Large Language Model-guided Semantic Alignment for Human Activity Recognition

HUA YAN, Lehigh University, USA
HENG TAN, Lehigh University, USA
YI DING, University of Texas at Dallas, USA
PENGFEI ZHOU, University of Pittsburgh, USA
VINOD NAMBOODIRI, Lehigh University, USA
YU YANG, Lehigh University, USA

Human Activity Recognition (HAR) using Inertial Measurement Unit (IMU) sensors is critical for applications in healthcare, safety, and industrial production. However, variations in activity patterns, device types, and sensor placements create distribution gaps across datasets, reducing the performance of HAR models. To address this, we propose LanHAR, a novel system that leverages Large Language Models (LLMs) to generate semantic interpretations of sensor readings and activity labels for cross-dataset HAR. This approach not only mitigates cross-dataset heterogeneity but also enhances the recognition of new activities. LanHAR employs an iterative re-generation method to produce high-quality semantic interpretations with LLMs and a two-stage training framework that bridges the semantic interpretations of sensor readings and activity labels. This ultimately leads to a lightweight sensor encoder suitable for mobile deployment, enabling any sensor reading to be mapped into the semantic interpretation space. Experiments on five public datasets demonstrate that our approach significantly outperforms state-of-the-art methods in both cross-dataset HAR and new activity recognition. The source code is publicly available at https://github.com/DASHLab/LanHAR.

CCS Concepts: • **Human-centered computing → Ubiquitous and mobile computing**.

Additional Key Words and Phrases: Human activity recognition, Natural language processing, Large language models

ACM Reference Format:
Hua Yan, Heng Tan, Yi Ding, Pengfei Zhou, Vinod Namboodiri, and Yu Yang. 2025. Large Language Model-guided Semantic Alignment for Human Activity Recognition. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 9, 4, Article 230 (December 2025), 25 pages. https://doi.org/10.1145/3770652

1 INTRODUCTION

Human Activity Recognition (HAR) based on data collected from Inertial Measurement Unit (IMU) sensors on mobile platforms such as smartphones and wearable devices is one of the key problems in mobile computing due to its important applications in fields such as healthcare [38, 39], safety [20, 40, 55], and industrial production [26, 54, 60]. However, variations in activity patterns, device types, and sensor placements across different individuals result in significant distributional differences between datasets, even when capturing the same activity [21, 52]. Consequently, the performance of HAR models deteriorates considerably in cross-dataset human activity

Authors' Contact Information: Hua Yan, Lehigh University, USA, huy222@lehigh.edu; Heng Tan, Lehigh University, USA, het221@lehigh.edu; Yi Ding, University of Texas at Dallas, USA, yi.ding@utdallas.edu; Pengfei Zhou, University of Pittsburgh, USA, pengfeizhou@pitt.edu; Vinod Namboodiri, Lehigh University, USA, vin423@lehigh.edu; Yu Yang, Lehigh University, USA, yuyang@lehigh.edu.



This work is licensed under a Creative Commons Attribution 4.0 International License.
© 2025 Copyright held by the owner/author(s).
ACM 2474-9567/2025/12-ART230
https://doi.org/10.1145/3770652

recognition scenarios [52]. Therefore, developing HAR models capable of generalizing across different datasets remains a critical research problem that needs solutions.

Existing work on cross-dataset HAR can be broadly categorized into two methods: domain adaptation and data augmentation. Domain adaptation methods [4, 14, 31, 61] aim to bridge the distribution gap between source and target datasets to enhance model performance in training. Data augmentation methods [30, 35, 43, 52] seek to increase the diversity of training data to improve the model's generalization ability. While both methods are valuable, they still suffer from two limitations: (i) Most existing work overlooks the physical meanings of activities. This can exacerbate the impact of classification errors on downstream applications. For instance, misclassifying a walking activity as running may have a relatively minor effect compared to misclassifying it as a significantly different activity such as biking (see quantitative results in Evaluation). (ii) Moreover, cross-dataset scenarios often introduce new activities that these methods struggle to handle, as these new activities are typically missing from the training phase.

We argue that semantic interpretations of sensor readings and activity labels can play a crucial role in addressing the aforementioned limitations as shown in Fig. 1. (i) For cross-dataset heterogeneity, interpreting sensor readings based on its physical meanings can help identify patterns associated with specific activities (e.g., jogging often exhibits regular periodicity). The underlying intuition is that, despite the variability across datasets, activities may still reveal common underlying patterns. By identifying these patterns, we can map the heterogeneous data into a common space, thereby mitigating the impact of dataset variability. (ii) When dealing with new activities, semantic interpretations of activity labels can enhance their meaning by generating detailed descriptions. Combining semantic interpretations of both sensor readings and activity labels creates the potential to map new data to new activity labels. However, manual semantic interpretation is resource-intensive. The recent success of large language models (LLMs) offers a promising solution, enabling the generation of semantic interpretations from both sensor readings and activity labels through natural language [11, 58]. Research has shown that LLMs have the ability to perceive aspects of the physical world [51], providing an opportunity to leverage these capabilities to bridge the gap in cross-dataset HAR through automated semantic interpretation.

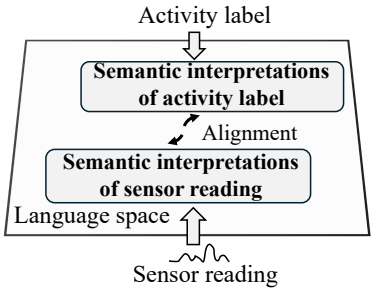


Fig. 1. Semantic interpretations for HAR

Despite the promise of semantic interpretations using LLMs, several challenges remain to be addressed: (i) Generation Challenge: The semantic interpretations generated by LLMs can suffer from issues like hallucinations and randomness. Ensuring consistent and high-quality semantic interpretations remains a significant challenge. (ii) Alignment Challenge: Once LLMs generate semantic interpretations of sensor readings and labels, the next step is to align them for human activity recognition, i.e., aligning sensor readings and labels based on their interpretations. While language models can be used to align these interpretations, general language models lack specific knowledge of IMU data and human activity recognition, limiting their ability to accurately understand and reason about activities. (iii) Deployment Challenge: Frequent access to cloud-based LLMs or deploying them locally on mobile devices is impractical due to high latency and resource constraints. A lightweight framework is needed to effectively leverage LLMs for on-device HAR.

To address these challenges, we design LanHAR a novel system that leverages LLMs to generate semantic interpretations of sensor readings and activity labels, aiming to improve cross-dataset human activity recognition. LanHAR consists of three key components. (i) Generation: we carefully design prompts to guide LLMs in generating semantic interpretations and develop an iterative re-generation method to ensure high-quality semantic outputs. (ii) Alignment: we design a semantic interpretation alignment module to enhance understanding of semantic interpretations based on a text encoder to encode interpretations and two contrastive learning tasks to

improve the alignment between activities and sensor readings. (iii) Deployment: We introduce a lightweight, two-stage training and inference framework. First, we train the semantic interpretation alignment module, then design and train a sensor reading encoder to map sensor data to the semantic interpretation space. For inference on mobile devices, we use the trained encoder to obtain sensor reading encodings and measure their similarity to activity label semantics to determine the HAR results.

In particular, our main contributions are as follows.

- We explore the use of LLM-generated semantic representations to improve cross-dataset human activity recognition. Our method maps both sensor readings and activity labels into a shared semantic space using natural language descriptions. This alignment reduces dataset heterogeneity and enables the recognition of activities that are not present in the training data.
- We design LanHAR, a novel system that leverages LLMs to generate semantic interpretations of sensor readings and activity labels, addressing the challenge of cross-dataset human activity recognition. The system features a semantic interpretation generation process with an iterative re-generation method to ensure high-quality outputs, alongside a two-stage training framework that transfers the capabilities of large language models into lightweight, privacy-preserving models suitable for sustainable deployment on resource-constrained edge devices.
- We evaluate our system based on five public datasets. The experimental results demonstrate that it outperforms other state-of-the-art methods in cross-dataset activity recognition, achieving an improvement of 7.35% in accuracy and 13.16% in F1 score. Additionally, for new activity recognition, it shows further enhancement, with improvements of 43.67% in accuracy.

2 RELATED WORK

2.1 Human Activity Recognition

Human Activity Recognition (HAR) leverages data from various sensors such as accelerometers and gyroscopes to detect and classify human activities [56, 57]. To mitigate the issue of data heterogeneity, existing works can be classified into three categories, including semi-supervised learning methods, domain adaptation methods, and data augmentation methods.

In HAR, collecting large-scale labeled datasets is particularly challenging due to the high cost of manual annotation, while abundant unlabeled sensor data are readily available. To address this imbalance, semi-supervised methods [24, 29, 35, 53, 59] have been developed to maximize the use of unlabeled data by learning valuable feature representations without requiring explicit labels. These approaches enable models to adapt more efficiently to downstream tasks with only a minimal amount of labeled data, which is crucial for improving generalization across diverse datasets. For example, UniMTS [59] is a unified pre-training framework designed for motion time-series data, with zero-shot and few-shot generalization across diverse human activity recognition tasks. LIMU-BERT [53] leverages masked language modeling and self-supervised learning to effectively utilize unlabeled IMU sensor data. Despite their capabilities, these models still depend on a certain amount of labeled data to train classifiers, which may struggle when applied to target datasets that lack any labeled data. GOAT [24] uses pre-training with textual attributes to improve activity recognition across multiple datasets, supporting cross-dataset generalization.

The core idea of domain adaptation methods [4, 14, 31, 61] involves taking a classifier that has been pre-trained on a source domain and using it with an unlabeled dataset from a target domain, then adjusting the source model's weights to enhance its performance in the target domain. For example, SDMix [22] aims to mix features from the source and target domains in the feature space while emphasizing the preservation of semantic discrimination, in order to reduce the discrepancy between different domains. UDAHAR [4] systematically studies the effectiveness of various unsupervised domain adaptation methods for human activity recognition tasks. It benchmarks multiple

UDA techniques, such as feature alignment and adversarial adaptation, providing a comprehensive evaluation of their robustness under different domain shifts. While domain adaptation methods mitigate domain shifts, they often struggle with large domain discrepancies, depend heavily on abundant source data

Data augmentation [30, 35, 42, 43, 48, 52] in human activity recognition aims to increase the diversity and quality of training data, thereby enhancing the model's generalization ability and accuracy. For example, UniHAR [52] designs physical-informed augmentation methods to improve the cross-dataset HAR performance.

However, All the methods are valuable, they still suffer from two limitations: (i) Most existing work overlooks the physical meanings of activities. This can exacerbate the impact of classification errors on downstream applications. For instance, misclassifying a walking activity as running may have a relatively minor effect compared to misclassifying it as a significantly different activity such as biking. (ii) Moreover, cross-dataset scenarios often introduce new activities that these methods struggle to handle, as these new activities are typically missing from the training phase. Our potential idea to solve these limitations is to use and map semantic interpretations of sensor readings and activity labels in a shared semantic space to mitigate cross-dataset heterogeneity and generate new activity.

Recognizing new activities has been a longstanding challenge in HAR, particularly for machine learning-based methods, which are unable to identify activities that were not present during the training phase. The most relevant method [6] attempts to address the challenge of recognizing new activities by using pre-defined attributes, such as actions of specific body parts. The method trains two mappings: one from sensor readings to predefined attributes, and another from predefined attributes to activity labels. However, the quality of these predefined attributes significantly influences the model performance. Additionally, this work is not well-suited for cross-dataset new activities, as the new activities may differ significantly from the original dataset.

2.2 LLM for Human Activity Recognition

Recently, there has been a surge of interest in leveraging the capabilities of LLMs [12, 25, 41] to enhance mobile computing applications [17, 50]. Penetrative AI [51] explores the potential of leveraging large language models to comprehend and interact with the physical world. Kim et al. [15] harness the power of large language models to improve health prediction using data collected from wearable sensors. IMUGPT2.0 [18] utilizes LLMs for virtual IMU data generation since IMU-labeled data is challenging to collect. It incorporate motion filtering and diversity metrics to enhance the efficiency and practicality of synthetic data generation for HAR. MotionGPT [10] fuses language data with large-scale motion models, and motion-language pre-training that can enhance the performance of motion-related tasks becomes feasible. HARGPT [9] leverages LLMs as zero-shot human activity recognizers by framing sensor-based activity recognition as a natural language processing task. ContextGPT [1] uses LLMs to extract commonsense knowledge about the environmental contexts of human activities, enhancing context-aware HAR with limited supervision.

Recent works exploring the use of large language models for human activity recognition have demonstrated the potential of LLMs to enhance HAR in various ways and have inspired our approach to leverage LLMs for generating semantic interpretations of IMU data. Nevertheless, while many studies explore LLMs in the context of HAR, a few directly address HAR tasks, and even those that still face challenges in enabling efficient deployment on resource-constrained mobile devices.

3 MOTIVATION

3.1 LLMs Possess the Ability to Perceive the Physical World

Existing research [9, 51] has demonstrated that LLMs possess the ability to perceive and interpret the physical world. For instance, LLMs can analyze various types of sensor data to infer a person's location or activity. Building on this capability, we explore the use of LLMs to analyze sensor readings and describe activity labels, thereby

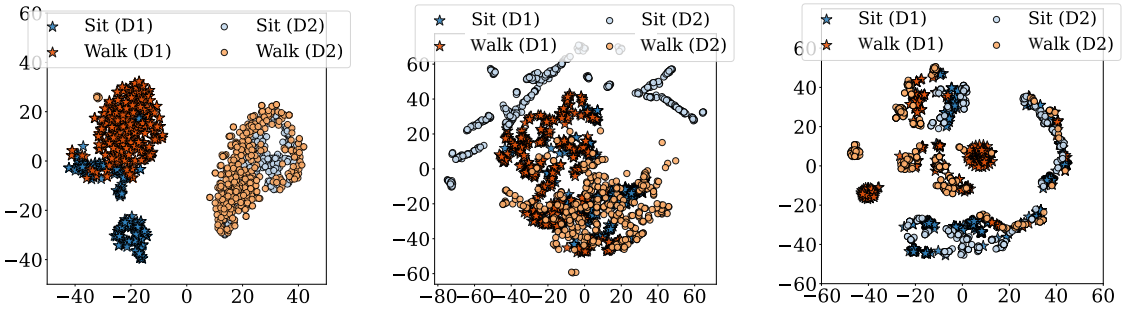
Semantic interpretation of activity label example

- **Response from GPT-4:** Jogging involves a steady and moderate running pace where both feet are off the ground simultaneously… During jogging, the accelerometer may detect a rhythmic, repetitive pattern of acceleration and deceleration corresponding to each step… The gyroscope may capture consistent oscillatory rotations of the body, especially in the vertical and horizontal planes, reflecting the body's movement and balance adjustments with each stride.

Semantic interpretation of sensor reading example

- **Response from GPT-4:** Possible accelerometer patterns: (i) Strong Periodic Vibrations: The sensor data likely reflects strong periodic vibrations, indicating intense and pronounced changes in acceleration, with a fast vibration frequency and large amplitude. (ii) Significant Acceleration Peaks: There are significant acceleration peaks in the data, showing large changes in acceleration within each vibration cycle.

Fig. 2. Example of semantic interpretations of sensor readings and activity labels



(a) Raw data (b) Self-supervised learning (c) Semantic interpretations

Fig. 3. Data distribution under three settings across two datasets

generating semantic interpretations for both. Specifically, we design prompts that include data introduction, data analysis, relevant knowledge, and task instructions (detailed in Section 4.3) to guide LLMs in generating semantic interpretations of sensor readings. Similarly, we guide LLMs to describe activity labels. Fig. 2 provides an example where GPT-4 generates its understanding of jogging alongside an analysis of sensor data sequences.

3.2 Semantic Interpretations of Sensor Readings for Cross-dataset HAR

We explain why LLM-generated semantic interpretation offers a valuable method to address data heterogeneity in cross-dataset HAR. The key intuition is that, despite variations in how heterogeneous data manifests, there are often underlying common activity patterns. By identifying these shared patterns, we can map the original heterogeneous sensor data into a common shared space (i.e., a language space with encoded semantic interpretations), effectively reducing the impact of data heterogeneity.

To understand how data heterogeneity is mitigated using generated semantic interpretations, we compare data distributions under three settings: raw data, data encoded by encoders, and generated semantic interpretations. Data encoding is a common method in existing research [2], aimed at learning better representations or embeddings of sensor readings. For data encoding, we employ a self-supervised learning model, BERT [7], as a representative method, following prior work [53] that masks and predicts parts of the sensor readings. For

Table 1. Relative KL divergence between the distributions of the same activity across two datasets.

Activity	Raw data	Self-supervised learning	Semantic interpretations
Walking	0.4965	0.2331	0.1633
Sitting	0.1669	0.1983	0.1541
Going upstairs	0.2444	0.1875	0.0420
Going downstairs	0.1753	0.1518	0.0995
Average	0.2708	0.1926	0.1147

semantic interpretation, we use GPT-4 to generate interpretations from raw IMU data based on a designed prompt (detailed in Section 4.3). These interpretations are then input into a pre-trained language model (BERT) to produce representations in the language space. We use two public datasets—UCI [34] (D1) and Motion [23] (D2) as a cross-dataset example including four common activities: walking, sitting, going upstairs, and going downstairs.

Qualitative results: We visualize the results from the three settings using t-distributed Stochastic Neighbor Embedding (t-SNE) [44] to project them onto a 2D surface. As an example, we plot two activities, "sit" (blue) and "walk" (orange), in Fig. 3. Initially, in the raw data, the same activities from different datasets are far apart. After applying self-supervised learning, the distribution gap between the datasets decreases, but the activities also become less distinguishable. However, when using semantic interpretations, the distribution gap reduces further, and the same activities (same color) cluster more closely, demonstrating improved grouping of the same activities.

Quantitative results: We quantitatively assess the impact of introducing semantic interpretations for sensor readings across all activities. To measure the distribution differences between the two datasets under the three settings, we use relative Kullback-Leibler (KL) divergence, computed as the ratio of KL divergence between the distributions of the same activity to KL divergence between the distributions of different activities [16], a widely used metric for comparing distributions. Compared to KL divergence, relative KL divergence normalizes differences in scale, mitigates the inherent closeness of word embeddings, and ensures a fair comparison across activities. Table 1 presents the relative KL divergence for all activities across the two datasets under each setting. The results show that introducing semantic interpretations significantly reduces the distribution gap for all activities. On average, the KL divergence is reduced by 40.45%.

3.3 New Activity Recognition

Recognizing new activities has been a longstanding challenge in HAR, particularly for machine learning-based methods, which are unable to identify activities that were not present during the training phase. The most relevant method to address this challenge involves using predefined attributes of activities to establish connections between new and existing activities [6]. However, the performance of this method relies heavily on the quality of manually defined activity attributes, making it costly and difficult to generalize to new labels.

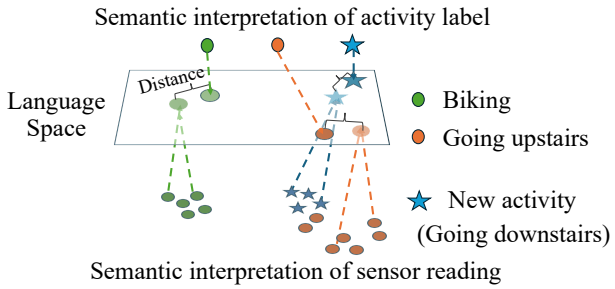


Fig. 4. Semantic interpretations for new activities.

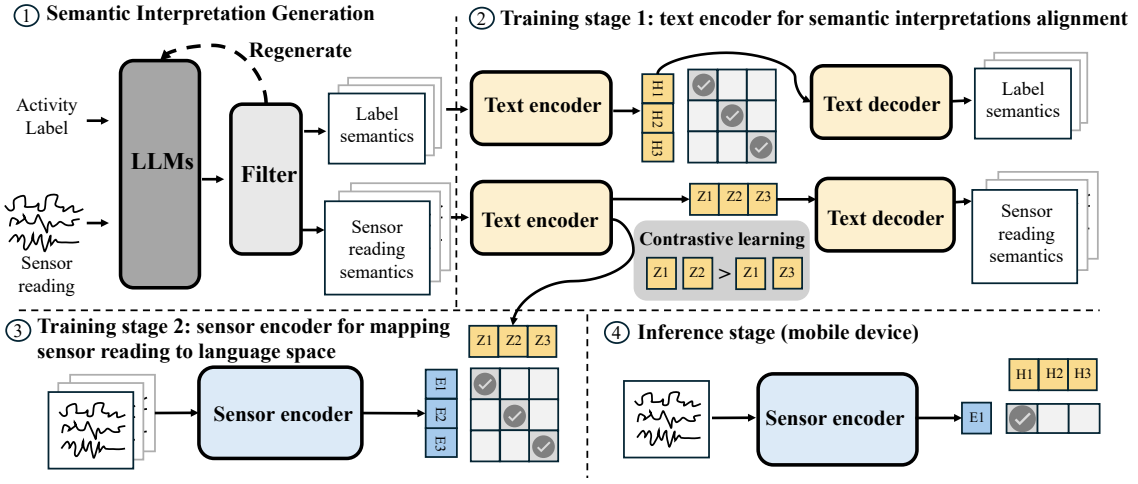


Fig. 5. Overview of LanHAR. (1) Utilize LLMs for generating semantic interpretations of sensor reading and activity labels. (2) Train a text encoder to encode two types of semantic interpretations and achieve their alignment. Hi and Z_i denote embeddings of semantic interpretations of activity labels and sensor reading. (3) Train a sensor encoder to align sensor reading and semantic interpretation. Ei denote embeddings of sensor reading. (4) For inference, only use the sensor encoder to generate embeddings for the sensor readings Ei and then compute the similarity with the pre-stored embeddings of the activity labels Hi to obtain the human activity recognition results.

We argue that combining the semantic interpretations of sensor readings and labels offers a new perspective for addressing this challenge. By converting all activity labels into a language space using their semantic interpretations, we can represent labels even without corresponding sensor readings (which is common for new labels). Similarly, with sensor readings mapped into the same language space, we can match each reading to a label, regardless of whether the label was previously encountered. Fig. 4 illustrates this concept, where we assume "biking" and "going upstairs" are known activities from the training phase, while "going downstairs" is a new activity. In our method, we first convert all activity labels into the same language space using their LLM-generated semantic interpretations. We then obtain semantic interpretations for the sensor readings. Even though "going downstairs" is unseen during training, we can still measure its distance to all activity labels in the language space, enabling the recognition of new activities. In our setting, we assume that we have the knowledge of possible activity labels in the inference stage. It is important to note that our approach does not consider open-set recognition or arbitrary label discovery. Instead, we follow a closed-set assumption, where the candidate activity labels are known in advance during inference. To support this, we pre-stored all semantic interpretations of activity labels and use them to guide activity recognition during inference. Note that this does not violate our "new activity" objective where we do not have access to the training data of new activities, instead, simply the "name" of the activities.

4 SYSTEM DESIGN

4.1 Problem Formulation

We consider LanHAR to be implemented in a cloud-client setting, where each client possesses locally collected, unlabeled IMU datasets. The cloud holds an initial source dataset, $\mathbf{X}^s$, collected from a subset of clients, with labels $\mathbf{Y}^s$ provided by clients or experts. Given an unlabeled target dataset $\mathbf{X}^t$ from the clients, our goal is to achieve high activity recognition accuracy on $\mathbf{X}^t$, even when the label set $\mathbf{Y}^s$ differs from $\mathbf{Y}^t$.

4.2 Design Overview

We design LanHAR, a novel system that uses LLM-generated semantic interpretations of sensor readings and activity labels for cross-dataset human activity recognition. As shown in Fig. 5, LanHAR consists of four processes:
(1) LLMs for semantic interpretations. We utilize LLMs to generate semantic interpretations of sensor readings $\mathbf{X}^s$ and activity labels $\mathbf{Y}^s$. We introduce an iterative re-generation process to filter out low-quality interpretations due to issues such as hallucinations.
(2) Text encoder for semantic interpretations alignment. We train a text encoder (i.e., initialized by a pre-trained language model) to encode and align the semantic interpretations generated in step (1). This alignment enables us to leverage language for human activity recognition by matching the semantic interpretation of sensor readings to the most similar semantic interpretation of activity labels within the language space.
(3) Sensor encoder for mapping sensor reading to language space. We train a sensor reading encoder, based on the text encoder from step (2), which is capable of mapping any sensor reading into the language space resulted from semantic interpretations.
(4) Inference on the mobile device. For inference, we deploy the sensor encoder from (3) on mobile devices to generate sensor embeddings, which are then compared with pre-stored (new) activity label embeddings to determine the activity through similarity computation.

4.3 LLMs for Semantic Interpretations

We introduce how we guide LLMs to generate semantic interpretations of sensor readings and activity labels. We also discuss our method to handling low-quality interpretations to ensure the reliability of LLM responses.

4.3.1 Obtaining Semantic Interpretations of Sensor Readings. We carefully design a prompt to help LLMs better understand the problem setting and deliver more accurate and meaningful answers. An example is shown in Fig. 6. The prompt consists of four key parts: data introduction, data analysis, knowledge, and task introduction.
(i) Data introduction: We provide details about the data, including its source, the meaning of each dimension, the sampling frequency, and the context in which the data is collected during different activities.
(ii) Data analysis: Intermediate steps (i.e., Chain-of-Thought [49]) can enhance LLMs' reasoning abilities. Our offline testing reveals that while LLMs are capable of performing various data analyses, skipping intermediate steps often leads to less confident responses. To address this, we design multiple auxiliary analysis steps (e.g., amplitude, frequency, time series analysis, and statistical measures such as mean, standard deviation, maximum, and minimum values). These steps help LLMs generate more accurate and precise answers.
(iii) Knowledge: We include relevant background knowledge to support the LLMs understanding the analysis procedure and focusing on the critical aspects among many possible features (e.g., impact peaks and vertical–horizontal dominance). The background knowledge encompasses a few key aspects of sensor pattern characteristics and highlights their importance in distinguishing between different specific activities or broader activity categories. The background knowledge is mainly derived from established findings in prior research on IMU-based human activity recognition [3, 5, 13, 19, 27, 46, 47]. For example, we referred to studies on how vertical and horizontal components reflect different movement mechanisms, how inertial signals vary across activities, and how dominant gyroscope axes can indicate trunk rotation or arm-swing patterns. When needed, we also consulted open sources (e.g., Wikipedia) for basic terminology. All information was then consolidated through a combination of LLM-based summarization and manual refinement.
(iv) Task introduction: We outline the task clearly, specifying what the LLMs need to achieve and the format of the final output. This ensures that the LLMs have a clear understanding of the task requirements and can deliver standardized responses that meet our expectations.

Prompt for semantic interpretation of sensor reading

- **Data Introduction:** The data to be analyzed consists of a short segment of IMU sensor recordings. The first three columns are accelerometer data, and the last three columns are gyroscope data, sampled at 50Hz. The context is that the IMU sensor reading may be in one of the following states: [walking, biking, jogging, going upstairs, going downstairs, stilling]…
- **Data Analysis:** amplitude analysis, frequency analysis, time series analysis, mean, std, max and min…
- **Knowledge:** Jogging: the accelerometer may detect a rhythmic, repetitive pattern of acceleration and deceleration corresponding to each step, typically showing peaks for foot strikes…
- **Task Introduction:** You task is analyzing the pattern of the above IMU data segment. We summarize the analysis into the following categories. For each category, analyze and answer the most fitting option if one applies. If you think there is a pattern that particularly fits, you are also welcome to add:
  - Category 1 Periodicity: [periodic rotation pattern, more intense periodic pattern, lack of periodic pattern…
  - …
- We provide an example of a final output:…

Fig. 6. An example prompt for obtaining semantic interpretation of sensor readings.

Prompt for semantic interpretation of activity label

- **Prompt:** You need to generate a detailed description for a given activity. The description should include three aspects:
  - General Description: Provide an overview of the activity, including its purpose, common settings where it is performed, and any other relevant details.
  - Sensor Patterns Detected by Accelerometer and Gyroscope: Describe the typical patterns detected by accelerometers and gyroscopes during the activity.
  - Body Parts Involved: Identify and describe the primary body parts and muscle groups engaged during the activity.

Fig. 7. An example prompt for obtaining semantic interpretation of activity labels.

4.3.2 Obtaining Semantic Interpretations of Activity Labels. Similarly, we design a prompt to guide LLMs in generating semantic interpretations of activity labels. In this prompt, we outline the task for the LLMs, instructing them to generate a description for a given activity. The description should cover three key aspects: a general overview of the activity, the potential states or patterns detected by the accelerometer and gyroscope during the activity, and the body parts likely involved in performing the activity. An example is shown in Fig. 7.

4.3.3 An Iterative Re-generation Method to Ensuring the Quality of LLM Responses. The quality of LLMs' interpretations is critical for our method. However, due to hallucinations or other incidental factors, LLMs may sometimes generate inaccurate or illogical responses. Through manual review, we identify numerous instances of this issue when generating semantic interpretations. To improve the accuracy of LLM responses, we develop an

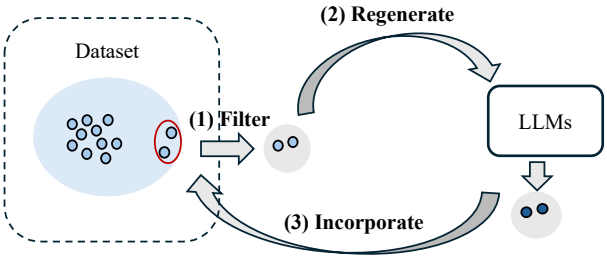


Fig. 8. The iterative re-generation method to ensuring the quality of LLM responses. (1) Filter inaccurate semantic interpretations. (2) Regenerate new semantic interpretations by LLMs. (3) Incorporate the new semantic interpretation based on its impact on the intra-cluster distance.

iterative re-generation method to ensure high-quality outputs. The key intuition is that refining responses through clustering-based filtering will lead to a lower intra-cluster distance within the same activity. As illustrated in Fig. 8, the framework first clusters the responses and identifies the k most problematic ones based on their deviation from the cluster center. These data points are re-input into the LLM to regenerate the semantic interpretations. We then assess whether to incorporate the new interpretations based on its impact on the intra-cluster distance. The details are as follows. We show in the evaluation that this process improves the data quality.

(1) Filter out inaccurate semantic interpretations: We first encode the generated semantic interpretations into dense vector representations using a pre-trained BERT encoder. We then apply a clustering-based method (i.e., K-means) to identify anomalous data points within the generated semantic interpretations of the same activity. After clustering, we select k outliers based on their distance from the cluster center, with those farther than the average intra-cluster distance considered low-quality responses for refinement.

(2) Regenerate new semantic interpretations by LLMs: After identifying the k outliers, we update the task introduction in the prompt (Section 4.3.1) to inform the LLMs of the inaccurate responses. The revised prompt includes the following: "*This is your previous response to the task. Please analyze it step by step to identify any logical errors, inconsistencies with real-world knowledge, or discrepancies with the input data. Provide a corrected response according to the required format.*"

(3) Incorporate the new semantic interpretation based on its impact on the cluster structure: If the newly generated interpretations reduce the intra-cluster distance, making the responses within the same cluster more compact, we accept the new semantic interpretation; otherwise, we retain the original one. We repeat this process until either the cluster structure remains stable (e.g., the intra-cluster distance change falls below a predefined threshold) for multiple continuous iterations or the specified number of iterations is reached.

Note that the iterative re-generation process occurs only during the offline training stage and does not impact the efficiency of online inference when deployed on mobile devices. In our study, only 13.30% of the semantic interpretations were identified as inaccurate.

4.4 Text Encoder for Semantic Interpretations Alignment

After generating the semantic interpretations of sensor readings and activity labels, we train a text encoder (initialized with a pre-trained language model) to encode and align these interpretations in a shared language space. To ensure the text encoder effectively captures the relationship between the semantic interpretations of sensor readings and activity labels, as well as their alignment, we design two subtasks: a contrastive learning task and a reconstruction task.

4.4.1 Text Encoder. We adopt a pre-trained language model as the text encoder due to its robust ability to understand semantic interpretations through language. Specifically, we use the text encoder to encode the

semantic interpretations of sensor readings $\mathbf{S}^z$ and activity labels $\mathbf{S}^l$, which are represented as $\mathbf{Z} = g_{\text{en}}(\mathbf{S}^z)$ and $\mathbf{H} = g_{\text{en}}(\mathbf{S}^l)$, where $g_{\text{en}}(\cdot)$ denotes the text encoder. In our work, we utilize the BERT architecture as the text encoder, initializing it with pre-trained BERT parameters [7]. BERT consists of multiple Transformer encoder layers, each incorporating multi-head self-attention mechanisms and feedforward neural networks, allowing it to generate word representations in a bidirectional context. We chose BERT because of its superior ability to understand text semantics and compare text similarity, as it captures word meanings in a bidirectional context, enabling nuanced comprehension and effective comparisons. Unlike generative models that process text in a left-to-right (or autoregressive) manner, BERT considers both the left and right context of each word simultaneously during training, allowing it to build deeper semantic representations based on the full sentence structure. However, this text encoder structure is flexible and can be substituted with other language models if needed.

4.4.2 Alignment Task. Inspired by multimodal alignment (e.g., vision and language) [32], we adopt contrastive learning to align the semantic representations of activity labels with those of sensor readings. This alignment enables the use of language to guide human activity recognition by matching the semantic interpretations of sensor readings to the most similar activity label interpretations in the language space. First, we normalize both the embeddings of the semantic interpretations of sensor readings and activity labels to have unit length. Then, we compute the similarity between the normalized embeddings using the dot product, defined as:

$$s(\mathbf{H}, \mathbf{Z}) = \frac{\mathbf{H}}{||\mathbf{H}||} \cdot \frac{\mathbf{Z}}{||\mathbf{Z}||}. \tag{1}$$

The goal of training is to maximize the similarity between matching pairs of semantic interpretations and minimize the similarity between non-matching pairs. Given a batch of N matching pairs, the loss function is defined as:

$$\mathcal{L}_{\text{align}} = -\log \frac{\exp(s(H_i, Z_i)/\tau)}{\sum_{j=1}^{N} \exp(s(H_i, Z_j)/\tau)} - \log \frac{\exp(s(H_i, Z_i)/\tau)}{\sum_{j=1}^{N} \exp(s(H_j, Z_i)/\tau)}, \tag{2}$$

where τ is a temperature parameter that scales the similarities, and H_i and Z_i represent the embeddings of the semantic interpretations of sensor readings and activity labels for the i-th matching pair.

4.4.3 Contrastive Learning Task. We design two contrastive learning subtasks to enable the text encoder to fully understand the knowledge related to human activity recognition. Firstly, we use pre-defined category-level relations (details in Section 5.6) to guide the comparison between the semantic interpretations of sensor readings and activity labels. For semantic interpretations of activity labels, two activity labels in the same category have a higher similarity than two activity labels in the different categories. The loss function for this task is defined as

$$\mathcal{L}_{C_{a_1}} = -\sum_{\mathcal{P}_{a_1}} (ln\sigma(s(H_i^{c_m} \cdot H_j^{c_m}) - s(H_i^{c_m} \cdot H_k^{c_n}))), \tag{3}$$

where $\mathcal{P}_{a_1}$ represents comparison pairs among the semantic interpretations of activity labels. $H_i^{c_m}$ and $H_j^{c_m}$ denote the embeddings of two activity label samples i and j, that belong to category c_m. Similarly, we apply the same comparison method to the semantic interpretations of sensor reading. The loss function for this task is defined as

$$\mathcal{L}_{C_{a_2}} = -\sum_{\mathcal{P}_{a_2}} (ln\sigma(s(Z_i^{c_m} \cdot Z_j^{c_m}) - s(Z_i^{c_m} \cdot Z_k^{c_n}))), \tag{4}$$

where $\mathcal{P}_{a_2}$ represents comparison pairs among the semantic interpretations of sensor readings. $Z_i^{c_m}$ and $Z_j^{c_m}$ denote the embeddings of two semantic interpretations of sensor readings samples i and j, that belong to category c_m.

Secondly, to enrich the semantic interpretations of activity labels and enable the model to adapt to diverse descriptions, we generate multiple descriptions for each activity label. These descriptions guide the comparison,

ensuring that the similarity of descriptions for the same activity is higher than that for different activities. The loss function for this task is defined as $\mathcal{L}_{C_{a_3}} = -\sum_{\mathcal{P}_{a_3}}(ln\sigma(s(H_i^{l_m} \cdot H_j^{l_m}) - s(H_i^{l_m} \cdot H_j^{l_n})))$, where $\mathcal{P}_{a_3}$ represents comparison pairs among the semantic interpretations of activity labels. $H_i^{l_m}$ and $H_j^{l_m}$ denote the embeddings of two different semantic interpretations of the activity label l_m.

4.4.4 Reconstruction Task. Lastly, we design a reconstruction task to retain the characteristics of the language model, ensuring it can function as a language model to understand the new activity descriptions and new sensor pattern descriptions. We design a text decoder to reconstruct semantic interpretations of sensor reading and activity labels. The objective of the text decoder can be formulated as $\hat{\mathbf{S}}^{\mathbf{z}} = g_{\text{de}}(\mathbf{Z})$ and $\hat{\mathbf{S}}^{\mathbf{l}} = g_{\text{de}}(\mathbf{H})$, where $\mathbf{Z}$ and $\mathbf{H}$ denote embeddings of semantic interpretations of sensor readings and activity labels, respectively, and $\hat{\mathbf{S}}^{\mathbf{z}}$ and $\hat{\mathbf{S}}^{\mathbf{l}}$ denote reconstructed semantic interpretations of sensor reading and activity labels, respectively. The text decoder consists of multiple layers of the Transformer decoder, each containing multi-head attention mechanisms and feedforward neural networks. The loss function of the reconstruction task is defined as $\mathcal{L}_{\text{re}} = \frac{1}{N}\sum_i^N CE(\hat{S}_i^z, S_i^z) + \frac{1}{N}\sum_i^N CE(\hat{S}_i^l, S_i^l)$, where CE denotes Cross-Entropy (CE) loss and N denotes the number of training samples. In total, the loss function of training the text encoder is defined as

$$\mathcal{L}_{\text{text}} = \mathcal{L}_{\text{align}} + \alpha(\mathcal{L}_{C_{a_1}} + \mathcal{L}_{C_{a_2}} + \mathcal{L}_{C_{a_3}}) + \beta(\mathcal{L}_{\text{re}}), \tag{5}$$

where α and β are weighting factors that control the contributions of the different tasks.

4.5 Sensor Encoder for Mapping Sensor Reading to Language Space

To bridge IMU sensor readings with their semantic interpretations and enable mobile deployment, we introduce a sensor encoder that maps IMU sensor readings to the language space, aligning them with the semantic interpretation encodings of the same sensor readings. We first describe the structure of the sensor encoder, followed by an explanation of the training process.

4.5.1 Sensor Encoder Structure. The objective of the sensor encoder is to capture key features and generate embeddings from IMU data, which can be formulated as: $\mathbf{E} = f_{sen}(\mathbf{X})$ where $\mathbf{X} \in \mathbb{R}^{S_{\text{dim}} \times L}$ represents the IMU sensor readings, and $\mathbf{E}$ denotes the corresponding embeddings. The structure of the sensor encoder is flexible, allowing for either a CNN or Transformer-based architecture. In our work, we adopt a Transformer-based sensor encoder. First, following [53], we normalize the IMU sensor readings $\mathbf{X}$ to obtain the normalized readings, $\mathbf{X^{nor}}$. Next, we project the normalized sensor readings through a linear layer and apply layer normalization to the projected data: $\mathbf{X}^{'} = \text{LayerNorm}(W_p\mathbf{X^{nor}})$ where W_p is the weight matrix of the linear layer. Then, we add positional encoding to the normalized data:$\mathbf{X^0} = \mathbf{X}^{'} + \text{PositionEncoding}(\mathbf{X}^{'})$ The position-encoded data $\mathbf{X^0}$ is then fed through the Transformer encoder, which processes the data across multiple layers to capture complex relationships and dependencies. Each Transformer layer consists of two main sub-layers: Multi-Head Self-Attention and a Feed-Forward Network, both followed by layer normalization and residual connections. For the i-th encoder layer, the operations are as follows: First, the Multi-Head Self-Attention mechanism calculates self-attention scores, producing the output:

$$\begin{aligned} \mathbf{X^{A_i}} &= \text{Attention}(\mathbf{X^{i-1}}) \\ \mathbf{X^{AN_i}} &= \text{LayerNorm}(\mathbf{X^{i-1}} + \mathbf{X^{A_i}}) \end{aligned} \tag{6}$$

Next, the output from the Multi-Head Self-Attention sub-layer is passed through the Feed-Forward Network:

$$\begin{aligned} \mathbf{X^{ANF_i}} &= \text{FeedForward}(\mathbf{X^{AN_i}}) \\ \mathbf{X^i} &= \text{LayerNorm}(\mathbf{X^{ANF_i}} + \mathbf{X^{AN_i}}) \end{aligned} \tag{7}$$

This process is repeated across multiple encoder layers, each feeding its output to the next.

4.5.2 Sensor Encoder Training. Similar to the alignment task in Section 4.4.2, we adopt contrastive learning to align embeddings of sensor reading with the embeddings of semantic interpretation of sensor reading. First, the embeddings of both the sensor reading and its semantic interpretation are normalized to unit length. Their similarity is calculated using the dot product between the normalized embeddings, denoted as $s(\mathbf{E}, \mathbf{Z}) = \frac{\mathbf{E}}{||\mathbf{E}||} \cdot \frac{\mathbf{Z}}{||\mathbf{Z}||}$. The goal of the training is to maximize the similarity between matching pairs while minimizing the similarity between non-matching pairs. Given a batch of N sensor reading and semantic interpretation of sensor reading pairs, the loss function is computed as:

$$\mathcal{L}_{\text{sensor}} = -\log \frac{\exp(s(E_i, Z_i)/\tau)}{\sum_{j=1}^{N} \exp(s(E_i, Z_j)/\tau)} - \log \frac{\exp(s(E_i, Z_i)/\tau)}{\sum_{j=1}^{N} \exp(s(E_j, Z_i)/\tau)} \tag{8}$$

where τ is a temperature parameter.

4.6 Training and Inference Process

Training process: (i) Training the text encoder: we use the semantic interpretations of sensor readings and activity labels from the source dataset to train the text encoder, embedding human activity recognition-related knowledge and semantics. (ii) Training the sensor encoder for all datasets using raw IMU data and its semantic interpretations (i.e., unlabeled data): The sensor encoder is designed to map any sensor reading into the semantic interpretation space of sensor readings.

Inference process: The model operates in an inference-only mode and does not require access to additional unlabeled data, as both the sensor encoder and activity label embeddings are pre-trained and deployed on the device. At inference time, IMU sensor readings are encoded and compared against the pre-stored activity label embeddings to compute similarity scores. The label with the highest similarity is selected as the recognized activity. This same process also applies when recognizing new activities.

5 EVALUATION

5.1 Evaluation Setup

5.1.1 Dataset. We first evaluate the performance of LanHAR using four public datasets with simple activities that have been extensively employed in existing research [52]. (1) HHAR [37] includes individuals performing six activities (sitting, standing, walking, going downstairs, going upstairs, and biking) using six different types of mobile phones. (2) UCI [34] captures six activities (i.e., standing, sitting, lying, walking, going downstairs, and going upstairs). (3) Motion [23] involves 24 participants performing six activities (sitting, standing, walking, going upstairs, going downstairs, and jogging). (4) Shoaib [36] includes 10 participants engaged in seven activities (sitting, standing, walking, going upstairs, going downstairs, jogging, and biking). In addition, we also select datasets (i.e., PAMAP2 [33]) with more complex activities to show the performance of LanHAR. PAMAP2 consists of data collected from 9 participants performing 12 different activities, including both simple activities such as sitting, standing, and walking, and more complex activities such as vacuum cleaning and ironing.

5.1.2 Data Preprocessing. We preprocess all datasets to ensure consistent sampling rates and window sizes. The sampling rate is set to 50 Hz to simulate a resource-constrained scenario, following prior studies [52]. For cross-dataset activity recognition, we focus on four common activities (walking, going upstairs, going downstairs, and still) that are shared across the four datasets. Following prior work [52], we group sitting and standing under the unified category still. We select one dataset as the labeled source and one of the remaining three as the target, resulting in 12 source–target combinations. Datasets are categorized based on their roles as either source or target. For source datasets, each is split into a training set (80%) and a validation set (20%). For target datasets, the entire dataset is used as the test set to directly assess cross-dataset performance. In addition, during the second stage of training, we incorporate unlabeled data from the source datasets to jointly train the sensor encoder. This

helps the model learn more generalizable representations, even though the main supervision still comes from a single labeled source. For new activity recognition, we train the model on the four common activities from the source dataset. We then select activities from the target dataset that differ from the four common activities as new activities. Finally, we evaluate the model's ability to recognize these new activities by determining which activity labels the new data correspond to.

5.1.3 Baselines. We compare our method LanHAR with three categories of baselines: conventional HAR, cross-dataset HAR, and new activity baselines.
(i) Conventional HAR baselines: DCNN [56] is widely used in the field of human activity recognition for their ability to automatically extract local features from time-series sensor data. **Transformers [45]** utilize self-attention mechanisms to capture long-range dependencies in sensor data. **LIMU-BERT [53]** leverages masked language modeling and self-supervised learning to effectively utilize unlabeled IMU sensor data.
(ii) Cross-dataset HAR baselines: SDMix [22] It is a method for cross-domain activity recognition that uses the mixup method that addresses semantic inconsistencies caused by domain gaps. **UDAHAR [4]** In our setting, we follow the feature alignment methods UDA paradigm, which aims to minimize the domain gap by explicitly aligning the feature distributions of the source and target domains. **UniHAR [52]** is a method for cross-dataset activity recognition. It designs physical-informed augmentation methods to improve the cross-dataset HAR performance. **UniMTS [59]** is a unified pre-training framework designed for motion time-series data.
(iii) New activity baselines: DCNN [56] To adapt it for recognizing new activities, we fine-tune it using a small portion of data from the target dataset. **Nuactiv [6]** is a method designed for new activities. It predefined activity label-related attributes: actions of a specific part of the body. **UniMTS [59]** is a unified pre-training framework designed for motion time-series data, with zero-shot and few-shot generalization across diverse human activity recognition tasks. **HARGPT [9]** leverages LLMs as zero-shot human activity recognizers by framing sensor-based activity recognition as a natural language processing task.
(iv) Variants of our model: (1) LanHAR without Subtasks of Text Encoder (w/o STE). We remove the subtasks from the text encoder and focus its training on alignment. **(2) LanHAR without Training of Text Encoder (w/o TTE).** We remove the training part of the text encoder and directly use a pre-trained BERT model to replace it. **(3) LanHAR with LLAMA2 (w LLAMA2).** We replace GPT-4 with LLAMA2 to obtain semantic interpretations. **(4) LanHAR with GPT-3 (w GPT-3).** We replace GPT-4 with GPT-3 to obtain semantic interpretations. **(5) LanHAR with Different Language Model (w DLM).** We use the transformer architecture to replace BERT in LanHAR. **(6) LanHAR without Iterative re-Generation (w/o IG).** We remove the iterative re-generation method. **(7) LanHAR without Iterative re-Generation and designed Prompt (w/o IGP).** We remove the iterative re-generation method and the designed prompt. We directly use a simple prompt only containing data and task requirements.

5.1.4 Metric. We evaluate our model using two metrics: accuracy and F1 score. For cross-dataset HAR performance, we use accuracy to measure the correctness of the classification results for the four selected activities. For new activities, we use accuracy to evaluate the correctness of the classification results for the new activity only.

5.1.5 Implementation. We implement our method and baselines using PyTorch 1.9.0 in a Python 3.7 environment and train them on two NVIDIA RTX A5000 GPUs, each with 24 GB of memory. For LLMs, we choose GPT-4 as our LLMs to obtain semantic interpretations of sensor readings and activity labels. For the text encoder, we use the BERT architecture as its structure and initialize it with the pre-trained BERT parameters [7]. For the sensor encoder, we set the embedding dimension to 768 to maintain consistency with pre-trained BERT. The multi-head attention parameter is set to 2 and the encoder layer consists of 3 layers. We utilize the AdamW optimizer with a learning rate of 1e-5 and set the batch size to 32. For iterative re-generation, we set the threshold for intra-cluster distance change to 0.0005.

Table 2. Overall performance. Bolded scores indicate the best, while underlined scores represent the second best.

Method	Source	HHAR			UCI			Motion			Shoaib			Average
	Target	UCI	Motion	Shoaib	HHAR	Motion	Shoaib	HHAR	UCI	Shoaib	HHAR	UCI	Motion	
DCNN	Accuracy	0.181	0.220	0.208	0.230	0.122	0.299	0.514	0.605	0.564	0.470	0.651	0.571	0.392
	F1 scores	0.153	0.136	0.098	0.154	0.069	0.245	0.321	0.424	0.443	0.419	0.338	0.555	0.266
Transformer	Accuracy	0.465	0.505	0.387	0.512	0.451	0.400	0.428	0.449	0.423	0.506	0.450	0.401	0.449
	F1 scores	0.318	0.385	0.224	0.169	0.155	0.143	0.256	0.279	0.314	0.168	0.155	0.143	0.231
LIMU-BERT	Accuracy	0.615	0.257	0.355	0.310	0.315	0.169	0.352	0.292	0.344	0.509	0.392	0.321	0.353
	F1 scores	0.575	0.116	0.283	0.151	0.191	0.153	0.256	0.173	0.275	0.472	0.281	0.222	0.263
UDAHAR	Accuracy	0.473	0.481	0.249	0.446	0.417	0.315	0.428	0.559	0.591	0.452	0.551	0.561	0.460
	F1 scores	0.317	0.407	0.191	0.279	0.336	0.292	0.355	0.471	0.473	0.265	0.398	0.281	0.339
SDMix	Accuracy	0.656	0.694	0.547	0.619	0.567	0.626	0.446	0.531	0.580	0.461	0.549	0.611	0.574
	F1 scores	0.489	0.657	0.369	0.653	0.348	0.542	0.195	0.323	0.448	0.205	0.388	0.400	0.423
UniMTS	Accuracy	0.623	0.687	0.542	0.604	0.542	0.602	0.523	0.598	0.611	0.637	0.505	0.696	0.600
	F1 scores	0.523	0.501	0.387	0.610	0.351	0.528	0.357	0.436	0.509	0.327	0.402	0.511	0.454
UniHAR	Accuracy	0.739	0.752	0.613	0.805	0.786	0.741	0.667	0.735	**0.749**	0.525	0.767	**0.817**	0.721
	F1 scores	0.604	0.651	0.592	0.700	0.669	0.709	0.581	0.681	0.714	0.480	0.681	**0.777**	0.646
LanHAR	Accuracy	**0.830**	**0.812**	**0.730**	**0.855**	**0.811**	**0.782**	**0.697**	**0.764**	0.730	**0.717**	**0.789**	0.768	**0.774**
	F1 scores	**0.804**	**0.795**	**0.693**	**0.808**	**0.765**	**0.718**	**0.617**	**0.713**	**0.724**	**0.687**	**0.730**	0.721	**0.731**

5.2 Overall Performance

5.2.1 Comparison to Baselines Using Single Labeled Source Dataset. We compare our method on various baselines, and the results are presented in Table 2. Overall, our method demonstrates superior average performance across 12 different source and target dataset combinations, leading in the majority of settings. Compared to conventional HAR baselines, our method consistently outperforms them. These traditional baselines are not specifically designed for cross-dataset HAR, and their strong performance typically depends on large, carefully curated datasets. Compared to cross-dataset HAR baselines, our method outperforms them (except in the setting from Shoaib to Motion). Although SDMix and UDAHAR are also designed for cross-dataset HAR, they have specific requirements. For instance, UDAHAR may be more suitable for datasets with smaller distributional gaps. This consistent performance demonstrates the robustness and reliability of our method in diverse datasets.

Following the reviewer's suggestion, we extend our cross-dataset experiments to include all relevant activities, treating sitting and standing as separate classes rather than merging them into a single still category. Specifically, the evaluated activities now include sitting, standing, walking, going upstairs, and going downstairs. The new cross-dataset results for five activities are as shown in Appendix Table 5. As shown in the results, the inclusion of both sitting and standing as separate activities introduces additional difficulty, as these two activities exhibit highly similar sensor patterns and are often hard to distinguish, especially in cross-dataset settings. This leads to a noticeable performance drop across all methods. While the performance of our method also drops due to the increased difficulty, it still achieves the best results among all compared approaches. This suggests that, despite the challenge of distinguishing highly similar activities such as sitting and standing, our method remains comparatively more effective.

5.2.2 Comparison to Baselines Using Multiple Labeled Source Datasets. In addition to baselines trained on a single labeled source dataset, some existing works aim to mitigate cross-dataset heterogeneity by leveraging multiple labeled source datasets. In this section, we compare our method with two types of approaches. The specific implementations are as follows: (i) GOAT [24]: We implemented the GOAT framework following the original

paper's setting, where the model is pre-trained on three datasets and evaluated on a separate target dataset. Since most public datasets do not provide sensor position annotations, we removed the device location descriptions to ensure consistent input formats across datasets. (ii) Combining Public HAR Datasets [29]: We implement the two base models used in this work (i.e., HART [8] and DeepConvLSTM [28]) and adopt the multi-source pre-training setup on three datasets. However, the original method assumes access to the target dataset for fine-tuning, while our method does not require any target-domain labels. To ensure a fair comparison, we evaluated their approach under two settings: (i) No fine-tuning: testing on the target dataset without fine-tuning, and (ii) Fine-tuning: using 5% of labeled target data. Results are reported in Table 3.

From the results in the no fine-tuning scenario, we observe that merely using more data for pre-training does not guarantee improved performance. While aggregating multiple datasets can help reduce domain differences, effective performance typically requires an extensive and diverse set of source data. Otherwise, substantial distribution gaps between source and target domains still limit the generalization of pre-trained models. In contrast, although fine-tuning significantly boosts performance, it relies heavily on labeled target-domain data, which is frequently unavailable in practical scenarios. Our proposed method achieves robust performance without requiring any labeled data from the target domain. This demonstrates superior adaptability, performing notably better than other methods without fine-tuning and comparably to methods that use fine-tuning.

Table 3. Comparison to baselines using multiple labeled source datasets. Bolded scores indicate the best.

Method	Metric	HHAR	Motion	UCI	Shoaib
Goat	Accuracy	0.431	0.549	0.704	0.535
	F1 scores	0.156	0.262	0.573	0.402
Public-HART	Accuracy	0.432	0.542	0.695	0.593
(No fine-tuning)	F1 scores	0.169	0.253	0.567	0.462
Public-DeepConv	Accuracy	0.458	0.576	0.731	0.646
(No fine-tuning)	F1 scores	0.221	0.311	0.609	0.545
Public-HART	Accuracy	**0.798**	**0.803**	**0.830**	**0.862**
(Fine-tuning)	F1 scores	**0.731**	**0.764**	**0.779**	**0.830**
Public-DeepConv	Accuracy	0.792	0.762	0.822	0.767
(Fine-tuning)	F1 scores	0.673	0.697	0.757	0.703
LanHAR	Accuracy	0.756	0.797	0.794	0.747
	F1 scores	0.704	0.760	0.749	0.712

5.3 New Activity Performance

We compare our method with four new activity recognition baselines. Specifically, we train the model on four common activities from the source dataset and evaluate it on different activities from the target dataset that are not among the four common activities, treating them as new activities. The trained model is then used to recognize these new activities, i.e., determining which activity new data belong to. Fig. 9 presents the performance of all methods in recognizing new activities across multiple target datasets.

For new activity recognition, our model outperforms all the baselines: Our model performance is superior to that of a fine-tuned DCNN model, which indicates its advantage over simple fine-tuning on small datasets. Additionally, our model performs better than Nuactiv, which relies on predefined attributes for activities. Defining these attributes is challenging and varies significantly across different activity contexts, directly impacting Nuactiv's performance. This limitation hinders Nuactiv's performance in cross-dataset settings. Similarly, HARGPT, as a prompt-based HAR method, has limited ability to recognize new activities. UniMTS, although trained on large-scale simulated data, still does not perform well in zero-shot scenarios on real-world datasets.

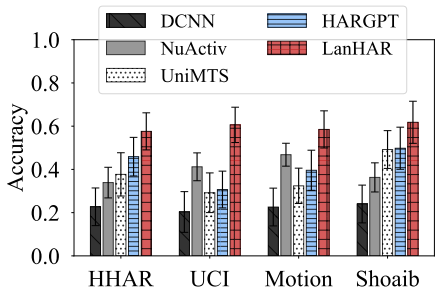


Fig. 9. New activity performance.

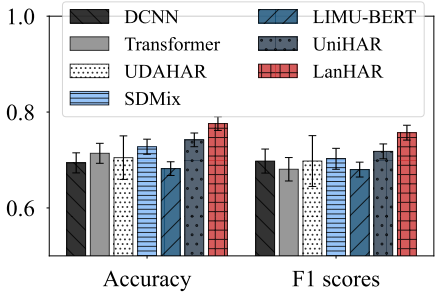


Fig. 10. Complex activity performance.

5.4 Complex Activity Performance

In addition to simple activities such as sitting and walking, we also select datasets containing more complex activities such as vacuum cleaning, ironing, and rope jumping to validate the effectiveness of our model. However, since different datasets contain different complex activities, to our knowledge, there are no two truly comparable datasets for cross-dataset HAR evaluation. Therefore, instead of using two different datasets, we select a single dataset and split it based on individuals to simulate cross-dataset performance. Specifically, we randomly partition the PAMAP2 dataset into two subsets based on individuals' IDs and select the partition with the highest KL divergence, indicating the greatest data variance across two subsets, as the final partitioning result. The evaluation result is presented in Fig. 10. As illustrated, our model demonstrates superior performance compared to other models on datasets that include complex activities, achieving an improvement in accuracy of at least 4.58% and at most 13.78%, and an increase in F1 scores of at least 5.43% and at most 11.32%.

5.5 Non-cross-dataset Performance

To evaluate the model's performance in a standard within-dataset setting, we conduct additional non-cross-dataset experiments. In this setup, the model is trained and tested on the same dataset (i.e., HHAR, Motion, UCI, and Shoaib). Specifically, each dataset is split into training and testing sets using an 8:2 ratio. The results are as shown in Table 4.

As shown in the results, most methods, including ours, achieve high accuracy in this setting, likely because each single dataset is collected from a relatively homogeneous group of participants under similar conditions, resulting in low distributional variability. Although our model is not the best-performing method on every dataset, it remains highly competitive. In terms of accuracy, the best-performing method on each dataset outperforms the worst-performing one by an average of 7.17%, while the gap between the best-performing method and ours is only 3.72% on average. Similarly, for F1-score, the average difference between the best and worst methods per dataset is 9.05%, while the best-performing method exceeds ours by just 3.91% on average. The slightly lower performance of our model may be due to the loss of some dataset-specific information during the semantic interpretation process, which helps generalization across datasets but may limit performance within a single dataset.

5.6 Category-level HAR Performance

We further evaluate the category-level accuracy of HAR performance. An activity is considered correctly classified if it is assigned to a similar activity category and incorrectly classified if it falls into a dissimilar category. The motivation for this evaluation is that even if a model does not provide the correct prediction, we expect its misclassification to be a similar activity rather than a completely unrelated one. Specifically, we group seven activities (walking, jogging, biking, sitting, standing, going upstairs, going downstairs, and lying) into three

Table 4. Non-cross-dataset performance. Bolded scores indicate the best.

Method	Metric	HHAR	UCI	Motion	Shoaib
DCNN	Accuracy	**0.995**	0.947	0.980	**0.982**
	F1 scores	**0.993**	0.954	0.967	**0.982**
Transformer	Accuracy	0.932	0.924	0.926	0.940
	F1 scores	0.929	0.918	0.903	0.935
LIMU-BERT	Accuracy	0.961	**0.976**	**0.983**	0.940
	F1 scores	0.961	**0.962**	0.980	0.947
UDAHAR	Accuracy	0.992	0.930	0.980	0.978
	F1 scores	0.991	0.938	0.967	0.978
SDMix	Accuracy	0.949	0.957	**0.983**	0.924
	F1 scores	0.949	0.934	**0.982**	0.931
UniMTS	Accuracy	0.936	0.935	0.953	0.961
	F1 scores	0.912	0.924	0.947	0.954
UniHAR	Accuracy	0.927	0.937	0.971	0.900
	F1 scores	0.886	0.904	0.949	0.882
LanHAR	Accuracy	0.956	0.951	0.922	0.967
	F1 scores	0.949	0.954	0.906	0.965

categories based on their similarity: Category 1 (walking, jogging, biking), Category 2 (sitting, standing, lying), and Category 3 (going upstairs, going downstairs). We compare our method against all the baselines. The average results are shown in Fig. 11. As shown, our method improves category-level HAR accuracy compared to the baselines. This improvement has the potential to reduce the impact of classification errors on downstream applications that require a broader activity categorization rather than fine-grained recognition.

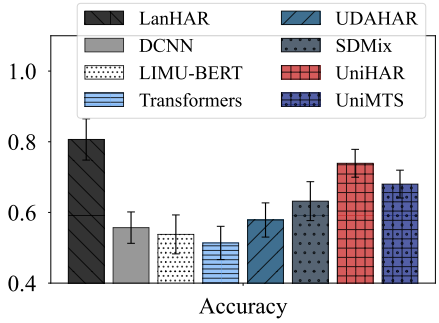


Fig. 11. Category-level HAR Performance.

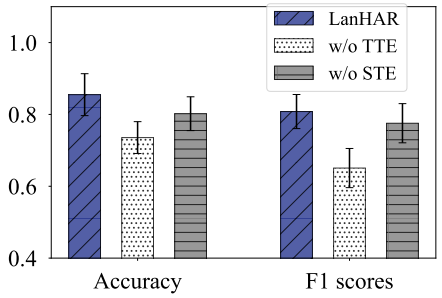


Fig. 12. The Effect of Text Encoder.

5.7 Ablation Study

5.7.1 The Effect Text Encoder. To assess the impact of the subtasks in the text encoder, we compare our method with a variant (i.e., w/o STE), as shown in Fig. 12. In the w/o STE variant, we remove the subtasks (i.e., the two contrastive learning and reconstruction tasks) and focus only on aligning the semantic interpretations of sensor readings and activity labels. As shown in the figure, removing the subtasks leads to a decline in performance compared to our full model. This demonstrates that the subtasks we designed help the model better understand IMU sensor readings and HAR-related semantics.

To further explore the effect of the text encoder on semantic interpretation alignment, we replace the carefully designed text encoder with a pre-trained BERT model and directly use it for training in Stage 2. The results are shown in Fig. 12. From the figure, it is clear that the performance of w/o TTE is lower than both w/o STE and our full model. This suggests that pre-trained language models without fine-tuning cannot fully capture HAR-related semantics or effectively align different HAR-related semantic interpretations.

5.7.2 The Impact of Different Pre-trained LMs. Our text encoder is flexible and allows the integration of different language models. In our work, we employ a BERT-based architecture, fine-tuned on our designed tasks. To investigate the impact of different language model architectures on HAR performance, we replace the BERT-based encoder with GPT-2. As shown in Fig. 13, this replacement results in decreased HAR performance. BERT is generally more suitable for tasks that require understanding sentence meaning and comparing text similarities, as its architecture and pre-training are optimized for capturing semantics and contextual relationships in a bidirectional manner. In contrast, GPT-2, with its unidirectional processing, is less effective at understanding sentence-level meaning and comparing text similarity, though it excels in generating coherent and contextually relevant text. This explains the performance drop when GPT-2 is used for HAR tasks.

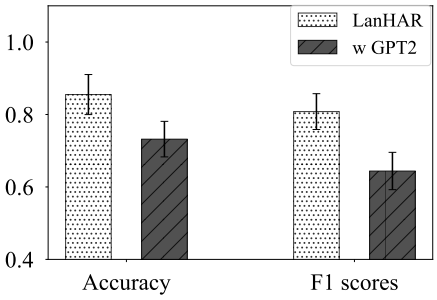


Fig. 13. The impact of different pre-trained Language Models as text encoders.

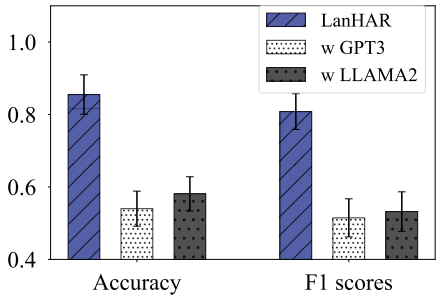


Fig. 14. The impact of different Large Language Models.

5.7.3 The Impact of Different L LMs. Given the varying capabilities of different LLMs, we explore how three LLMs (GPT-4, GPT-3, and LLAMA2) impact the interpretation of sensor readings and activity labels in HAR performance. As shown in Fig. 14, the performance of GPT-3 and LLAMA2 is noticeably lower than that of GPT-4, highlighting the importance of the LLM's capability in language-centered human activity recognition. These results suggest that stronger LLMs with advanced reasoning abilities are better suited for enhancing HAR performance.

5.7.4 The Impact of Different Response Qualities of LLMs. To ensure the quality of the LLM's interpretation of sensor readings, we (i) design specific prompts to guide the LLM's responses and (ii) develop an iterative re-generation method to filter out low-quality semantic interpretations and generate improved ones. To demonstrate the effectiveness of the automatic iterative re-generation method, we compare the relative KL divergence of the same activities across different datasets before and after applying the method. Specifically, for each activity, we measure the relative KL divergence between any two datasets both with and without applying the iterative re-generation method (i.e., "w iterative re-generation" and "w/o iterative re-generation"), as shown in Fig. 15. After applying the method, the average relative KL divergence for the same activity across different datasets decreases by 16.92%, indicating that the automatic iterative re-generation method significantly improves the response qualities of LLMs and reduces the distribution gap in heterogeneous data.

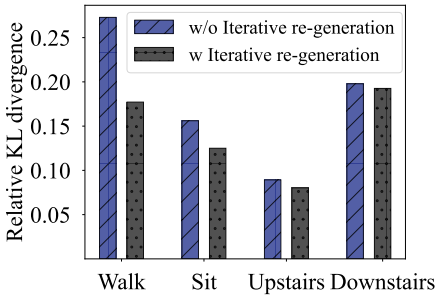


Fig. 15. Relative KL divergence with and without the iterative re-generation method.

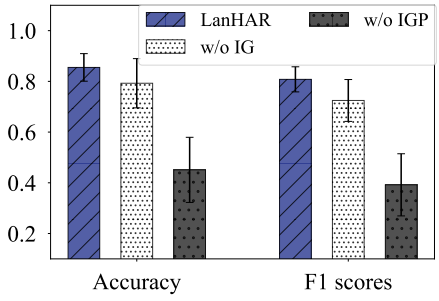


Fig. 16. The impact of different quality of LLMs' response.

To verify the impact of these methods on HAR performance, we compare our method with two variants: one using only simple prompts (w/o IGP) and another using only specifically designed prompts (w/o IG). As shown in Fig. 16, the performance when using only simple prompts is the lowest, indicating that LLMs require guidance to generate high-quality responses. The combination of specifically designed prompts and the automatic iterative re-generation method performs the best, demonstrating that our method effectively enhances the quality of LLM responses, leading to improved performance in human activity recognition.

5.8 Parameter Sensitivity

In our model, selecting the appropriate parameters for the sensor encoder is crucial, as it directly affects its suitability for deployment on mobile devices (e.g., ensuring the parameter size is sufficiently small). In this section, we analyze the impact of the embedding dimensions and the number of encoding layers on performance.

5.8.1 Effect of Different Number of Attention Heads. Fig. 17 shows the performance comparison $w.r.t.$ the number of attention heads in the sensor encoder. Overall, the performance is relatively stable under different numbers of attention heads. We choose the number of attention heads of 2 because it provides good performance while reducing the model's parameter size.

5.8.2 Effect of Different Number of Encoding Layers. Fig. 18 shows the performance comparison with respect to different numbers of encoding layers in the sensor encoder. As observed, a smaller number of encoding layers (e.g., 2 layers) results in insufficient representation, whereas a larger number (e.g., 6 layers) increases model complexity. We select 3 encoding layers as they strike a balance, delivering good performance without significantly increasing model complexity.

5.9 Deployment Analysis

To evaluate the suitability of the sensor encoder for mobile deployment, we analyze its parameter size and inference time. Deploying LanHAR on mobile devices involves three key steps: (i) developing an Android application, (ii) converting the model into a .pt file, and (iii) integrating it for processing accelerometer and gyroscope data for activity recognition. For our case study, we select the Pixel 7 smartphone as the deployment platform. The sensor encoder, combined with the activity label embeddings, occupies approximately 40 MB of memory, of which 0.01 MB is allocated for the embeddings. To assess its real-time performance, we conduct 100 inference trials, measuring the time required to recognize an activity. The results indicate an average processing time of 0.607 seconds, with the minimum and maximum times recorded at 0.716 seconds and 0.527 seconds, respectively. For comparison, we evaluate the model's inference efficiency on a single A5000 GPU with a batch

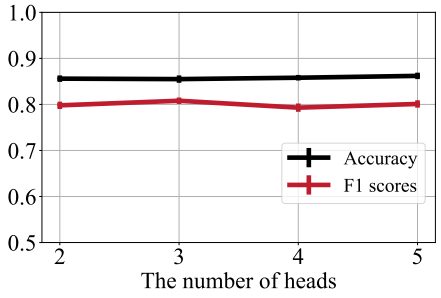


Fig. 17. The effect of different number of at-tention heads.

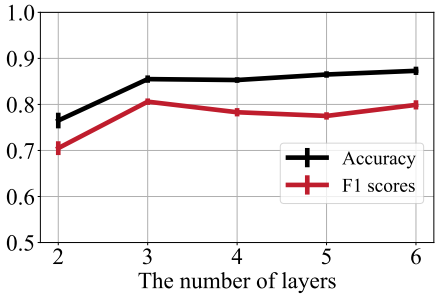


Fig. 18. The effect of different number of en-coding layers.

size of 1. The average inference time for a single activity is 0.0007 seconds, with the minimum and maximum times recorded at 0.0006 seconds and 0.001 seconds, respectively. Although inference on the smartphone is significantly slower than on the GPU, it remains within a practical range for real-world mobile applications, supporting real-time inference with an IMU sampling frequency of around 40 Hz (i.e., 0.025 seconds per reading). This demonstrates that LanHAR can operate effectively on mobile devices. To further enhance efficiency, future optimizations, such as quantization and hardware acceleration, could be explored.

6 DISCUSSION

Lessons learned: We summarize the following lessons to benefit future studies: (1) Introducing semantic interpretation can mitigate heterogeneity in cross-dataset HAR and offers the potential to recognize new activities, as indicated in Table 2 and Fig. 9. (2) The success of LLMs in cross-dataset HAR is partly due to their ability to summarize patterns and represent them using language. This leads to a lower KL divergence between cross-dataset data distributions compared to raw data. (3) Improving the quality of LLM responses through iterative re-generation, guided by the KL divergence, enhances HAR performance. The evaluation shows that generating high-quality responses has a direct positive impact on HAR accuracy.

Comparison to other ways of utilizing LLMs: In addition to our method, there are two other potential methods for utilizing LLMs in HAR: training a foundation model specifically for HAR or fine-tuning an existing open-source LLM, such as Llama. A key advantage of our method lies in its low cost and flexibility. On the one hand, our method does not require large amounts of data or computing resources for model training or fine-tuning. On the other hand, different LLMs can be flexibly integrated into our framework. As demonstrated in our evaluation, incorporating a high-quality LLM could further enhance the performance of our method.

Other ways of improving LLMs responses: There are ongoing efforts in the AI community aimed at improving the trustworthiness of LLMs through methods such as retrieval-augmented generation and memory-augmented models. Our method offers a general framework for post-hoc factuality verification, where advancements in these AI methods can be seamlessly integrated into our framework.

Limitations: (i) The process of obtaining semantic interpretations using LLMs is time-consuming, primarily due to the slow inference speed of current LLMs. We anticipate that future advancements in accelerating LLMs will enhance the efficiency of our method. (ii) Although we used four public datasets in the evaluation, they remain relatively small in scale with a limited number of activities. A larger dataset could provide a more comprehensive evaluation and further validate our method's effectiveness. (iii) Although our evaluation on PAMAP2 simulates domain heterogeneity by splitting participants, it does not constitute a true cross-dataset validation. This design choice was motivated by the need to assess model performance on complex activities (e.g., vacuum cleaning,

ironing), which are sparsely available and inconsistently defined across public datasets. Even though, we believe this setting is usable to evaluate the performance on complex activities when no additional datasets with the same complex activities are available.

7 CONCLUSION

In this work, we incorporate LLM-generated semantic interpretations of both sensor readings and activity labels to mitigate cross-dataset heterogeneity and to enable the recognition of unseen activities by mapping both sensor readings and activity labels into a shared semantic space. We design LanHAR, a lightweight system that features a semantic interpretation generation process with an iterative re-generation mechanism to ensure output quality. Our two-stage training and inference framework allows the transfer of LLM capabilities to mobile and resource-constrained environments, enabling practical deployment scenarios. We conduct comprehensive experiments across five public HAR datasets. The results demonstrate that our method consistently outperforms state-of-the-art approaches in cross-dataset activity recognition. In future work, we plan to extend our framework to support open-set activity recognition, integrate physics-informed semantic interpretations and handle more complex, fine-grained activity patterns to further improve robustness and generalizability.

Acknowledgments

This work was supported in part by the National Science Foundation under Grant No. 2246080, 2427915. Vinod Namboodiri was partially supported by the National Science Foundation under Grant No. 2345057. We sincerely thank the anonymous reviewers for their valuable comments and constructive suggestions, which have greatly helped improve the quality of this paper.

References

[1] Luca Arrotta, Claudio Bettini, Gabriele Civitarese, and Michele Fiori. 2024. Contextgpt: Infusing llms knowledge into neuro-symbolic activity recognition models. In *2024 IEEE International Conference on Smart Computing (SMARTCOMP)*. IEEE, 55–62.

[2] Daniel Auge, Julian Hille, Etienne Mueller, and Alois Knoll. 2021. A survey of encoding techniques for signal processing in spiking neural networks. *Neural Processing Letters* 53, 6 (2021), 4693–4710.

[3] Arnab Barna, Abdul Kadar Muhammad Masum, Mohammad Emdad Hossain, Erfanul Hoque Bahadur, and Mohammed Shamsul Alam. 2019. A study on human activity recognition using gyroscope, accelerometer, temperature and humidity data. In *2019 international conference on electrical, computer and communication engineering (ecce)*. IEEE, 1–6.

[4] Youngjae Chang, Akhil Mathur, Anton Isopoussu, Junehwa Song, and Fahim Kawsar. 2020. A systematic study of unsupervised domain adaptation for robust human-activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 1 (2020), 1–30.

[5] Jingcheng Chen, Yining Sun, and Shaoming Sun. 2021. Improving human activity recognition performance by data fusion and feature engineering. *Sensors* 21, 3 (2021), 692.

[6] Heng-Tze Cheng, Feng-Tso Sun, Martin Griss, Paul Davis, Jianguo Li, and Di You. 2013. Nuactiv: Recognizing unseen new activities using semantic attribute-based learning. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*. 361–374.

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

[8] Sannara Ek, François Portet, and Philippe Lalanda. 2022. Lightweight transformers for human activity recognition on mobile devices. *arXiv preprint arXiv:2209.11750* (2022).

[9] Sijie Ji, Xinzhe Zheng, and Chenshu Wu. 2024. HARGPT: Are LLMs Zero-Shot Human Activity Recognizers? *arXiv preprint arXiv:2403.02727* (2024).

[10] Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. 2023. Motiongpt: Human motion as a foreign language. *Advances in Neural Information Processing Systems* 36 (2023), 20067–20079.

[11] Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. 2024. Motiongpt: Human motion as a foreign language. *Advances in Neural Information Processing Systems* 36 (2024).

[12] Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. 2023. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728* (2023).

[13] Justin J Kavanagh and Hylton B Menz. 2008. Accelerometry: a technique for quantifying movement patterns during walking. *Gait & posture* 28, 1 (2008), 1–15.
[14] Md Abdullah Al Hafiz Khan, Nirmalya Roy, and Archan Misra. 2018. Scaling human activity recognition via deep learning-based domain adaptation. In *2018 IEEE international conference on pervasive computing and communications (PerCom)*. IEEE, 1–9.
[15] Y Kim, X Xu, D McDuff, C Breazeal, and HW Park. 2024. Health-LLM: large language models for health prediction via wearable sensor data, arXiv. *arXiv preprint arXiv:2401.06866* (2024).
[16] Solomon Kullback and Richard A Leibler. 1951. On information and sufficiency. *The annals of mathematical statistics* 22, 1 (1951), 79–86.
[17] Sunjae Lee, Junyoung Choi, Jungjae Lee, Munim Hasan Wasi, Hojun Choi, Steve Ko, Sangeun Oh, and Insik Shin. 2024. Mobilegpt: Augmenting llm with human-like app memory for mobile task automation. In *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking*. 1119–1133.
[18] Zikang Leng, Amitrajit Bhattacharjee, Hrudhai Rajasekhar, Lizhe Zhang, Elizabeth Bruda, Hyeokhyen Kwon, and Thomas Plötz. 2024. Imugpt 2.0: Language-based cross modality transfer for sensor-based human activity recognition. *arXiv preprint arXiv:2402.01049* (2024).
[19] Qiang Li, John A Stankovic, Mark A Hanson, Adam T Barth, John Lach, and Gang Zhou. 2009. Accurate, fast fall detection using gyroscopes and accelerometer-derived posture information. In *2009 sixth international workshop on wearable and implantable body sensor networks*. IEEE, 138–143.
[20] Vasiliki Liagkou, Sofia Sakka, and Chrysostomos Stylios. 2022. Security and Privacy Vulnerabilities in Human Activity Recognition systems. In *2022 7th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM)*. IEEE, 1–6.
[21] Wei Liu, Yi Ding, Shuai Wang, Yu Yang, and Desheng Zhang. 2022. Para-pred: Addressing heterogeneity for city-wide indoor status estimation in on-demand delivery. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 3407–3417.
[22] Wang Lu, Jindong Wang, Yiqiang Chen, Sinno Jialin Pan, Chunyu Hu, and Xin Qin. 2022. Semantic-discriminative mixup for generalizable sensor-based cross-domain activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 2 (2022), 1–19.
[23] Mohammad Malekzadeh, Richard G Clegg, Andrea Cavallaro, and Hamed Haddadi. 2019. Mobile sensor data anonymization. In *Proceedings of the international conference on internet of things design and implementation*. 49–58.
[24] Shenghuan Miao and Ling Chen. 2024. Goat: A generalized cross-dataset activity recognition framework with natural language supervision. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 8, 4 (2024), 1–28.
[25] Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. 2025. A comprehensive overview of large language models. *ACM Transactions on Intelligent Systems and Technology* 16, 5 (2025), 1–72.
[26] Friedrich Niemann, Stefan Lüdtke, Christian Bartelt, and Michael Ten Hompel. 2021. Context-aware human activity recognition in industrial processes. *Sensors* 22, 1 (2021), 134.
[27] Nurul Retno Nurwulan and Gjergji Selamaj. 2021. A comparative evaluation of acceleration and jerk in human activity recognition using machine learning techniques. In *Proceedings of the 1st International Conference on Electronics, Biomedical Engineering, and Health Informatics: ICEBEHI 2020, 8-9 October, Surabaya, Indonesia*. Springer, 55–61.
[28] Francisco Javier Ordóñez and Daniel Roggen. 2016. Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. *Sensors* 16, 1 (2016), 115.
[29] Riccardo Presotto, Sannara Ek, Gabriele Civitarese, François Portet, Philippe Lalanda, and Claudio Bettini. 2023. Combining public human activity recognition datasets to mitigate labeled data scarcity. In *2023 IEEE International Conference on Smart Computing (SMARTCOMP)*. IEEE, 33–40.
[30] Hangwei Qian, Tian Tian, and Chunyan Miao. 2022. What makes good contrastive learning on small-scale wearable-based tasks?. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*. 3761–3771.
[31] Xin Qin, Yiqiang Chen, Jindong Wang, and Chaohui Yu. 2019. Cross-dataset activity recognition via adaptive spatial-temporal transfer learning. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 4 (2019), 1–25.
[32] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. PMLR, 8748–8763.
[33] Attila Reiss. 2012. PAMAP2 Physical Activity Monitoring. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5NW2H.
[34] Jorge-L Reyes-Ortiz, Luca Oneto, Albert Samà, Xavier Parra, and Davide Anguita. 2016. Transition-aware human activity recognition using smartphones. *Neurocomputing* 171 (2016), 754–767.
[35] Aaqib Saeed, Tanir Ozcelebi, and Johan Lukkien. 2019. Multi-task self-supervised learning for human activity detection. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 2 (2019), 1–30.
[36] Muhammad Shoaib, Stephan Bosch, Ozlem Durmaz Incel, Hans Scholten, and Paul JM Havinga. 2014. Fusion of smartphone motion sensors for physical activity recognition. *Sensors* 14, 6 (2014), 10146–10176.

[37] Allan Stisen, Henrik Blunck, Sourav Bhattacharya, Thor Siiger Prentow, Mikkel Baun Kjærgaard, Anind Dey, Tobias Sonne, and Mads Møller Jensen. 2015. Smart devices are different: Assessing and mitigatingmobile sensing heterogeneities for activity recognition. In *Proceedings of the 13th ACM conference on embedded networked sensor systems*. 127–140.
[38] Abdulhamit Subasi, Kholoud Khateeb, Tayeb Brahimi, and Akila Sarirete. 2020. Human activity recognition using machine learning methods in a smart healthcare environment. In *Innovation in health informatics*. Elsevier, 123–144.
[39] Abdulhamit Subasi, Mariam Radhwan, Rabea Kurdi, and Kholoud Khateeb. 2018. IoT based mobile healthcare system for human activity recognition. In *2018 15th learning and technology conference (L&T)*. IEEE, 29–34.
[40] Han Sun and Yu Chen. 2022. Real-time elderly monitoring for senior safety by lightweight human action recognition. In *2022 IEEE 16th International Symposium on Medical Information and Communication Technology (ISMICT)*. IEEE, 1–6.
[41] Heng Tan, Hua Yan, and Yu Yang. 2025. LLM-Guided Reinforcement Learning: Addressing Training Bottlenecks through Policy Modulation. *arXiv preprint arXiv:2505.20671* (2025).
[42] Chi Ian Tang, Ignacio Perez-Pozuelo, Dimitris Spathis, Soren Brage, Nick Wareham, and Cecilia Mascolo. 2021. Selfhar: Improving human activity recognition through self-training with unlabeled data. *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies* 5, 1 (2021), 1–30.
[43] Terry T Um, Franz MJ Pfister, Daniel Pichler, Satoshi Endo, Muriel Lang, Sandra Hirche, Urban Fietzek, and Dana Kulić. 2017. Data augmentation of wearable sensor data for parkinson's disease monitoring using convolutional neural networks. In *Proceedings of the 19th ACM international conference on multimodal interaction*. 216–220.
[44] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).
[45] A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems* (2017).
[46] Aiguo Wang, Guilin Chen, Jing Yang, Shenghui Zhao, and Chih-Yung Chang. 2016. A comparative study on human activity recognition using inertial sensors in a smartphone. *IEEE Sensors Journal* 16, 11 (2016), 4566–4578.
[47] Jin Wang, Ronghua Chen, Xiangping Sun, Mary FH She, and Yuchuan Wu. 2011. Recognizing human daily activities from accelerometer signal. *Procedia Engineering* 15 (2011), 1780–1786.
[48] Jinqiang Wang, Tao Zhu, Jingyuan Gan, Liming Luke Chen, Huansheng Ning, and Yaping Wan. 2022. Sensor data augmentation by resampling in contrastive learning for human activity recognition. *IEEE Sensors Journal* 22, 23 (2022), 22994–23008.
[49] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems* 35 (2022), 24824–24837.
[50] Hao Wen, Yuanchun Li, Guohong Liu, Shanhui Zhao, Tao Yu, Toby Jia-Jun Li, Shiqi Jiang, Yunhao Liu, Yaqin Zhang, and Yunxin Liu. 2024. Autodroid: Llm-powered task automation in android. In *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking*. 543–557.
[51] Huatao Xu, Liying Han, Qirui Yang, Mo Li, and Mani Srivastava. 2024. Penetrative ai: Making llms comprehend the physical world. In *Proceedings of the 25th International Workshop on Mobile Computing Systems and Applications*. 1–7.
[52] Huatao Xu, Pengfei Zhou, Rui Tan, and Mo Li. 2023. Practically Adopting Human Activity Recognition. In *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*. 1–15.
[53] Huatao Xu, Pengfei Zhou, Rui Tan, Mo Li, and Guobin Shen. 2021. Limu-bert: Unleashing the potential of unlabeled data for imu sensing applications. In *Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems*. 220–233.
[54] Hua Yan, Heng Tan, Haotian Wang, Desheng Zhang, and Yu Yang. 2024. Robust route planning under uncertain pickup requests for last-mile delivery. In *Proceedings of the ACM Web Conference 2024*. 3022–3030.
[55] Hua Yan, Hao Wang, Desheng Zhang, and Yu Yang. 2023. Identifying regional driving risks via transductive cross-city transfer learning under negative transfer. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 2877–2886.
[56] Jianbo Yang, Minh Nhut Nguyen, Phyo Phyo San, Xiaoli Li, and Shonali Krishnaswamy. 2015. Deep convolutional neural networks on multichannel time series for human activity recognition.. In *Ijcai*, Vol. 15. Buenos Aires, Argentina, 3995–4001.
[57] Shuochao Yao, Shaohan Hu, Yiran Zhao, Aston Zhang, and Tarek Abdelzaher. 2017. Deepsense: A unified deep learning framework for time-series mobile sensing data processing. In *Proceedings of the 26th international conference on world wide web*. 351–360.
[58] Xiyuan Zhang, Ranak Roy Chowdhury, Jiayun Zhang, Dezhi Hong, Rajesh K Gupta, and Jingbo Shang. 2023. Unleashing the Power of Shared Label Structures for Human Activity Recognition. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 3340–3350.
[59] Xiyuan Zhang, Diyan Teng, Ranak Roy Chowdhury, Shuheng Li, Dezhi Hong, Rajesh K. Gupta, and Jingbo Shang. 2024. UniMTS: Unified Pre-training for Motion Time Series. arXiv:2410.19818 [eess.SP] https://arxiv.org/abs/2410.19818
[60] Xiaochen Zheng, Meiqing Wang, and Joaquín Ordieres-Meré. 2018. Comparison of data preprocessing approaches for applying deep learning to human activity recognition in the context of industry 4.0. *Sensors* 18, 7 (2018), 2146.
[61] Zhijun Zhou, Yingtian Zhang, Xiaojing Yu, Panlong Yang, Xiang-Yang Li, Jing Zhao, and Hao Zhou. 2020. Xhar: Deep domain adaptation for human activity recognition with smart devices. In *2020 17th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 1–9.

Appendix

Comparison to Baselines with Five Simple Activities

Following the reviewer's suggestion, we extend our cross-dataset experiments to include all relevant activities, treating sitting and standing as separate classes rather than merging them into a single still category. Specifically, the evaluated activities now include sitting, standing, walking, going upstairs, and going downstairs. The new cross-dataset results for five activities are as shown in Table 5.

Table 5. Overall performance with 5 simple activities (walking, going upstairs, going downstairs,sitting and standing). Bolded scores indicate the best, while underlined scores represent the second best.

Method	Source	HHAR			UCI			Motion			Shoaib			Average
	Target	UCI	Motion	Shoaib	HHAR	Motion	Shoaib	HHAR	UCI	Shoaib	HHAR	UCI	Motion	
DCNN	Accuracy	0.231	0.154	0.181	0.271	0.175	0.292	0.245	0.342	0.518	0.323	0.462	0.362	0.296
	F1 scores	0.162	0.091	0.084	0.163	0.107	0.251	0.092	0.246	0.486	0.208	0.405	0.218	0.209
Transformer	Accuracy	0.285	0.266	0.270	0.230	0.244	0.213	0.221	0.246	0.441	0.220	0.372	0.277	0.274
	F1 scores	0.195	0.198	0.205	0.078	0.158	0.216	0.072	0.182	0.416	0.072	0.287	0.105	0.182
LIMU-BERT	Accuracy	0.264	0.214	0.408	0.411	0.230	0.262	0.186	0.129	0.275	0.442	0.313	0.224	0.279
	F1 scores	0.150	0.139	0.350	0.381	0.075	0.191	0.076	0.064	0.165	0.412	0.234	0.073	0.193
UDAHAR	Accuracy	0.266	0.316	0.348	0.270	0.173	0.376	0.228	0.286	0.468	0.254	0.496	0.304	0.315
	F1 scores	0.128	0.215	0.250	0.125	0.104	0.339	0.074	0.206	0.447	0.137	0.436	0.155	0.218
SDMix	Accuracy	0.375	0.349	0.433	0.462	0.390	0.461	0.371	0.327	0.300	0.418	0.197	0.518	0.383
	F1 scores	0.244	0.273	0.370	0.395	0.245	0.441	0.300	0.281	0.172	0.386	0.135	0.435	0.306
UniMTS	Accuracy	0.439	0.457	0.482	0.441	0.402	0.511	0.407	0.437	0.514	0.492	0.402	0.553	0.461
	F1 scores	0.355	0.361	0.383	0.392	0.328	0.405	0.301	0.382	0.416	0.354	0.327	0.468	0.372
UniHAR	Accuracy	0.575	0.523	0.489	0.475	0.569	0.628	0.553	0.570	0.551	0.464	0.569	0.579	0.545
	F1 scores	0.470	0.399	0.413	0.421	0.489	0.586	0.457	0.510	0.480	0.397	0.491	0.501	0.467
LanHAR	Accuracy	**0.629**	**0.585**	**0.557**	**0.576**	**0.624**	**0.631**	**0.587**	**0.611**	**0.558**	**0.561**	**0.596**	**0.580**	**0.591**
	F1 scores	**0.510**	**0.461**	**0.447**	**0.443**	**0.528**	**0.588**	**0.502**	**0.527**	**0.484**	**0.492**	**0.512**	**0.505**	**0.497**