_0, q4_0, q4_1, iq4_nl, q5_0, q5_1 (default: f16) (env: LLAMA_ARG_CACHE_TYPE_K)
`-ctv, --cache-type-v TYPE`	KV cache data type for V allowed values: f32, f16, bf16, q8_0, q4_0, q4_1, iq4_nl, q5_0, q5_1 (default: f16) (env: LLAMA_ARG_CACHE_TYPE_V)
`-dt, --defrag-thold N`	KV cache defragmentation threshold (DEPRECATED) (env: LLAMA_ARG_DEFRAG_THOLD)
`--rpc SERVERS`	comma-separated list of RPC servers (host:port) (env: LLAMA_ARG_RPC)
`--mlock`	force system to keep model in RAM rather than swapping or compressing (env: LLAMA_ARG_MLOCK)

Figure 6.1 : Extract of some options for hosting LLMs with Llama.cpp

6.3 LLMS AND NUMERIC VALUES

By definition, LLMs are really good with NLP. As seen in Chapters 4 and 5, LLMs seem to struggle when provided raw numerical data from IMU-based sensors and we believe that it would be the case with most numerical-based data. This is shown when performance increases once we remove the gyroscope data in Chapter 5 and further validated by our experiments with dimensionality reduction. This aligns with our understanding as NLP is very distinct from strict numerical-based inputs. It is like comparing how processors communicate versus how humans communicate, it involves thinking differently and a certain level of translation.

6.3.1 NUMERICAL TO NATURAL LANGUAGE

Given that LLMs are better at processing natural language over numerical values, it is logical to assume that converting the raw inputs of the IMU and other sensors into text would improve performance. There has been recent work which has contributed to this approach by using encoding/decoding methods with wearable sensor data in order to communicate via natural language with a LLM [8]. Although, this approach also consists of combining multiple wearable sensors with a video component and a training component is also necessary.

In the summer of 2025, Google released their paper *SensorLM* [73] which greatly contributes to this approach and specifically uses IMU-based data from Fitbit and Google Pixel Watch devices. In this work, they achieved impressive results and a significant leap forward for HAR tasks using IMU data. But, they had to curate a dataset by adding captions for the IMU data

in three categories: Semantic, Statistical and Structural. They also had access to over 59.7 million hours of data from over 103 000 participants.

Although we did consider introducing a certain level of numerical to text translation to communicate our data with the models, we realized, and these works further demonstrate, that this approach negates the benefits and assumptions that LLMs are capable of performing HAR tasks without heavily pre-processing data, combining data sources, or needing to train/fine-tune models in order achieve acceptable performance.

6.3.2 MULTI-MODAL APPROACH

Another approach includes providing context alongside the IMU data to help the LLM better comprehend the big-picture element. This is a relatively recent approach and involves aligning the IMU data with other sources, such as video, audio, text, etc. as a multi-modal approach for HAR tasks [36]. Research using this multi-modal approach with LLMs for HAR tasks shows promising results [21] [12].

Providing context in the prompts was discussed by the LIARA team. It would be interesting to combine the information from other sensors, such as the location of the participant using UWB-radar or even simply the time of day. Unfortunately, this approach also negates the hypothesized benefits of LLMs being effective simple, zero-shot human activity recognizers. It also has the possibility to increase the cost or resources necessary for HAR tasks since more tokens are required to pass the supplementary information.

6.4 DATASETS SEEN BY THE LLMS

A point of concern is that the public datasets used in other works have been seen by the various LLMs when they were trained. This would mean that they perform well because they memorized the labels for the data and that they do not actually interpret it in order to determine the activity being performed. If this is the case, it would explain the good results in a zero-shot experiment context. However, would it really be zero-shot if the data was seen in training?

6.5 ACTIVITY VARIATIONS

We observe noticeable data variations in the various daily living activities of our dataset. These fall under two categories: inter- and intra-individual variations. The former is variations between different participants in how they performed an activity while the latter covers variations in how an individual performed the same activity on two different occasions. An example of inter-individual variation, Figures 6.2 and 6.3 both display the acceleration data for a participant brushing their teeth. Yet, the data is very different between them as they probably brush their teeth in a different manner. In Figures 6.4 and 6.5, we can see intra-individual variation as both Figures show the acceleration data for a same participant drinking on two different occasions. We observe the movement of *drinking* to be more elongated in Figure 6.5 versus Figure 6.4.

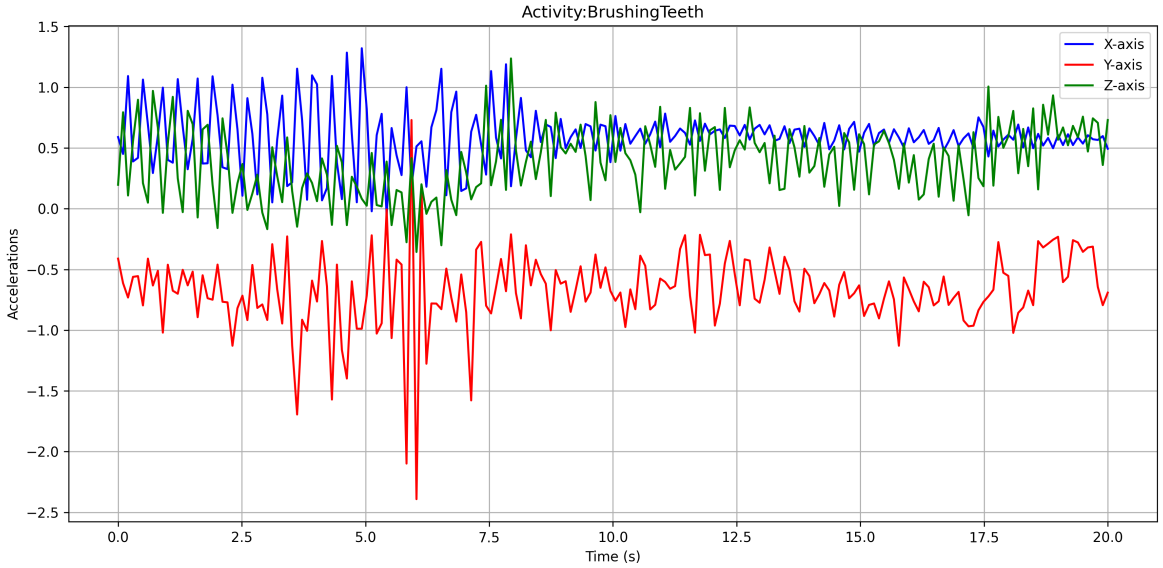

Figure 6.2 : Inter-Variability: Participant 28260ae9 Brushing their teeth

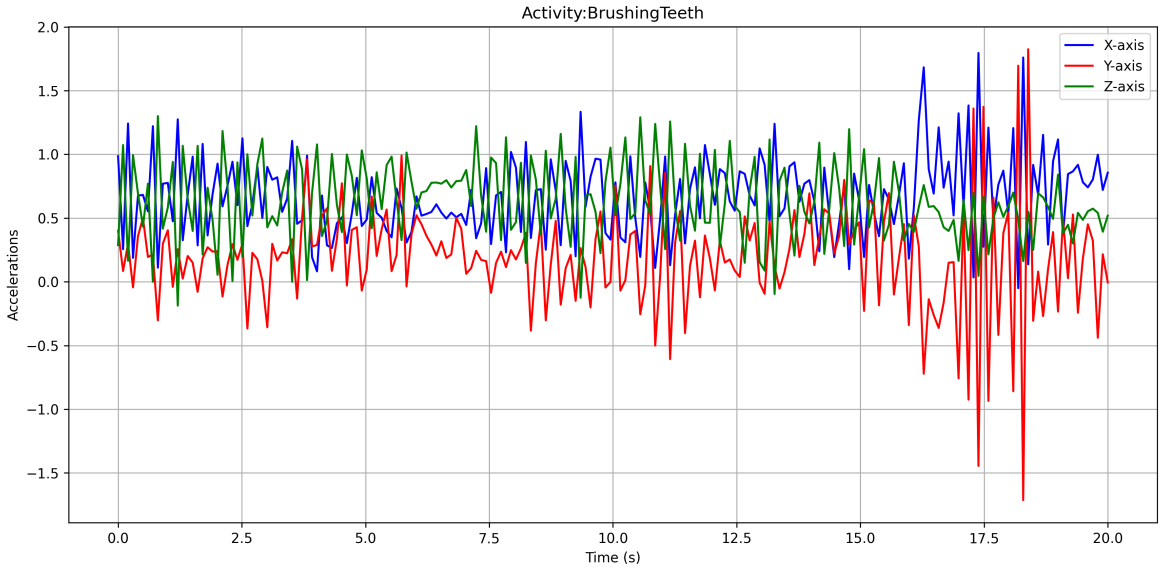

Figure 6.3 : Inter-Variability: Participant da6d42da Brushing their teeth

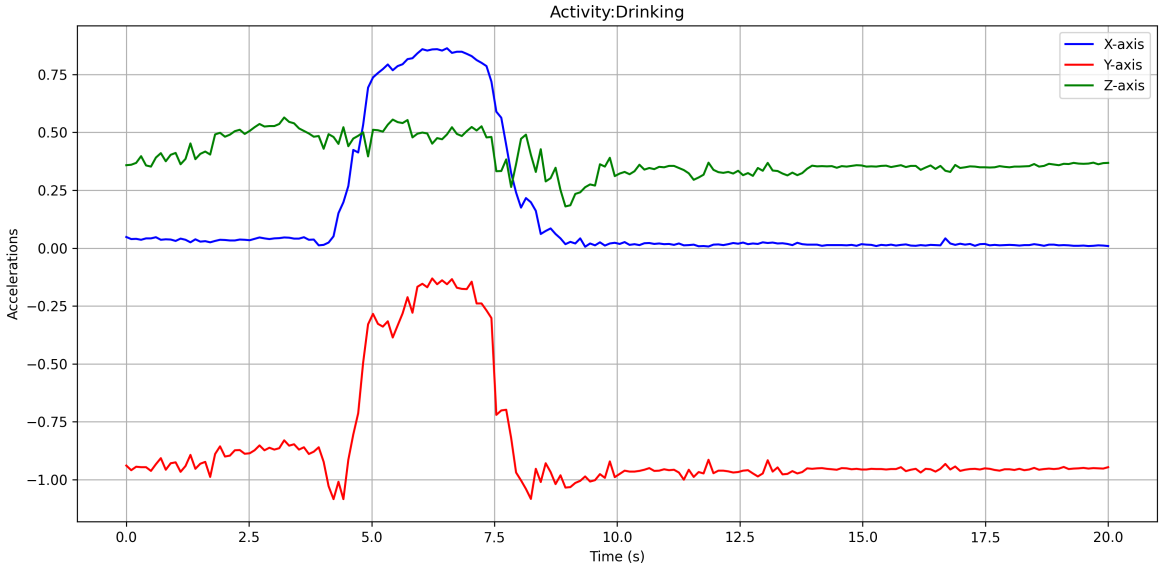

Figure 6.4 : Intra-Variability: Participant 41b00638 Drinking

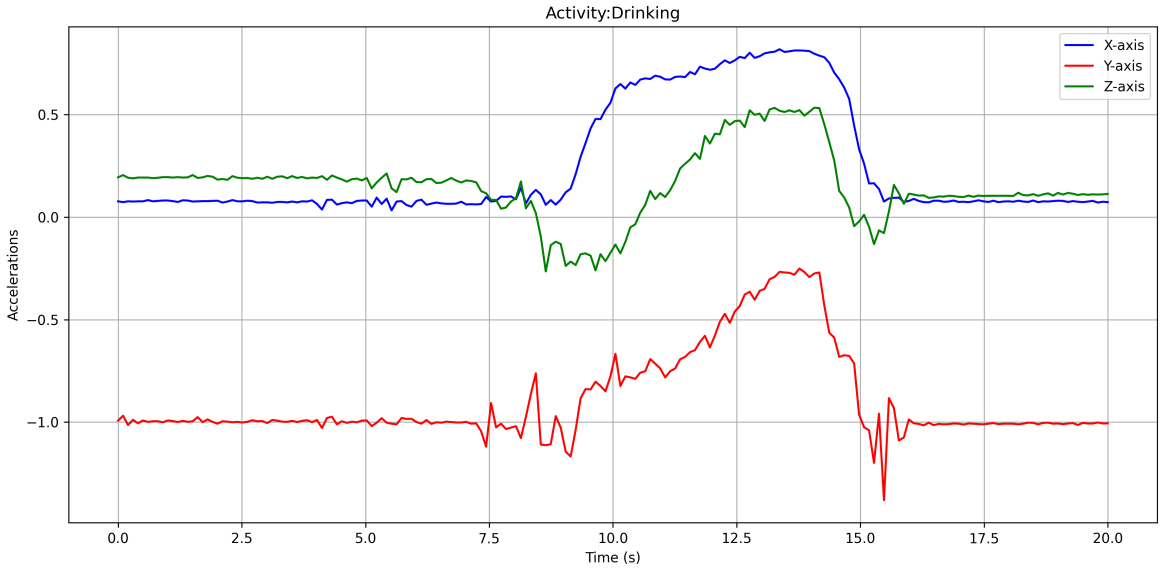

Figure 6.5 : Intra-Variability: Participant 41b00638 Drinking - 2

This variability is important as it serves to explain the inconsistencies in the models ability recognize activities between participants and different instances of an activity for the same participant. It serves to highlight how each individual performs the same daily living activities in their own manner which can be quite different to how the models perceive the activity. Some research projects have worked to tackle this with techniques to make the activity data more heterogeneous [27] [33]. This again requires further data pre-processing before it can be used for HAR tasks. Furthermore, given an individuals state (e.g. tiredness), they can perform the same activity differently. This can be further compounded by location, tools (e.g. fork vs chopsticks), physical factors (e.g. injury), etc. The result is that there is no standard, data-wise, to how a daily living activity is represented due to many combining factors.

6.6 PROMPT STRUCTURE AND LABEL BIAS

A limitation of our research is that we used the same base prompt structure and approach throughout our experiments. Recent works have demonstrated that the prompt structure can trigger internal associations within the LLM which will greatly affect the internal reasoning and subsequent response [4]. This means that our prompt could trigger the models to *think* in a certain manner that is not beneficial for our use case or dataset. Furthermore, it has also been demonstrated that model improvements (e.g. GPT-3 to GPT-4) could bring differences in model response for a same prompt, and not always in a beneficial way [4].

Additionally, within our prompt, we provided the list of possible labels to the model in the order in which they appear in the dataset. Yet, it has been demonstrated in recent works that the order of the labels in the prompt can create biases. This means that the model could

prioritize or *prefer* a certain label given its position in the list (e.g. first or last) [61] [4]. Since we have not experimented with the order of our labels, this could of had an effect on the results we obtained, especially since we changed the selected activities between experiments.

CHAPITRE VII

CONCLUSION

The ageing of the world's population is a major concern for the world's governments, especially in Western societies such as Canada that are already facing the impacts from this significant change in demographics. As this phenomenon increases the strain on health services and resources, human activity recognition has demonstrated itself to be a useful tool for healthy ageing solutions [35] [15]. Deep-learning models have demonstrated good performance on HAR tasks but show limitations when confronted with the generalization necessary of a real-world smart habitat. In contrast, the pre-trained reasoning of large language models have shown great potential in being effective generalizing zero-shot human activity recognizers in early works. Unfortunately, our experiments have shown the limitations of these models as zero-shot recognizers when confronted with with real-world data from an unseen dataset. This limitation is even more prevalent when facing more than a dozen possible activities.

7.1 CONTRIBUTIONS

The first contribution of this master's project is the evaluation of the LIARA data set for zero-shot HAR tasks using LLMs. As demonstrated in chapters four (4), our dataset does not

initially perform well when using a methodology similar to the HARGPT paper to perform HAR tasks on GPT-4o and Meta Llama 3. In the following chapter five (5), we demonstrated that by removing the gyroscope data, as is the case in the data-set CAPTURE24, and making a few modifications to our activities, we managed to attain more acceptable performances. The difference in performance, as discussed in the above section, is hypothesized to be caused by the fact that the datasets used in previous works are public. This means that they are seen by the models when trained. In our case, our laboratories dataset is not public, and therefore unseen by the models.

The second contribution of this master's project is the rigorous analysis of various optimization techniques to increase HAR tasks performance with LLMs. As demonstrated in the fifth (5) chapter of this master's thesis, the performance of LLMs improves significantly when selecting the "correct" activities. This implies that the activity must be clearly defined as vague activities, such as taking a shower, possesses significant inter-variability. Furthermore, we have successfully demonstrated that LLMs do not perform well with raw numerical data. This was seen in the experiments using features instead of the sensor data, especially with the removal of the gyroscope data, the biggest gain in model performance. Finally, we also demonstrated how simply "using a better model" yields significant gains in performance, reflecting the early age of LLMs and the various approaches demonstrated in the early works. This comes with the fact that we must not forget to rigorously compare their results to well established approaches, such as deep learning and even traditional machine learning methods.

7.2 FUTURE WORK

As the latest works presented in the discussion have shown, the new direction for using LLMs in the context of HAR focuses on a multi-modal approach. This approach includes including captions to the datasets to leverage the natural language capabilities of LLMs. It also includes the combining of data from multiple sensors, such as IMU, UWB-raders, video cameras, etc. in an effort to bring more information to the model and help it better determine the activity that the participant is performing. In that context, we believe it would be interesting to apply these methods to our laboratory dataset as we also have data from different sensors, such as UWB-radars, for the activities performed by the participants. We also hypothesize that we could get better performance from LLMs by including textual context to the model from passive sources of information, such as the time of day and the location (room) in which the sensor is in. Integrating these constraints is hypothesized to improve performance by narrowing the LLMs reasoning context, such as excluding labels that are not possible given the parameters.

7.3 PERSONAL NOTE

Although we started to get close to the performance that was observed in the previous work mentioned in the literature review, we must take into consideration that it is on only four (4) very data-distinct activities. Humans perform dozens of distinct activities each day. Our dataset has fourteen (14) and many activities are grouped into the same category (e.g. *Housework*). If we want to include sports as Google did in their *SensorLM* paper, that list grows even longer. Even Google with all their expertise and work achieved an F1-score of 0.29 [73] on a

test set of twenty (20) distinct activities for zero-shot recognition. Furthermore, this is after curating an IMU data-set with captions for the different data in order to communicate via natural language with the models. This aligns with our final experiment in which we included all of our activities and managed an F1-score of 0.15. And we have not even tackled the reality of varying environments when considering smart homes.

REFERENCES

[1] T. R. Bennett, J. Wu, N. Kehtarnavaz, and R. Jafari, "Inertial measurement unit-based wearable computers for assisted living applications: A signal processing perspective," *IEEE Signal Processing Magazine*, vol. 33, no. 2, pp. 28–35, 2016.

[2] B. S. P. T. K. M. Blunck, Henrik and A. Dey, "Heterogeneity Activity Recognition," UCI Machine Learning Repository, 2015, DOI: https://doi.org/10.24432/C5689X.

[3] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, ser. NIPS '20. Red Hook, NY, USA: Curran Associates Inc., 2020.

[4] M. Brucks and O. Toubia, "Prompt architecture induces methodological artifacts in large language models," *PLOS ONE*, vol. 20, no. 4, pp. 1–13, 04 2025. [Online]. Available: https://doi.org/10.1371/journal.pone.0319159

[5] K. M. Caramancion, "Large language models vs. search engines: Evaluating user preferences across varied information retrieval scenarios," *arXiv preprint arXiv:2401.05761*, 2024. [Online]. Available: https://arxiv.org/abs/2401.05761

[6] S. Chan, Y. Hang, C. Tong, A. Acquah, A. Schonfeldt, J. Gershuny, and A. Doherty, "Capture-24: A large dataset of wrist-worn activity tracker data collected in the wild for human activity recognition," *Scientific Data*, vol. 11, no. 1, p. 1135, Oct. 2024.

[7] K. Chapron, F. Thullier, P. Lapointe, J. Maître, K. Bouchard, and S. Gaboury, "Lipshok: Liara portable smart home kit," *Sensors*, vol. 22, no. 8, 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/8/2829

[8] W. Chen, J. Cheng, L. Wang, W. Zhao, and W. Matusik, "Sensor2text: Enabling natural language interactions for daily activity tracking using wearable sensors," *Proceedings of*

the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, vol. 8, no. 4, p. 1–26, Nov. 2024.

[9] X. Chen, R. A. Chi, X. Wang, and D. Zhou, "Premise order matters in reasoning with large language models," in *Proceedings of the 41st International Conference on Machine Learning*, ser. ICML'24. JMLR.org, 2024.

[10] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, and N. Fiedel, "Palm: scaling language modeling with pathways," *J. Mach. Learn. Res.*, vol. 24, no. 1, Jan. 2023.

[11] G. Civitarese, M. Fiori, P. Choudhary, and C. Bettini, "Large language models are zero-shot recognizers for activities of daily living," *arXiv preprint arXiv:2407.01238*, 2024. [Online]. Available: https://arxiv.org/abs/2407.01238

[12] I. Demirel, K. Thakkar, B. Elizalde, M. E. Marques, A. Sarathy, Y. Bai, U. Srinivas, J. Xu, S. Ren, and J. Narain, "Using llms for late multimodal sensor fusion for activity recognition," 2025. [Online]. Available: https://arxiv.org/abs/2509.10729

[13] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018. [Online]. Available: https://arxiv.org/abs/1810.04805

[14] V. Farrahi, U. Muhammad, M. Rostami, and M. Oussalah, "Accnet24: A deep learning framework for classifying 24-hour activity behaviours from wrist-worn accelerometer data under free-living environments," *International Journal of Medical Informatics*, vol. 172, Apr. 2023.

[15] R.-P. Filiou, M. Couture, M. Lussier, A. Aboujaoudé, G. Paré, S. Giroux, H. Kenfack Ngankam, P. Belchior, C. Bottari, K. Bouchard, S. Gaboury, C. Gouin-Vallerand, F. A. Etindele Sosso, and N. Bier, "Decision-making process of home and social care professionals using telemonitoring of activities of daily living for risk assessment: Embedded mixed methods multiple-case study," *J Med Internet Res*, vol. 27, Apr 2025. [Online]. Available: https://www.jmir.org/2025/1/e64713

[16] G. Gerganov and contributors, "llama.cpp: Llm inference in c/c++," https://github.com/ggml-org/llama.cpp, 2023.

[17] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, A. Yang, A. Fan, A. Goyal, A. Hartshorn, A. Yang, A. Mitra, A. Sravankumar, A. Korenev, A. Hinsvark, A. Rao, A. Zhang, A. Rodriguez, A. Gregerson, A. Spataru, B. Roziere, B. Biron, B. Tang, B. Chern, C. Caucheteux, C. Nayak, C. Bi, C. Marra, C. McConnell, C. Keller, C. Touret, C. Wu, C. Wong, C. C. Ferrer, C. Nikolaidis, D. Allonsius, D. Song, D. Pintz, D. Livshits, D. Wyatt, D. Esiobu, D. Choudhary, D. Mahajan, D. Garcia-Olano, D. Perino, D. Hupkes, E. Lakomkin, E. AlBadawy, E. Lobanova, E. Dinan, E. M. Smith, F. Radenovic, F. Guzmán, F. Zhang, G. Synnaeve, G. Lee, G. L. Anderson, G. Thattai, G. Nail, G. Mialon, G. Pang, G. Cucurell, H. Nguyen, H. Korevaar, H. Xu, H. Touvron, I. Zarov, I. A. Ibarra, I. Kloumann, I. Misra, I. Evtimov, J. Zhang, J. Copet, J. Lee, J. Geffert, J. Vranes, J. Park, J. Mahadeokar, J. Shah, J. van der Linde, J. Billock, J. Hong, J. Lee, J. Fu, J. Chi, J. Huang, J. Liu, J. Wang, J. Yu, J. Bitton, J. Spisak, J. Park, J. Rocca, J. Johnstun, J. Saxe, J. Jia, K. V. Alwala, K. Prasad, K. Upasani, K. Plawiak, K. Li, K. Heafield, K. Stone, K. El-Arini, K. Iyer, K. Malik, K. Chiu, K. Bhalla, K. Lakhotia, L. Rantala-Yeary, L. van der Maaten, L. Chen, L. Tan, L. Jenkins, L. Martin, L. Madaan, L. Malo, L. Blecher, L. Landzaat, L. de Oliveira, M. Muzzi, M. Pasupuleti, M. Singh, M. Paluri, M. Kardas, M. Tsimpoukelli, M. Oldham, M. Rita, M. Pavlova, M. Kambadur, M. Lewis, M. Si, M. K. Singh, M. Hassan, N. Goyal, N. Torabi, N. Bashlykov, N. Bogoychev, N. Chatterji, N. Zhang, O. Duchenne, O. Çelebi, P. Alrassy, P. Zhang, P. Li, P. Vasic, P. Weng, P. Bhargava, P. Dubal, P. Krishnan, P. S. Koura, P. Xu, Q. He, Q. Dong, R. Srinivasan, R. Ganapathy, R. Calderer, R. S. Cabral, R. Stojnic, R. Raileanu, R. Maheswari, R. Girdhar, R. Patel, R. Sauvestre, R. Polidoro, R. Sumbaly, R. Taylor, R. Silva, R. Hou, R. Wang, S. Hosseini, S. Chennabasappa, S. Singh, S. Bell, S. S. Kim, S. Edunov, S. Nie, S. Narang, S. Raparthy, S. Shen, S. Wan, S. Bhosale, S. Zhang, S. Vandenhende, S. Batra, S. Whitman, S. Sootla, S. Collot, S. Gururangan, S. Borodinsky, T. Herman, T. Fowler, T. Sheasha, T. Georgiou, T. Scialom, T. Speckbacher, T. Mihaylov, T. Xiao, U. Karn, V. Goswami, V. Gupta, V. Ramanathan, V. Kerkez, V. Gonguet, V. Do, V. Vogeti, V. Albiero, V. Petrovic, W. Chu, W. Xiong, W. Fu, W. Meers, X. Martinet, X. Wang, X. Wang, X. E. Tan, X. Xia, X. Xie, X. Jia, X. Wang, Y. Goldschlag, Y. Gaur, Y. Babaei, Y. Wen, Y. Song, Y. Zhang, Y. Li, Y. Mao, Z. D. Coudert, Z. Yan, Z. Chen, Z. Papakipos, A. Singh, A. Srivastava, A. Jain, A. Kelsey, A. Shajnfeld, A. Gangidi, A. Victoria, A. Goldstand, A. Menon, A. Sharma, A. Boesenberg, A. Baevski, A. Feinstein, A. Kallet, A. Sangani, A. Teo, A. Yunus, A. Lupu, A. Alvarado, A. Caples, A. Gu, A. Ho, A. Poulton, A. Ryan, A. Ramchandani, A. Dong, A. Franco, A. Goyal, A. Saraf, A. Chowdhury, A. Gabriel, A. Bharambe, A. Eisenman, A. Yazdan, B. James, B. Maurer, B. Leonhardi, B. Huang, B. Loyd, B. D. Paola, B. Paranjape, B. Liu, B. Wu, B. Ni, B. Hancock, B. Wasti, B. Spence, B. Stojkovic, B. Gamido, B. Montalvo, C. Parker, C. Burton, C. Mejia, C. Liu, C. Wang, C. Kim, C. Zhou, C. Hu, C.-H. Chu, C. Cai, C. Tindal, C. Feichtenhofer, C. Gao, D. Civin, D. Beaty, D. Kreymer, D. Li, D. Adkins, D. Xu, D. Testuggine, D. David, D. Parikh, D. Liskovich, D. Foss, D. Wang, D. Le, D. Holland, E. Dowling,

E. Jamil, E. Montgomery, E. Presani, E. Hahn, E. Wood, E.-T. Le, E. Brinkman, E. Arcaute, E. Dunbar, E. Smothers, F. Sun, F. Kreuk, F. Tian, F. Kokkinos, F. Ozgenel, F. Caggioni, F. Kanayet, F. Seide, G. M. Florez, G. Schwarz, G. Badeer, G. Swee, G. Halpern, G. Herman, G. Sizov, Guangyi, Zhang, G. Lakshminarayanan, H. Inan, H. Shojanazeri, H. Zou, H. Wang, H. Zha, H. Habeeb, H. Rudolph, H. Suk, H. Aspegren, H. Goldman, H. Zhan, I. Damlaj, I. Molybog, I. Tufanov, I. Leontiadis, I.-E. Veliche, I. Gat, J. Weissman, J. Geboski, J. Kohli, J. Lam, J. Asher, J.-B. Gaya, J. Marcus, J. Tang, J. Chan, J. Zhen, J. Reizenstein, J. Teboul, J. Zhong, J. Jin, J. Yang, J. Cummings, J. Carvill, J. Shepard, J. McPhie, J. Torres, J. Ginsburg, J. Wang, K. Wu, K. H. U, K. Saxena, K. Khandelwal, K. Zand, K. Matosich, K. Veeraraghavan, K. Michelena, K. Li, K. Jagadeesh, K. Huang, K. Chawla, K. Huang, L. Chen, L. Garg, L. A, L. Silva, L. Bell, L. Zhang, L. Guo, L. Yu, L. Moshkovich, L. Wehrstedt, M. Khabsa, M. Avalani, M. Bhatt, M. Mankus, M. Hasson, M. Lennie, M. Reso, M. Groshev, M. Naumov, M. Lathi, M. Keneally, M. Liu, M. L. Seltzer, M. Valko, M. Restrepo, M. Patel, M. Vyatskov, M. Samvelyan, M. Clark, M. Macey, M. Wang, M. J. Hermoso, M. Metanat, M. Rastegari, M. Bansal, N. Santhanam, N. Parks, N. White, N. Bawa, N. Singhal, N. Egebo, N. Usunier, N. Mehta, N. P. Laptev, N. Dong, N. Cheng, O. Chernoguz, O. Hart, O. Salpekar, O. Kalinli, P. Kent, P. Parekh, P. Saab, P. Balaji, P. Rittner, P. Bontrager, P. Roux, P. Dollar, P. Zvyagina, P. Ratanchandani, P. Yuvraj, Q. Liang, R. Alao, R. Rodriguez, R. Ayub, R. Murthy, R. Nayani, R. Mitra, R. Parthasarathy, R. Li, R. Hogan, R. Battey, R. Wang, R. Howes, R. Rinott, S. Mehta, S. Siby, S. J. Bondu, S. Datta, S. Chugh, S. Hunt, S. Dhillon, S. Sidorov, S. Pan, S. Mahajan, S. Verma, S. Yamamoto, S. Ramaswamy, S. Lindsay, S. Lindsay, S. Feng, S. Lin, S. C. Zha, S. Patil, S. Shankar, S. Zhang, S. Zhang, S. Wang, S. Agarwal, S. Sajuyigbe, S. Chintala, S. Max, S. Chen, S. Kehoe, S. Satterfield, S. Govindaprasad, S. Gupta, S. Deng, S. Cho, S. Virk, S. Subramanian, S. Choudhury, S. Goldman, T. Remez, T. Glaser, T. Best, T. Koehler, T. Robinson, T. Li, T. Zhang, T. Matthews, T. Chou, T. Shaked, V. Vontimitta, V. Ajayi, V. Montanez, V. Mohan, V. S. Kumar, V. Mangla, V. Ionescu, V. Poenaru, V. T. Mihailescu, V. Ivanov, W. Li, W. Wang, W. Jiang, W. Bouaziz, W. Constable, X. Tang, X. Wu, X. Wang, X. Wu, X. Gao, Y. Kleinman, Y. Chen, Y. Hu, Y. Jia, Y. Qi, Y. Li, Y. Zhang, Y. Zhang, Y. Adi, Y. Nam, Yu, Wang, Y. Zhao, Y. Hao, Y. Qian, Y. Li, Y. He, Z. Rait, Z. DeVito, Z. Rosnbrick, Z. Wen, Z. Yang, Z. Zhao, and Z. Ma, "The llama 3 herd of models," 2024. [Online]. Available: https://arxiv.org/abs/2407.21783

[18] S. Gupta, "Deep learning based human activity recognition (har) using wearable sensor data," *International Journal of Information Management Data Insights*, vol. 1, no. 2, p. 100046, Nov. 2021.

[19] H. Haresamudram, H. Rajasekhar, N. M. Shanbhogue, and T. Ploetz, "Large language models memorize sensor datasets! implications on human activity recognition research," no. arXiv:2406.05900, 2024, arXiv:2406.05900 [cs]. [Online]. Available: http://arxiv.org/abs/2406.05900

[20] Z. Hong, Y. Song, Z. Li, A. Yu, S. Zhong, Y. Ding, T. He, and D. Zhang, "Llm4har: Generalizable on-device human activity recognition with pretrained llms," in *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, ser. KDD '25. New York, NY, USA: Association for Computing Machinery, 2025, p. 4511–4521. [Online]. Available: https://doi.org/10.1145/3711896.3737226

[21] S. A. Imran, M. N. H. Khan, S. Biswas, and B. Islam, "Llasa: A multimodal llm for human activity analysis through wearable and smartphone sensors," 2025.

[22] Institut de la statistique du Québec, "Portrait des personnes aînées au québec," https://statistique.quebec.ca/fr/fichier/portrait-personnes-ainees-quebec.pdf, 2023, accessed: 2026-04-06.

[23] ——, "Portrait des personnes aînées au québec," Gouvernement du Québec, Québec, Rapport, 2023. [Online]. Available: https://statistique.quebec.ca/fr/fichier/portrait-personnes-ainees-quebec.pdf

[24] S. Ji, X. Zheng, and C. Wu, "Hargpt: Are llms zero-shot human activity recognizers?" in *2024 IEEE International Workshop on Foundation Models for Cyber-Physical Systems Internet of Things (FMSys)*, 2024, pp. 38–43.

[25] C. Jobanputra, J. Bavishi, and N. Doshi, "Human activity recognition: A survey," *Procedia Computer Science*, vol. 155, p. 698–703, 2019.

[26] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa, "Large language models are zero-shot reasoners," in *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., vol. 35. Curran Associates, Inc., 2022, pp. 22 199–22 213. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2022/file/8bb0d291acd4acf06ef112099c16f326-Paper-Conference.pdf

[27] P. Kumar and S. Suresh, "Deeptranshhar: Inter-subjects heterogeneous activity recognition approach in the non-identical environment using wearable sensors," *National Academy Science Letters*, vol. 45, no. 4, p. 317–323, Aug. 2022.

[28] P. Kumar, S. Chauhan, and L. K. Awasthi, "Human activity recognition (har) using deep learning: Review, methodologies, progress and future research directions," *Archives of Computational Methods in Engineering*, vol. 31, no. 1, p. 179–219, Jan. 2024.

[29] F. D. la Torre, J. K. Hodgins, A. W. Bargteil, X. Martin, J. R. Macey, A. T. Collado, and P. Beltran, "Guide to the carnegie mellon university multimodal activity (cmu-mmac) database," 2008. [Online]. Available: https://api.semanticscholar.org/CorpusID:16721121

[30] Z. Leng, A. Bhattacharjee, H. Rajasekhar, L. Zhang, E. Bruda, H. Kwon, and T. Plötz, "Imugpt 2.0: Language-based cross modality transfer for sensor-based human activity recognition," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 8, no. 3, p. 1–32, Aug. 2024.

[31] Y. Li, Y. Wu, J. Li, and S. Liu, "Prompting large language models for zero-shot domain adaptation in speech recognition," 12 2023, pp. 1–8.

[32] L. Liu, J. Meng, and Y. Yang, "Llm technologies and information search," *Journal of Economyand Technology*, vol. 2, pp. 269–277, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2949948824000398

[33] W. Lu, J. Wang, Y. Chen, S. J. Pan, C. Hu, and X. Qin, "Semantic-discriminative mixup for generalizable sensor-based cross-domain activity recognition," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 6, no. 2, p. 1–19, 2022.

[34] J. Maitre, K. Bouchard, C. Bertuglia, and S. Gaboury, "Recognizing activities of daily living from UWB radars and deep learning," *Expert Systems with Applications*, vol. 164, p. 113994, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417420307703

[35] J. Maitre, K. Bouchard, and S. Gaboury, "Fall detection with uwb radars and cnn-lstm architecture," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 4, p. 1273–1283, Apr. 2021.

[36] S. Moon, A. Madotto, Z. Lin, A. Dirafzoon, A. Saraf, A. Bearman, and B. Damavandi, "Imu2clip: Multimodal contrastive learning for imu motion sensors from egocentric videos and text narrations."

[37] A. Neumann, E. Kirsten, M. B. Zafar, and J. Singh, "Position is power: System prompts as a mechanism of bias in large language models (llms)," in *Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency*, ser. FAccT '25. New York, NY, USA: Association for Computing Machinery, 2025, p. 573–598. [Online]. Available: https://doi.org/10.1145/3715275.3732038

[38] Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A time series is worth 64 words: Long-term forecasting with transformers," *ArXiv*, vol. abs/2211.14730, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:254044221

[39] OpenAI, J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, R. Avila, I. Babuschkin, S. Balaji, V. Balcom, P. Baltescu, H. Bao, M. Bavarian, J. Belgum, I. Bello, J. Berdine, G. Bernadett-Shapiro, C. Berner, L. Bogdonoff, O. Boiko, M. Boyd, A.-L. Brakman, G. Brockman, T. Brooks, M. Brundage, K. Button, T. Cai, R. Campbell, A. Cann,

B. Carey, C. Carlson, R. Carmichael, B. Chan, C. Chang, F. Chantzis, D. Chen, S. Chen, R. Chen, J. Chen, M. Chen, B. Chess, C. Cho, C. Chu, H. W. Chung, D. Cummings, J. Currier, Y. Dai, C. Decareaux, T. Degry, N. Deutsch, D. Deville, A. Dhar, D. Dohan, S. Dowling, S. Dunning, A. Ecoffet, A. Eleti, T. Eloundou, D. Farhi, L. Fedus, N. Felix, S. P. Fishman, J. Forte, I. Fulford, L. Gao, E. Georges, C. Gibson, V. Goel, T. Gogineni, G. Goh, R. Gontijo-Lopes, J. Gordon, M. Grafstein, S. Gray, R. Greene, J. Gross, S. S. Gu, Y. Guo, C. Hallacy, J. Han, J. Harris, Y. He, M. Heaton, J. Heidecke, C. Hesse, A. Hickey, W. Hickey, P. Hoeschele, B. Houghton, K. Hsu, S. Hu, X. Hu, J. Huizinga, S. Jain, S. Jain, J. Jang, A. Jiang, R. Jiang, H. Jin, D. Jin, S. Jomoto, B. Jonn, H. Jun, T. Kaftan, Kaiser, A. Kamali, I. Kanitscheider, N. S. Keskar, T. Khan, L. Kilpatrick, J. W. Kim, C. Kim, Y. Kim, J. H. Kirchner, J. Kiros, M. Knight, D. Kokotajlo, Kondraciuk, A. Kondrich, A. Konstantinidis, K. Kosic, G. Krueger, V. Kuo, M. Lampe, I. Lan, T. Lee, J. Leike, J. Leung, D. Levy, C. M. Li, R. Lim, M. Lin, S. Lin, M. Litwin, T. Lopez, R. Lowe, P. Lue, A. Makanju, K. Malfacini, S. Manning, T. Markov, Y. Markovski, B. Martin, K. Mayer, A. Mayne, B. McGrew, S. M. McKinney, C. McLeavey, P. McMillan, J. McNeil, D. Medina, A. Mehta, J. Menick, L. Metz, A. Mishchenko, P. Mishkin, V. Monaco, E. Morikawa, D. Mossing, T. Mu, M. Murati, O. Murk, D. Mély, A. Nair, R. Nakano, R. Nayak, A. Neelakantan, R. Ngo, H. Noh, L. Ouyang, C. O'Keefe, J. Pachocki, A. Paino, J. Palermo, A. Pantuliano, G. Parascandolo, J. Parish, E. Parparita, A. Passos, M. Pavlov, A. Peng, A. Perelman, F. d. A. B. Peres, M. Petrov, H. P. d. O. Pinto, Michael, Pokorny, M. Pokrass, V. H. Pong, T. Powell, A. Power, B. Power, E. Proehl, R. Puri, A. Radford, J. Rae, A. Ramesh, C. Raymond, F. Real, K. Rimbach, C. Ross, B. Rotsted, H. Roussez, N. Ryder, M. Saltarelli, T. Sanders, S. Santurkar, G. Sastry, H. Schmidt, D. Schnurr, J. Schulman, D. Selsam, K. Sheppard, T. Sherbakov, J. Shieh, S. Shoker, P. Shyam, S. Sidor, E. Sigler, M. Simens, J. Sitkin, K. Slama, I. Sohl, B. Sokolowsky, Y. Song, N. Staudacher, F. P. Such, N. Summers, I. Sutskever, J. Tang, N. Tezak, M. B. Thompson, P. Tillet, A. Tootoonchian, E. Tseng, P. Tuggle, N. Turley, J. Tworek, J. F. C. Uribe, A. Vallone, A. Vijayvergiya, C. Voss, C. Wainwright, J. J. Wang, A. Wang, B. Wang, J. Ward, J. Wei, C. J. Weinmann, A. Welihinda, P. Welinder, J. Weng, L. Weng, M. Wiethoff, D. Willner, C. Winter, S. Wolrich, H. Wong, L. Workman, S. Wu, J. Wu, M. Wu, K. Xiao, T. Xu, S. Yoo, K. Yu, Q. Yuan, W. Zaremba, R. Zellers, C. Zhang, M. Zhang, S. Zhao, T. Zheng, J. Zhuang, W. Zhuk, and B. Zoph, "Gpt-4 technical report," no. arXiv:2303.08774, Mar. 2024, arXiv:2303.08774 [cs.CL]. [Online]. Available: http://arxiv.org/abs/2303.08774

[40] P. K. O'Neill, V. Lavrukhin, S. Majumdar, V. Noroozi, Y. Zhang, O. Kuchaiev, J. Balam, Y. Dovzhenko, K. Freyberg, M. D. Shulman, B. Ginsburg, S. Watanabe, and G. Kucsko, "Spgispeech: 5,000 hours of transcribed financial audio for fully formatted end-to-end speech recognition," no. arXiv:2104.02014, Apr. 2021, arXiv:2104.02014 [cs.CL]. [Online]. Available: http://arxiv.org/abs/2104.02014

[41] H. Pang, L. Zheng, and H. Fang, "Cross-attention enhanced pyramid multi-scale networks for sensor-based human activity recognition," *IEEE Journal of Biomedical and Health*

Informatics, vol. 28, no. 5, pp. 2733–2744, 2024.

[42] D. Park, G.-t. An, C. Kamyod, and C. G. Kim, "A study on performance improvement of prompt engineering for generative ai with a large language model," *Journal of Web Engineering*, vol. 22, no. 8, p. 1187–1206, Nov. 2023.

[43] C. Prince, N. Omrani, A. Maalaoui, M. Dabic, and S. Kraus, "Are we living in surveillance societies and is privacy an illusion? an empirical study on privacy literacy and privacy concerns," *IEEE Transactions on Engineering Management*, vol. 70, no. 10, pp. 3553–3570, 2023.

[44] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:160025533

[45] M. Rani and M. Kumar, "Human activity recognition (har): techniques, applications, and future directions," *International Journal of Data Science and Analytics*, vol. 22, no. 1, p. 73, Dec. 2026.

[46] A. Reiss and D. Stricker, "Introducing a new benchmarked dataset for activity monitoring," pp. 108–109, 2012.

[47] J.-L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, and D. Anguita, "Transition-aware human activity recognition using smartphones," *Neurocomput.*, vol. 171, no. C, p. 754–767, Jan. 2016. [Online]. Available: https://doi.org/10.1016/j.neucom.2015.07.085

[48] A. Robaczewski, J. Bouchard, K. Bouchard, and S. Gaboury, "Socially assistive robots: The specific case of the nao," *International Journal of Social Robotics*, vol. 13, no. 4, p. 795–831, 2021.

[49] D. Roggen, A. Calatroni, M. Rossi, T. Holleczek, K. Förster, G. Tröster, P. Lukowicz, D. Bannach, G. Pirkl, A. Ferscha, J. Doppler, C. Holzmann, M. Kurz, G. Holl, R. Chavarriaga, H. Sagha, H. Bayati, M. Creatura, and J. d. R. Millan, "Collecting complex activity datasets in highly rich networked sensor environments," in *2010 Seventh International Conference on Networked Sensing Systems (INSS)*, 2010, pp. 233–240.

[50] M. E. Rosenberger, J. E. Fulton, M. P. Buman, R. P. Troiano, M. A. Grandner, S. M. Kozey-Keadle, and W. L. Haskell, "The 24-hour activity cycle: A new paradigm for physical activity," *Medicine & Science in Sports & Exercise*, vol. 51, no. 3, pp. 454–464, 2019.

[51] A. Rousseau, P. Deléglise, and Y. Estève, "TED-LIUM: an automatic speech recognition dedicated corpus," in *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, N. Calzolari, K. Choukri, T. Declerck, M. U. Doğan, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, and S. Piperidis, Eds. Istanbul,

Turkey: European Language Resources Association (ELRA), May 2012, pp. 125–129. [Online]. Available: https://aclanthology.org/L12-1405/

[52] U. Saha, S. Saha, M. T. Kabir, S. A. Fattah, and M. Saquib, "Decoding human activities: Analyzing wearable accelerometer and gyroscope data for activity recognition," *IEEE Sensors Letters*, vol. 8, no. 8, pp. 1–4, 2024.

[53] M. Shoaib, S. Bosch, O. D. Incel, H. Scholten, and P. J. M. Havinga, "Fusion of smartphone motion sensors for physical activity recognition," *Sensors*, vol. 14, no. 6, pp. 10 146–10 176, 2014. [Online]. Available: https://www.mdpi.com/1424-8220/14/6/10146

[54] A. Singh, A. Fry, A. Perelman, A. Tart, A. Ganesh, A. El-Kishky, A. McLaughlin, A. Low, A. J. Ostrow, A. Ananthram, A. Nathan, A. Luo, A. Helyar, A. Madry, A. Efremov, A. Spyra, A. Baker-Whitcomb, A. Beutel, A. Karpenko, A. Makelov, A. Neitz, A. Wei, A. Barr, A. Kirchmeyer, A. Ivanov, A. Christakis, A. Gillespie, A. Tam, A. Bennett, A. Wan, A. Huang, A. M. Sandjideh, A. Yang, A. Kumar, A. Saraiva, A. Vallone, A. Gheorghe, A. G. Garcia, A. Braunstein, A. Liu, A. Schmidt, A. Mereskin, A. Mishchenko, A. Applebaum, A. Rogerson, A. Rajan, A. Wei, A. Kotha, A. Srivastava, A. Agrawal, A. Vijayvergiya, A. Tyra, A. Nair, A. Nayak, B. Eggers, B. Ji, B. Hoover, B. Chen, B. Chen, B. Barak, B. Minaiev, B. Hao, B. Baker, B. Lightcap, B. McKinzie, B. Wang, B. Quinn, B. Fioca, B. Hsu, B. Yang, B. Yu, B. Zhang, B. Brenner, C. R. Zetino, C. Raymond, C. Lugaresi, C. Paz, C. Hudson, C. Whitney, C. Li, C. Chen, C. Cole, C. Voss, C. Ding, C. Shen, C. Huang, C. Colby, C. Hallacy, C. Koch, C. Lu, C. Kaplan, C. Kim, C. J. Minott-Henriques, C. Frey, C. Yu, C. Czarnecki, C. Reid, C. Wei, C. Decareaux, C. Scheau, C. Zhang, C. Forbes, D. Tang, D. Goldberg, D. Roberts, D. Palmie, D. Kappler, D. Levine, D. Wright, D. Leo, D. Lin, D. Robinson, D. Grabb, D. Chen, D. Lim, D. Salama, D. Bhattacharjee, D. Tsipras, D. Li, D. Yu, D. J. Strouse, D. Williams, D. Hunn, E. Bayes, E. Arbus, E. Akyurek, E. Y. Le, E. Widmann, E. Yani, E. Proehl, E. Sert, E. Cheung, E. Schwartz, E. Han, E. Jiang, E. Mitchell, E. Sigler, E. Wallace, E. Ritter, E. Kavanaugh, E. Mays, E. Nikishin, F. Li, F. P. Such, F. d. A. B. Peres, F. Raso, F. Bekerman, F. Tsimpourlas, F. Chantzis, F. Song, F. Zhang, G. Raila, G. McGrath, G. Briggs, G. Yang, G. Parascandolo, G. Chabot, G. Kim, G. Zhao, G. Valiant, G. Leclerc, H. Salman, H. Wang, H. Sheng, H. Jiang, H. Wang, H. Jin, H. Sikchi, H. Schmidt, H. Aspegren, H. Chen, H. Qiu, H. Lightman, I. Covert, I. Kivlichan, I. Silber, I. Sohl, I. Hammoud, I. Clavera, I. Lan, I. Akkaya, I. Kostrikov, I. Kofman, I. Etinger, I. Singal, J. Hehir, J. Huh, J. Pan, J. Wilczynski, J. Pachocki, J. Lee, J. Quinn, J. Kiros, J. Kalra, J. Samaroo, J. Wang, J. Wolfe, J. Chen, J. Wang, J. Harb, J. Han, J. Wang, J. Zhao, J. Chen, J. Yang, J. Tworek, J. Chand, J. Landon, J. Liang, J. Lin, J. Liu, J. Wang, J. Tang, J. Yin, J. Jang, J. Morris, J. Flynn, J. Ferstad, J. Heidecke, J. Fishbein, J. Hallman, J. Grant, J. Chien, J. Gordon, J. Park, J. Liss, J. Kraaijeveld, J. Guay, J. Mo, J. Lawson, J. McGrath, J. Vendrow, J. Jiao, J. Lee, J. Steele, J. Wang, J. Mao, K. Chen, K. Hayashi, K. Xiao, K. Salahi, K. Wu, K. Sekhri, K. Sharma, K. Singhal, K. Li, K. Nguyen, K. Gu-Lemberg, K. King, K. Liu, K. Stone,

K. Yu, K. Ying, K. Georgiev, K. Lim, K. Tirumala, K. Miller, L. Ahmad, L. Lv, L. Clare, L. Fauconnet, L. Itow, L. Yang, L. Romaniuk, L. Anise, L. Byron, L. Pathak, L. Maksin, L. Lo, L. Ho, L. Jing, L. Wu, L. Xiong, L. Mamitsuka, L. Yang, L. McCallum, L. Held, L. Bourgeois, L. Engstrom, L. Kuhn, L. Feuvrier, L. Zhang, L. Switzer, L. Kondraciuk, L. Kaiser, M. Joglekar, M. Singh, M. Shah, M. Stratta, M. Williams, M. Chen, M. Sun, M. Cayton, M. Li, M. Zhang, M. Aljubeh, M. Nichols, M. Haines, M. Schwarzer, M. Gupta, M. Shah, M. Y. Guan, M. Huang, M. Dong, M. Wang, M. Glaese, M. Carroll, M. Lampe, M. Malek, M. Sharman, M. Zhang, M. Wang, M. Pokrass, M. Florian, M. Pavlov, M. Wang, M. Chen, M. Wang, M. Feng, M. Bavarian, M. Lin, M. Abdool, M. Rohaninejad, N. Soto, N. Staudacher, N. LaFontaine, N. Marwell, N. Liu, N. Preston, N. Turley, N. Ansman, N. Blades, N. Pancha, N. Mikhaylin, N. Felix, N. Handa, N. Rai, N. Keskar, N. Brown, O. Nachum, O. Boiko, O. Murk, O. Watkins, O. Gleeson, P. Mishkin, P. Lesiewicz, P. Baltescu, P. Belov, P. Zhokhov, P. Pronin, P. Guo, P. Thacker, Q. Liu, Q. Yuan, Q. Liu, R. Dias, R. Puckett, R. Arora, R. T. Mullapudi, R. Gaon, R. Miyara, R. Song, R. Aggarwal, R. J. Marsan, R. Yemiru, R. Xiong, R. Kshirsagar, R. Nuttall, R. Tsiupa, R. Eldan, R. Wang, R. James, R. Ziv, R. Shu, R. Nigmatullin, S. Jain, S. Talaie, S. Altman, S. Arnesen, S. Toizer, S. Toyer, S. Miserendino, S. Agarwal, S. Yoo, S. Heon, S. Ethersmith, S. Grove, S. Taylor, S. Bubeck, S. Banesiu, S. Amdo, S. Zhao, S. Wu, S. Santurkar, S. Zhao, S. R. Chaudhuri, S. Krishnaswamy, Shuaiqi, Xia, S. Cheng, S. Anadkat, S. P. Fishman, S. Tobin, S. Fu, S. Jain, S. Mei, S. Egoian, S. Kim, S. Golden, S. Q. Mah, S. Lin, S. Imm, S. Sharpe, S. Yadlowsky, S. Choudhry, S. Eum, S. Sanjeev, T. Khan, T. Stramer, T. Wang, T. Xin, T. Gogineni, T. Christianson, T. Sanders, T. Patwardhan, T. Degry, T. Shadwell, T. Fu, T. Gao, T. Garipov, T. Sriskandarajah, T. Sherbakov, T. Korbak, T. Kaftan, T. Hiratsuka, T. Wang, T. Song, T. Zhao, T. Peterson, V. Kharitonov, V. Chernova, V. Kosaraju, V. Kuo, V. Pong, V. Verma, V. Petrov, W. Jiang, W. Zhang, W. Zhou, W. Xie, W. Zhan, W. McCabe, W. DePue, W. Ellsworth, W. Bain, W. Thompson, X. Chen, X. Qi, X. Xiang, X. Shi, Y. Dubois, Y. Yu, Y. Khakbaz, Y. Wu, Y. Qian, Y. T. Lee, Y. Chen, Y. Zhang, Y. Xiong, Y. Tian, Y. Cha, Y. Bai, Y. Yang, Y. Yuan, Y. Li, Y. Zhang, Y. Yang, Y. Jin, Y. Jiang, Y. Wang, Y. Wang, Y. Liu, Z. Stubenvoll, Z. Dou, Z. Wu, and Z. Wang, "Openai gpt-5 system card," no. arXiv:2601.03267, May 2026, arXiv:2601.03267 [cs.CL]. [Online]. Available: http://arxiv.org/abs/2601.03267

[55] Statistics Canada, "Demographic estimates by age and gender, provinces and territories," https://www150.statcan.gc.ca/n1/pub/71-607-x/71-607-x2020018-eng.htm, 2025, catalogue no. 71-607-X. Accessed: 2026-04-06.

[56] S. Stein and S. J. McKenna, "Recognising complex activities with histograms of relative tracklets," *Computer Vision and Image Understanding*, vol. 154, pp. 82–93, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1077314216301345

[57] A. Stisen, H. Blunck, S. Bhattacharya, T. S. Prentow, M. B. Kjærgaard, A. Dey, T. Sonne, and M. M. Jensen, "Smart devices are different: Assessing and mitigatingmobile

sensing heterogeneities for activity recognition," ser. SenSys '15. New York, NY, USA: Association for Computing Machinery, 2015, p. 127–140. [Online]. Available: https://doi.org/10.1145/2809695.2809718

[58] X. Sun, X. Li, J. Li, F. Wu, S. Guo, T. Zhang, and G. Wang, "Text classification via large language models," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 8990–9005. [Online]. Available: https://aclanthology.org/2023.findings-emnlp.603/

[59] Y. Tang, L. Zhang, F. Min, and J. He, "Multiscale deep feature learning for human activity recognition using wearable sensors," *IEEE Transactions on Industrial Electronics*, vol. 70, no. 2, pp. 2106–2116, 2023.

[60] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, "Llama: Open and efficient foundation language models," 02 2023.

[61] Y. Wang, Y. Cai, M. Chen, Y. Liang, and B. Hooi, "Primacy effect of ChatGPT," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 108–115. [Online]. Available: https://aclanthology.org/2023.emnlp-main.8/

[62] Z. Wang, Y. Pang, and Y. Lin, "Large language models are zero-shot text classifiers," no. arXiv:2312.01044, Dec. 2023, arXiv:2312.01044 [cs]. [Online]. Available: http://arxiv.org/abs/2312.01044

[63] J. Wei, M. Bosma, V. Zhao Y., K. Guu, A. Wei Yu, B. Lester, N. Du, A. Dai M., and Q. Le V., "Finetuned language models are zero-shot learners," *arXiv preprint arXiv:2109.01652*, 2022.

[64] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. H. Chi, Q. V. Le, and D. Zhou, "Chain-of-thought prompting elicits reasoning in large language models," in *Proceedings of the 36th International Conference on Neural Information Processing Systems*, ser. NIPS '22. Red Hook, NY, USA: Curran Associates Inc., 2022.

[65] World Health Organization, "Ageing and health," https://www.who.int/news-room/fact-sheets/detail/ageing-and-health, Oct. 2025, accessed: 2026-04-06.

[66] ——, "WHO's work on the UN decade of healthy ageing (2021-2030)," https://www.who.int/initiatives/decade-of-healthy-ageing, nd, accessed: 2026-04-06.

[67] ——, "UN decade of healthy ageing," https://unece.org/sites/default/files/2023-11/9_WHO_UN%20Decade%20of%20Healthy%20Ageing.pdf, 2021, presentation hosted by the United Nations Economic Commission for Europe (UNECE). Accessed: 2026-04-06.

[68] ——, “WHO global network for age-friendly cities and communities,” https://extranet.who.int/agefriendlyworld/who-network/, nd, accessed: 2026-04-06.

[69] ——, “Municipalité amie des aînées – Age-Friendly World,” https://extranet.who.int/agefriendlyworld/network/municipalite-amie-des-aines-quebec/, 2023, accessed: 2026-04-06.

[70] Q. Xia, T. Maekawa, and T. Hara, “Unsupervised human activity recognition through two-stage prompting with chatgpt,” no. arXiv:2306.02140, 2023. [Online]. Available: http://arxiv.org/abs/2306.02140

[71] H. Yuan, S. Chan, A. P. Creagh, C. Tong, A. Acquah, D. A. Clifton, and A. Doherty, “Self-supervised learning for human activity recognition using 700,000 person-days of wearable data,” *npj Digital Medicine*, vol. 7, no. 1, Apr. 2024.

[72] H. Zhang and L. Xu, “Multi-stmt: Multi-level network for human activity recognition based on wearable sensors,” *IEEE Transactions on Instrumentation and Measurement*, vol. 73, pp. 1–12, 2024.

[73] Y. Zhang, K. Ayush, S. Qiao, A. A. Heydari, G. Narayanswamy, M. A. Xu, A. A. Metwally, S. Xu, J. Garrison, X. Xu, T. Althoff, Y. Liu, P. Kohli, J. Zhan, M. Malhotra, S. Patel, C. Mascolo, X. Liu, D. McDuff, and Y. Yang, “Sensorlm: Learning the language of wearable sensors,” no. arXiv:2506.09108, 2025, arXiv:2506.09108 [cs]. [Online]. Available: http://arxiv.org/abs/2506.09108

ANNEX A: ETHICS CERTIFICATE

This master's thesis is subject to an ethics certificate. The project was initially approved on September 10th 2020. It was renewed twice during this project; once on August 20th 2024 and another time on August 18th 2025. The certificate number is: CER VN 2021-487