

OntoAR-HAR: Ontology-Guided Agentic RAG with Personalized Memory for Human Activity Recognition

Xiangqing Lu, Tao Zhu, *Member, IEEE*, Liang Yu, Shuangjian Li, Lulu Ban, Furong Duan, Mingxing Nie, Liming Chen, *Senior Member, IEEE*, and Yaping Wan

Abstract—LLMs increasingly support sensor-based human activity recognition (HAR) by linking motion signals with activity semantics and physical knowledge. Retrieval-augmented generation (RAG) grounds decisions in sensor examples and domain features, enabling training-free target-user inference with less dependence on raw numerical arrays. However, existing LLM/RAG-based HAR typically searches single-level activity spaces using static population knowledge. Consequently, they compare many unrelated candidates using the same static population evidence and overlook personalized motion patterns. We propose OntoAR-HAR, an ontology-guided agentic RAG framework combining static population and personalized knowledge. Its tree-structured activity ontology decomposes classification into local decisions among semantically related activities. The static store supplies node-constrained examples, while offline-discovered executable signal features provide numerical evidence for local distinctions. During inference, the personalized store collects selected user observations, which the LLM integrates with static knowledge and signal features while traversing the ontology to an activity label. Cross-subject and heterogeneous-sensing experiments show complementary gains from hierarchical retrieval and personalized memory across multiple LLM backbones, achieving state-of-the-art performance. By explicitly modeling hierarchical relations across activity granularities, OntoAR-HAR also addresses the long-overlooked semantic inclusion problem and avoids treating related labels as independent categories.

Index Terms—Human activity recognition, inertial sensors, retrieval-augmented generation, ontologies, personalization.

I. INTRODUCTION

Sensor-based human activity recognition (HAR) maps multichannel motion windows to activity labels for applications such as health monitoring, sports analytics, and mobile interaction. Recent LLM-based HAR has progressed from direct prompting toward retrieval-grounded inference. HARGPT studies whether an LLM can select activities from activity descriptions and sensor sequences through direct prompting [1]. Following the broader RAG paradigm [2], RAG-HAR retrieves sensor observations and statistical descriptors as contextual evidence [3], while ZARA combines retrieved motion examples with domain knowledge in an LLM-based agent [4].

(Corresponding author: Tao Zhu.)

Xiangqing Lu, Tao Zhu, Liang Yu, Mingxing Nie, and Yaping Wan are with the School of Computer Science, University of South China, Hengyang, China (e-mail: xqlu@stu.usc.edu.cn; tzhu@usc.edu.cn; lyu@stu.usc.edu.cn; niemx@usc.edu.cn; ypw@usc.edu.cn).

Shuangjian Li and Liming Chen are with the School of Computer Science and Technology, Dalian University of Technology, Dalian, China (e-mail: shuangjianli@mail.dlut.edu.cn; limingchen0922@dlut.edu.cn).

Lulu Ban is with Hunan University, Changsha, China (e-mail: llban@hnu.edu.cn).

Furong Duan is with the School of Computing, Ulster University, Belfast, United Kingdom (e-mail: duan-f@ulster.ac.uk).

These approaches demonstrate the main advantage of RAG for HAR: observed motion patterns and signal knowledge can be placed directly in the decision context, reducing reliance on an LLM’s interpretation of raw numerical arrays alone. Here, RAG denotes retrieval-augmented LLM inference whose final output is an activity label. Its effectiveness depends not only on embedding similarity, but also on how activity candidates and retrieved evidence are organized.

Current RAG-style HAR systems commonly organize activity labels and their examples in a single-level retrieval space. This is problematic when labels express semantic inclusion or share a physical concept. For example, upstairs and downstairs are both forms of walking; sitting, standing, and lying are stationary activities; and eating soup and eating chips share an eating motion but differ in local wrist dynamics. Retrieving across all labels at once introduces candidates that are irrelevant to the current distinction and makes it difficult to associate a physical feature with the local decision where it is useful. Moreover, a static population store contains only source-user sensor examples and is not supplemented with observations from the current user, so it cannot represent that user’s characteristic motion amplitude, rhythm, or execution style.

We propose **OntoAR-HAR**, an ontology-guided agentic RAG framework for HAR that combines a static population knowledge store with a personalized knowledge store. The ontology represents inclusion and sibling relations among activity labels and turns one global retrieval problem into a sequence of local decisions. Static population retrieval evidence and executable physical features are attached to each decision node. At inference time, the separate personalized store records selected observations from the current user. Together, the two stores provide similar sensor examples retrieved from population data and the current user’s previous observations. An LLM integrates these examples with numerical feature reports and selects the next ontology branch until it reaches an activity leaf. The framework uses a labeled source collection, a pretrained sensor encoder, and offline feature construction; target inference does not update model parameters.

The contribution of this paper is a structured agentic RAG framework that combines an activity ontology with a personalized knowledge store for HAR. Ontology-guided retrieval limits each reasoning step to a semantically coherent candidate group, addressing activity-label inclusion and local ambiguity, while personalized retrieval complements static population activity knowledge with user-specific observations. We evaluate this design on eight public HAR datasets through cross-subject recognition, component ablations, and heterogeneous sensing

conditions.

II. RELATED WORK

a) Representation learning for HAR.: Convolutional–recurrent models such as DeepConvLSTM [5] established end-to-end sensor representation learning. Self-supervised methods learn transferable temporal structure through contrastive or reconstruction objectives [6]–[9]. Cross-domain HAR further uses adversarial alignment, language supervision, and large-scale motion pretraining [10]–[13]. These models provide strong learned embeddings; OntoAR-HAR uses such an encoder for retrieval and focuses on how retrieved evidence is organized and combined with activity semantics.

The progression from channel-aware supervised models to broadly pretrained encoders is relevant to retrieval quality. Attend and Discriminate weights informative sensor channels when learning an activity representation [14]; self-supervised learning has also been studied at the scale of hundreds of thousands of person-days of wearable recordings [15]. UniMTS uses unified motion pretraining [13], while One Model to Fit Them All uses language-assisted cross-dataset representations [16]. MobHAR instead studies source-free transfer to mobile devices [17]. These methods address the representation or transfer stage. For a retrieval-grounded decision, a useful embedding is necessary but does not specify which neighboring labels should be compared, which physical feature resolves their ambiguity, or how previous observations from the current person enter the comparison. Our method keeps the sensor encoder fixed and addresses these decisions in the retrieval and reasoning stages.

b) Language and retrieval for HAR.: HARGPT studies direct prompting with activity descriptions and sensor sequences [1]. SensorLLM and SensorLM learn interfaces between continuous signals and language models [18], [19]. Retrieval-augmented HAR instead supplies similar observations or feature knowledge as context [3], [4]. OntoAR-HAR differs in organizing retrieval by an activity ontology and maintaining separate static population and user-specific stores. Related neuro-symbolic HAR constrains learning through semantic knowledge [20]; our framework uses executable signal features as numerical evidence within ontology-guided retrieval.

Other approaches bring motion and language into a common representation before classification. IMU2CLIP aligns inertial signals with language-grounded representations [21], and COMODO transfers information from video to IMU data through cross-modal distillation [22]. ZeroHAR uses sensor context for zero-shot wearable action recognition [23]. These approaches make semantic information available to a predictor, whereas an evidence-grounded LLM must additionally decide how to use a retrieved example when its nearest label is only a partial semantic match. Flat retrieval presents all retrieved labels together. Our tree expresses the broader activity first and then associates feature reports with the specific sibling distinction under consideration. This matters for labels such as *walking*, *upstairs*, and *downstairs*, where a similarity match at one granularity need not settle the finer decision.

c) Adaptation and memory.: Test-time adaptation methods such as TENT and CoTTA update model parameters or statistics on target streams [24], [25]. OntoAR-HAR does not perform such updates. Its personalized knowledge base is an episodic retrieval mechanism: selected observations become additional context for later decisions. We therefore describe this component as *personalization*, not parameter-based test-time adaptation.

This distinction changes the unit of adaptation. Parameter adaptation changes the mapping from a sensor window to an embedding or label; episodic personalization leaves that mapping fixed and changes the reference examples available to the LLM. The latter can reflect a user’s recurring movement pattern without altering the static population store. It also introduces a selection problem: a mistaken pseudo-label could become a future reference. For this reason, the personal store is populated only after an evidence check, and population retrieval remains available at every later decision. Our ablation separates the effect of the hierarchy from the additional effect of this memory.

III. METHOD

A. Problem Formulation and Overview

Let the labeled source collection be $\mathcal{D}_s = \{(x_i, y_i, b_i)\}$, where x_i is a multichannel sensor window, $y_i \in \mathcal{Y}$ is an activity label, and b_i identifies a sensor view or body location. At inference, the system observes an ordered stream $x_1^q, x_2^q, \dots$ from a held-out user and predicts labels without gradient updates. Activity labels are organized as a rooted tree $\mathcal{T} = (\mathcal{V}, \mathcal{E})$. Leaves correspond to $\mathcal{Y}$ and internal nodes represent semantic groups. A decision at node v considers only $\text{child}(v)$.

OntoAR-HAR has an offline construction phase (Figure 1) and an online inference phase (Figure 2). Offline construction defines the ontology, builds node-constrained vector indexes, and associates executable features with local decisions. Online inference recursively retrieves static population and personalized examples, constructs numerical evidence, and asks an LLM to choose one child at a time.

B. Ontology and Feature Knowledge Construction

a) Confusion-guided ontology construction.: We manually construct the activity hierarchy from retrieval-only predictions on class-balanced activity samples from the training set. We first collect the confusion matrix and place activities with clear confusion relationships into coarse groups, each of which becomes a child of the root. A group may contain several activity labels. We then inspect the confusion submatrix within each group. If a subset is more strongly confused with one another than with the remaining members, we insert an intermediate node above that subset. For example, within a five-activity group, activities 1, 4, and 5 form a nested subgroup when their mutual confusions are notably stronger than their confusions with activities 2 and 3. We repeat this inspection within the resulting subgroups until no further clear subdivision is found. Every activity remains a single leaf, and the completed hierarchy is fixed before evaluation.

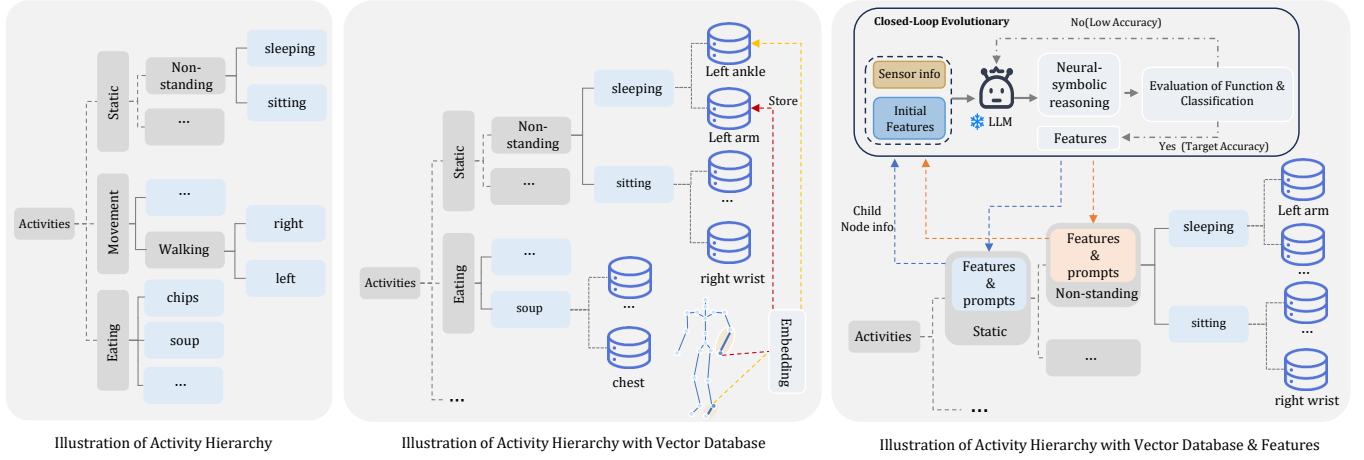


Fig. 1. Offline construction. The activity hierarchy is manually grouped and refined using retrieval-only confusions on balanced training samples. Source embeddings form the static population store; an LLM iteratively proposes signal features whose executable implementations are evaluated on confused examples.

TABLE I
REPRESENTATIVE SIGNAL-FEATURE FAMILIES CONSIDERED DURING OFFLINE CONSTRUCTION. THE RULE-BASED GENERATOR IMPLEMENTS SUPPORTED CANDIDATES; THE VALIDATED FEATURE SET IS SPECIFIC TO EACH DECISION NODE.

Feature family	Representative statistics or derived quantities
Time-domain statistics	Mean, standard deviation, variance, maximum, minimum, median, root mean square, peak, range, interquartile range, skewness, kurtosis, and signal magnitude area.
Frequency-domain statistics	Low-, mid-, and high-frequency band powers and their ratios; dominant frequency and power; secondary peak frequency; spectral centroid, entropy, skewness, and kurtosis; weighted mean frequency; and spectral energy.
Short-time Fourier transform	Maximum, mean, and standard deviation in each frequency band, plus statistics of short-time spectral entropy and spectral centroid.
Change and impact	Mean, root mean square, and standard deviation of first differences; zero-crossing rate; impact rate of change; and peaks.
Periodicity and dependence	First autocorrelation peak, first minimum, and first zero crossing; fourth-order autoregressive coefficients; and recurrence rate.
Multiscale complexity	Wavelet-packet energy and permutation entropy.
Posture and inter-axis relations	Gravity-direction statistics and inter-axis Pearson correlation coefficients.

The confusion matrix is a design aid rather than an automatically optimized tree objective. Balancing the training examples prevents a frequent class from dominating the observed error pairs. For each proposed internal node, we check that its descendants form a meaningful activity group and that the split gives a local comparison the system can actually make. A class belongs to exactly one branch at a given level; an intermediate node groups leaves without adding a new output label. The resulting tree therefore preserves the original classification task while changing the order in which candidate sets are considered. The hierarchy is built from source training data only and is not revised using the held-out inference users.

b) Iterative feature construction.: At each ambiguous node, an LLM proposes a structured list of feature names, physical interpretations, and calculations from the child labels and sensor metadata. Table I summarizes representative candidate families. A rule-based generator implements supported proposals. On examples misclassified by retrieval-only inference, we test execution and measure separation by mean standardized feature distance to retrieved activity

references. Errors and remaining misclassifications inform the next prompt. Algorithm 1 specifies the loop and retained outputs. Each internal node v stores $\Theta_v = (\mathcal{F}_v, \pi_v)$, where $\mathcal{F}_v$ is the selected executable feature set and π_v describes the local decision.

This construction connects a feature to a specific ambiguity. For example, a distinction between stationary postures calls for different evidence from a distinction between level walking and stair motion. Candidate descriptions alone are insufficient: the proposal must be translated into an executable calculation for the available sensor channels. The loop first checks whether that calculation runs on the hard examples, then uses retrieval-only misses to identify where its reported values remain uninformative. Its feedback contains both execution failures and residual decision errors, so a later proposal can revise the feature definition rather than merely restate the activity names. The generated functions and their node-specific descriptions are fixed before target-user inference.

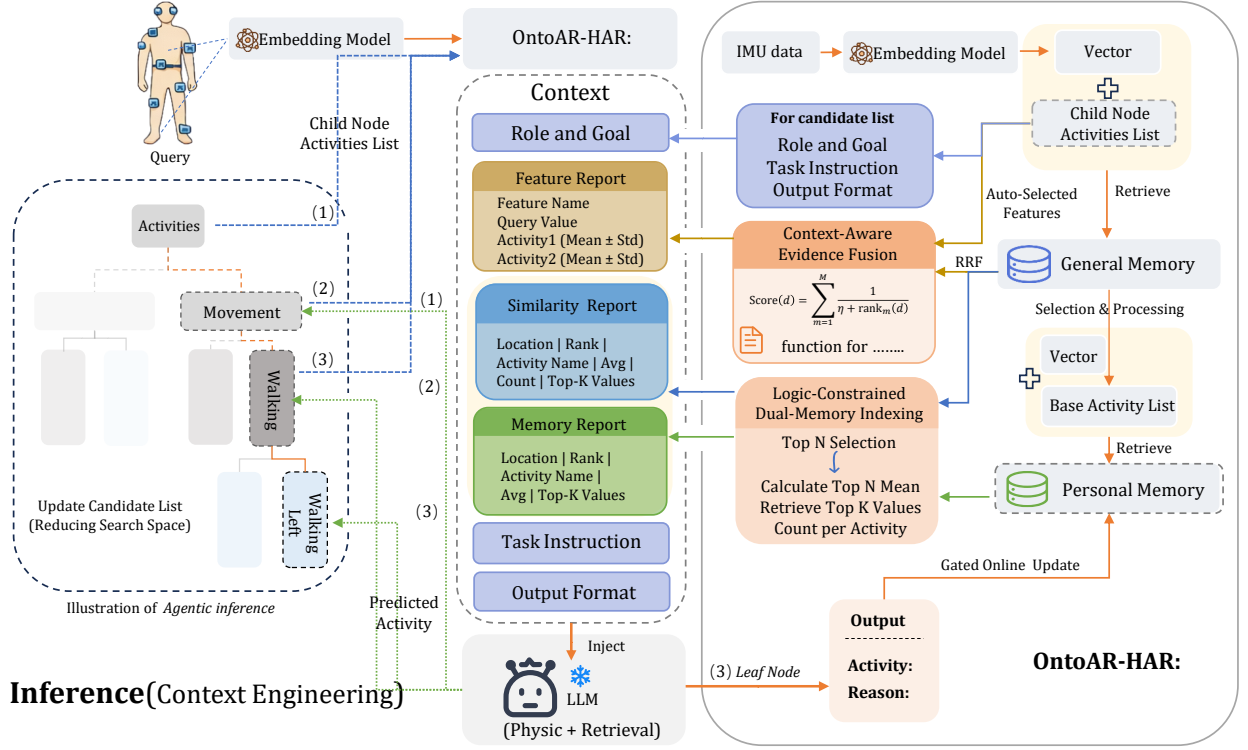


Fig. 2. Online inference. At each ontology node, retrieval is restricted to the current subtree. Static population examples, personalized examples, and executable-feature reports form the decision context; the selected child determines the next local problem.

Algorithm 1: Iterative feature generation at node v

Input: Child labels, sensor metadata, retrieval-only errors H .
1 prompt := InitialPrompt(labels, sensors); history := empty.
2 **for** round = 1 to 5 **do**
3 features := ParseJSON(LLM(prompt)); **retry** if invalid.
4 function := GenerateByRules(features); **retry** if unloadable.
5 errors := Execute(function, H).
6 accuracy, misses := DistanceProxy(function, first 20 of H).
7 Record features, function, errors, and accuracy.
8 **if** no errors and (accuracy exceeds 0.8 or is unavailable), **break**.
9 prompt := FeedbackPrompt(features, errors, misses).
10 **end for**; save history and the last generated function.

C. Static Population and Personalized Knowledge Stores

For each sensor view, the *static population store* $\mathcal{D}_g$ contains source embeddings and labels. A logical mask restricts retrieval at node v to examples whose labels occur below v :

$$\mathcal{R}_g(x_q, v) = \text{topk}\{s(e_q, e_i) : y_i \in \text{leaf}(v)\}, \quad (1)$$

where e_q and e_i are encoder embeddings and s is cosine similarity. The indexes are physically shared but logically filtered, avoiding a separate database for every node.

The *personalized store* $\mathcal{D}_p$ has the same representation but is initialized empty for each inference user. It contains previously observed query windows paired with accepted predictions. Retrieval first obtains the active labels from $\mathcal{D}_g$ and then searches $\mathcal{D}_p$ under the same label and node constraints. The two stores remain separate so that static population evidence is always visible and personal examples cannot replace the source reference distribution.

This order makes the static collection an anchor for the current local decision. The source search identifies plausible

labels under the current ontology node; a subsequent personal search asks whether accepted windows from this user support any of those labels. When no personal example satisfies the constraints, the decision proceeds with source examples and feature evidence. As the personal store grows, each accepted window remains associated with the user and its predicted leaf, so a later branch search can retrieve it only where that leaf is relevant. This avoids treating a user's history as an unrestricted global neighbor list.

For M sensor views, ranked lists are combined using reciprocal rank fusion (RRF) [26]:

$$\text{RRF}(d) = \sum_{m=1}^M \frac{1}{\eta + \text{rank}_m(d)}, \quad (2)$$

with $\eta = 60$. The retrieved examples define reference distributions for features in $\mathcal{F}_v$. OntoAR-HAR reports the query value, reference mean and standard deviation, and standardized deviation for each child. This makes the LLM compare explicit numerical evidence rather than calculate signal statistics from raw arrays.

Rank fusion is useful here because embeddings from different sensor views may have different score ranges: it combines their ordering without requiring raw similarity scores to be on the same scale. The feature report performs a different task. For a candidate child, it summarizes how the query's executable feature values compare with retrieved examples associated with that child. A large standardized deviation flags a mismatch for inspection, while the population and personal examples retain their separate provenance. Thus, view

TABLE II

CROSS-SUBJECT RESULTS (%). ACC AND MACRO-F1 ARE SHOWN IN SEPARATE ROWS. THE DATASET DESCRIPTORS GIVE THE NUMBER OF ACTIVITY CLASSES AND SENSOR CHANNELS; ONTOAR-HAR REPORTS THE MEAN WITH STANDARD DEVIATION AS A SUBSCRIPT OVER TEN RANDOM SAMPLES. BEST RESULTS ARE BOLD, AND SECOND-BEST RESULTS ARE UNDERLINED.

Method	Metric	Easy			Medium			Hard		Avg.
		Opp.	UCI-HAR	Shoaib	PAMAP2	USC-HAD	Mhealth	WISDM	DSADS	
# Classes		4	6	7	12	12	12	18	19	–
# Channels		30	6	30	18	6	15	6	30	–
<i>Cross-modal alignment (full-shot)</i>										
ImageBind	Acc	82.8	90.9	98.5	94.3	52.1	82.9	77.2	90.0	83.6
	F1	84.6	90.7	98.5	94.6	52.2	83.2	76.8	89.3	83.7
IMU2CLIP	Acc	71.9	87.1	98.2	90.7	55.7	94.5	62.7	88.7	81.2
	F1	70.0	87.0	<u>98.1</u>	91.2	52.0	95.0	62.8	88.2	80.5
<i>Self-supervised representation learning (full-shot)</i>										
TST	Acc	89.1	91.6	73.5	88.9	66.2	86.0	70.2	80.0	80.7
	F1	91.1	91.4	72.3	89.8	60.9	83.8	69.1	76.1	79.3
TARNet	Acc	82.6	91.1	67.9	92.4	57.6	82.9	58.5	30.7	70.5
	F1	83.4	91.2	66.1	90.6	48.7	77.5	58.5	23.0	67.4
TS2Vec	Acc	80.7	92.1	87.4	91.8	56.1	93.3	69.5	84.3	81.9
	F1	83.3	<u>92.1</u>	87.2	92.9	53.6	93.8	68.8	82.3	81.8
<i>Supervised and pretrained HAR models (full-shot)</i>										
THAT	Acc	83.1	86.8	92.3	96.5	53.9	89.0	60.3	82.1	80.5
	F1	84.5	86.7	92.2	96.7	55.0	89.9	60.9	79.0	80.6
IMUGPT	Acc	84.8	87.0	61.7	85.6	41.1	71.3	67.8	71.9	71.4
	F1	84.9	87.0	61.8	83.8	42.4	63.6	67.4	71.0	70.2
TimesNet	Acc	80.0	88.5	82.4	88.0	58.3	81.7	67.4	77.3	77.9
	F1	82.4	88.6	82.6	88.0	48.7	79.1	67.1	78.5	76.9
GPT4TS	Acc	83.6	88.4	73.5	92.6	61.7	87.8	63.8	79.9	78.9
	F1	86.1	88.5	74.1	93.4	58.9	85.2	64.5	78.0	78.6
SHARE	Acc	89.0	<u>92.2</u>	77.5	94.1	72.9	97.0	77.0	92.0	86.5
	F1	90.2	<u>92.1</u>	78.0	<u>94.8</u>	66.7	97.1	76.5	91.9	85.9
UniMTS	Acc	89.1	91.5	98.5	<u>96.2</u>	56.5	<u>96.3</u>	<u>80.5</u>	<u>92.9</u>	<u>87.7</u>
	F1	90.1	91.7	98.5	96.7	61.6	<u>96.7</u>	<u>79.9</u>	<u>92.6</u>	88.5
<i>Retrieval-based inference</i>										
ZARA	Acc	92.5	90.0	97.1	76.7	60.0	86.3	65.6	84.2	81.6
	F1	92.5	90.0	97.1	76.9	60.1	86.1	64.1	84.4	81.4
OntoAR-HAR _{Gemini 2.0 Flash}	Acc	85.7 $\pm$ 2.4	90.1 $\pm$ 5.3	96.8 $\pm$ 2.0	82.6 $\pm$ 1.5	<u>74.8</u> $\pm$ 3.2	74.6 $\pm$ 2.9	77.9 $\pm$ 1.8	89.8 $\pm$ 3.4	84.0 $\pm$ 1.1
	F1	85.5 $\pm$ 2.4	90.9 $\pm$ 4.1	96.8 $\pm$ 2.0	83.2 $\pm$ 1.6	<u>75.2</u> $\pm$ 3.3	74.0 $\pm$ 3.2	77.2 $\pm$ 1.9	89.2 $\pm$ 3.5	84.0 $\pm$ 0.8
OntoAR-HAR _{Gemini 2.5 Flash}	Acc	89.8 $\pm$ 2.3	89.6 $\pm$ 5.4	94.5 $\pm$ 4.3	84.5 $\pm$ 1.7	70.7 $\pm$ 2.2	83.9 $\pm$ 3.0	77.0 $\pm$ 2.4	90.9 $\pm$ 2.4	85.1 $\pm$ 1.7
	F1	89.9 $\pm$ 2.3	89.5 $\pm$ 5.4	93.4 $\pm$ 4.8	85.2 $\pm$ 2.1	70.8 $\pm$ 2.2	83.6 $\pm$ 3.0	76.2 $\pm$ 2.4	90.4 $\pm$ 2.7	84.9 $\pm$ 1.7
OntoAR-HAR _{Gemini 3.0 Flash}	Acc	<u>91.3</u> $\pm$ 1.6	93.1 $\pm$ 5.7	96.4 $\pm$ 2.8	86.7 $\pm$ 1.1	80.2 $\pm$ 3.8	84.0 $\pm$ 4.1	80.7 $\pm$ 3.0	93.2 $\pm$ 3.8	88.2 $\pm$ 2.3
	F1	<u>91.5</u> $\pm$ 1.6	92.8 $\pm$ 5.9	96.5 $\pm$ 2.9	87.3 $\pm$ 1.1	80.1 $\pm$ 3.4	84.6 $\pm$ 2.6	80.2 $\pm$ 3.3	92.8 $\pm$ 4.3	<u>88.2</u> $\pm$ 2.0

fusion determines which examples enter the context, and the numerical report helps interpret those examples at the active ontology node.

D. Recursive Inference and Memory Update

At node v , the context $\mathcal{C}_v$ contains the local instruction π_v , similar examples retrieved and fused from $\mathcal{D}_g$ and $\mathcal{D}_p$, and the feature report. The LLM returns a child and a short evidence record:

$$\hat{c}_v = \arg \max_{c \in \text{child}(v)} P_{\text{LLM}}(c | \mathcal{C}_v). \quad (3)$$

The agent moves to $\hat{c}_v$ until reaching a leaf $\hat{y}$. This coarse-to-fine path directly represents cases such as activity $\rightarrow$ locomotion $\rightarrow$ walking direction, so a feature is evaluated only where it distinguishes siblings.

After prediction, a prompt-based gate decides whether $(x_q, \hat{y})$ enters $\mathcal{D}_p$. The gate checks agreement between static population retrieval, personal retrieval when available, and the node feature ranges; ambiguous or contradictory cases are not inserted. Subsequent samples from the same user can retrieve

accepted episodes, allowing the context to reflect recurring personal motion patterns without changing encoder or LLM weights.

IV. EXPERIMENTS

A. Setup

We use Opportunity [27], UCI-HAR [28], Shoaib [29], PAMAP2 [30], USC-HAD [31], Mhealth [32], WISDM [33], and DSADS [34]. These datasets span 4–19 activities, 6–30 channels, and waist, wrist, chest, ankle, arm, leg, and pocket placements. Database subjects and inference subjects are disjoint. In the cross-subject evaluation, inference subsets are sampled without replacement and balanced by activity and user; their sizes are 200, 240, 210, 240, 240, 240, 288, and 190 windows, respectively. Cross-subject results use seeds 0, 1, 42, 123, 520, 527, 1024, 2026, 2778, and 3407. Accuracy measures leaf-label correctness, while macro-F1 gives equal weight to the per-class F1 scores.

The eight datasets vary in more than label count. Their sensors cover different body positions and channel configu-

TABLE III
SEMANTIC-INCLUSION EVALUATION. THE UPPER BLOCK REPORTS DATASET-LEVEL HIERARCHICAL F1; THE LOWER BLOCK GIVES OVERALL METRICS. PERCENTAGE CHANGES ARE IN POINTS.

Dataset	Hierarchical F1 (%)			Single	Hier.	Δ
	Single	Hier.	Δ Dataset			
DSADS	95.26	96.05	+0.79	Shoaib	99.29	99.52 +0.23
Mhealth	89.31	91.04	+1.73	UCI-HAR	96.67	97.50 +0.83
Opp.	89.50	92.75	+3.25	USC-HAD	78.54	81.11 +2.57
PAMAP2	89.03	89.58	+0.55	WISDM	81.77	82.99 +1.22
Avg.	89.92	91.32	+1.40			
Acc. (%)	84.24	87.17	+2.92	Macro-F1 (%)	84.36	87.10 +2.75
Tree dist.	0.428	0.366	-0.062	Cross-br. (%)	5.18	4.96 -0.22

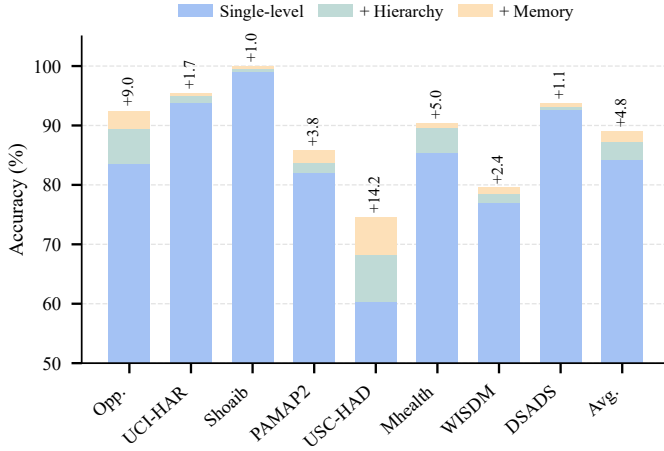


Fig. 3. Controlled component study with Gemini 2.5 Flash. Each vertical bar starts from single-level retrieval; the upper segments show the incremental gains from hierarchy and personalized memory.

rations, so nearest-neighbor retrieval must operate over different signal views. The cross-subject setting tests whether source-user examples remain useful for a person excluded from the database. Balancing the inference subset by both activity and user keeps the aggregate from being driven by the most common recorded action or the most heavily sampled participant. The controlled component comparison uses the cross-subject setting and Gemini 2.5 Flash, isolating the effect of the decision structure and memory within that evaluation configuration.

a) Signal windows and subject separation.: We retain the segmentation used for each dataset rather than forcing every source into the same number of time steps. Opportunity uses non-overlapping 2-s windows of 60 samples; UCI-HAR uses its 2.56-s, 128-sample windows with 50% overlap. Shoaib and Mhealth use non-overlapping 2-s windows of 100 samples, while PAMAP2 and USC-HAD use 2-s windows of 200 samples. WISDM uses 10-s windows of 200 samples after timestamp alignment of smartwatch accelerometer and gyroscope recordings; DSADS supplies predefined 5-s windows of 125 samples. These choices explain the different channel/window sizes reported in Table VI.

The source and inference partitions are disjoint by subject. Opportunity reserves subject 4; Shoaib reserves subjects 1 and

9; PAMAP2 reserves 5 and 6; USC-HAD reserves 13 and 14; Mhealth reserves 1 and 6; and DSADS reserves 2 and 4. UCI-HAR retains its predefined nine-subject test partition. For WISDM, eight of the 47 users with aligned smartwatch recordings are set aside for inference. The stated inference sizes result from class-balanced sampling within these held-out partitions, with the available examples per user also accounted for. This separation keeps a user’s earlier accepted query windows distinct from the source examples used to initialize the static store.

Baselines cover cross-modal models (ImageBind [35] and IMU2CLIP [21]), self-supervised temporal models (TST [36], TARNet [37], and TS2Vec [8]), supervised or pretrained HAR models (THAT [38], IMUGPT [39], TimesNet [40], GPT4TS [41], SHARE [42], and UniMTS [13]), and retrieval-based ZARA [4]. These methods use different supervision regimes, so Table II labels full-shot and retrieval settings rather than treating them as identical. OntoAR-HAR uses Mantis embeddings and temperature zero for LLM calls. Cross-view feature-reference fusion uses the standard RRF constant $\eta = 60$ and retains at most 100 fused items; these settings are fixed across datasets and LLM backbones rather than tuned per dataset. Experiments run on Windows 11 with an Intel Core i5-12600KF CPU, 32 GB RAM, and an NVIDIA GeForce RTX 4060 Ti GPU (8 GB), using Python 3.9, PyTorch 2.8, NumPy 1.26.4, and scikit-learn 1.6.1; LLM inference uses hosted APIs.

b) Baseline implementation.: The cross-modal full-shot baselines use a downstream classifier with their sensor embeddings. ImageBind starts from the pretrained `imagebind_huge` checkpoint and tunes the backbone and linear head; IMU2CLIP is first pretrained on Ego4D because official pretrained weights are unavailable, then its encoder and classifier are tuned for the target task. Among the temporal baselines, TST is pretrained before downstream fine-tuning, TARNet jointly optimizes reconstruction and classification, and TS2Vec learns contrastive representations before a dataset-specific linear readout. The direct-prompt HARGPT input is downsampled to 10 Hz to fit its prompt setting. For the zero-shot setting, IMUGPT uses a DeepConvLSTM backbone without its supervised distribution-calibration phase. ZARA follows its default pairwise feature knowledge setup with temperature zero. These details explain why the supervised and retrieval rows of Table II are informative comparisons but do not share an identical training budget.

B. Cross-Subject Results and Component Analysis

Table II reports 88.2 average accuracy and F1 for OntoAR-HAR with Gemini 3.0 Flash, versus 87.7/88.5 for UniMTS and 81.6/81.4 for ZARA. UniMTS is a demanding but non-equivalent reference because it uses full-shot optimization, whereas OntoAR-HAR performs retrieval-based inference without task-specific or target-user parameter updates; ZARA is the closer protocol match. OntoAR-HAR is strongest on USC-HAD (80.2 accuracy versus ZARA’s 60.0) and competitive on WISDM and DSADS, while UniMTS remains stronger on PAMAP2 and Mhealth. ZARA slightly leads on

TABLE IV
HETEROGENEOUS-DATASET RESULTS (%) OVER SIX HARMONIZED ACTIVITY CATEGORIES WITHOUT BODY-LOCATION FILTERING. ACC AND MACRO-F1 ARE SHOWN SIDE BY SIDE. BEST RESULTS ARE BOLD, AND SECOND-BEST RESULTS ARE UNDERLINED.

Method	UCI-HAR		PAMAP2		USC-HAD		Mhealth		DSADS		Average	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1
ImageBind	14.1	5.0	14.6	8.1	11.8	7.3	7.9	1.7	2.1	1.2	10.1	4.7
IMU2CLIP	17.8	11.4	1.9	1.1	12.8	9.3	7.9	1.3	4.6	1.1	9.0	4.8
IMUGPT	1.1	0.3	8.9	1.5	6.0	6.9	9.8	2.5	7.5	2.0	6.7	2.6
HARGPT	15.0	12.7	11.1	2.1	9.5	3.6	10.4	7.4	5.8	3.4	10.4	5.8
ZARA	16.7	4.9	17.1	8.6	16.7	4.8	20.0	6.7	16.7	6.8	17.4	6.4
UniMTS	35.2	22.0	47.2	43.6	30.5	27.8	70.7	61.8	31.5	23.7	43.0	35.8
OntoAR-HAR _{Gemini 2.0}	45.83	41.63	54.29	46.63	41.67	38.57	50.00	36.71	36.67	22.66	45.69	37.24
OntoAR-HAR _{Gemini 2.5}	<u>48.33</u>	45.26	<u>56.43</u>	<u>51.96</u>	<u>36.67</u>	35.04	50.00	36.92	41.67	32.37	<u>46.62</u>	<u>40.31</u>
OntoAR-HAR _{Gemini 3.0}	49.17	<u>43.94</u>	60.00	57.49	41.67	<u>37.61</u>	<u>51.00</u>	<u>39.64</u>	41.67	<u>31.87</u>	48.70	42.11

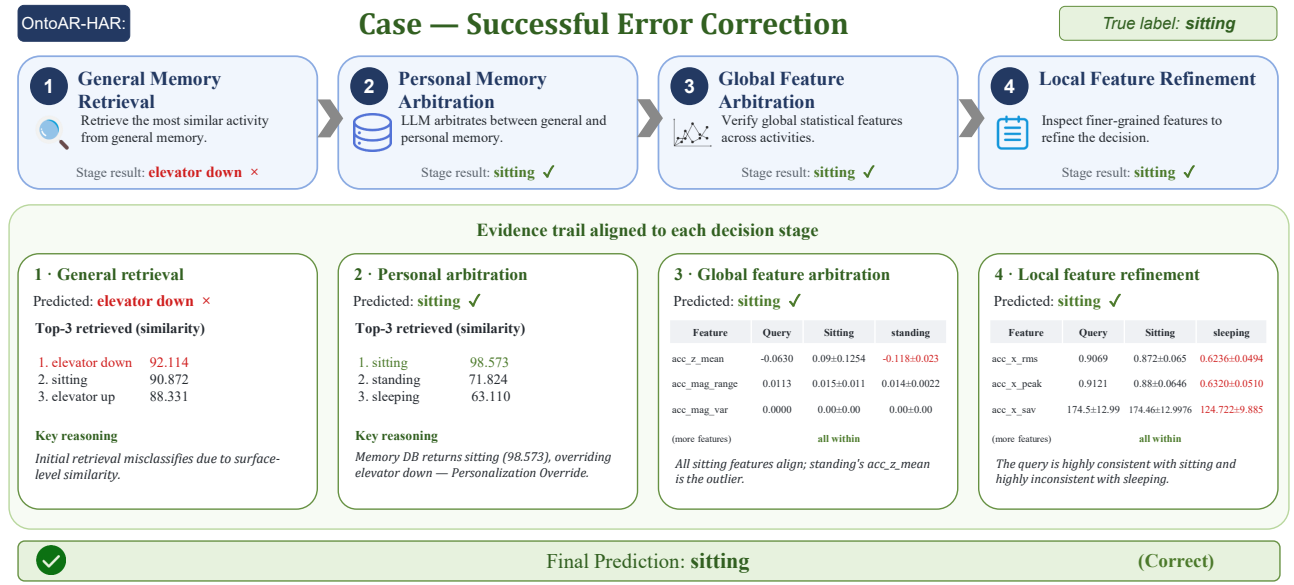


Fig. 4. Example decision path. Static population retrieval, personalized retrieval, and node-specific feature reports are kept separate in the context, allowing the final decision to be traced through the ontology.

Opportunity and Shoaib, consistent with hierarchy being most useful when several activities share a broader motion concept.

Figure 3 isolates the two structural components with Gemini 2.5 Flash. Average accuracy rises from 84.0 with single-level retrieval to 87.2 with hierarchy and 89.0 after adding personalized memory. The hierarchy helps most on closely related subclasses, while the additional memory gain shows that current-user examples complement static population evidence.

The two increments answer different questions. The first retains retrieval and LLM reasoning but replaces the global candidate set with successive sibling comparisons, testing whether label organization improves the use of otherwise similar evidence. The second retains that hierarchy and permits accepted prior windows from the inference user to appear beside the population neighbors. Their cumulative result is consistent with complementary roles: the tree selects the comparison to make, and memory changes the examples available for making it. The experiment does not require parameter updates to either the sensor encoder or the LLM.

a) *Semantic-inclusion evaluation.*: We compare controlled single-level and hierarchy outputs using path-aware metrics. Hierarchical F1 measures overlap between the true and predicted ontology paths after removing the common root; tree distance is their shortest-path length; cross-branch error counts predictions outside the true top-level branch. Table III shows gains on all datasets: average Hierarchical F1 rises from 89.92 to 91.32, while tree distance falls from 0.428 to 0.366 and cross-branch error from 5.18% to 4.96%. Together with the 84.24-to-87.17 accuracy gain, hierarchy improves both leaf recognition and the semantic proximity of residual errors.

C. Heterogeneous-Dataset Evaluation

We form a mixed benchmark from UCI-HAR, PAMAP2, USC-HAD, Mhealth, and DSADS using six harmonized activity categories and no body-location filtering. Database and inference users remain disjoint, and retrieval searches the heterogeneous collection globally.

TABLE V
DETAILED PERFORMANCE OF DIFFERENT LLM BACKBONES IN THE CONTROLLED COMPARISON (%). ACCURACY AND MACRO-F1 ARE SHOWN SIDE BY SIDE. BEST RESULTS ARE BOLD, AND SECOND-BEST RESULTS ARE UNDERLINED.

Backbone	Opp.		UCI		Shoaib		PAMAP2		USC		Mhealth		WISDM		DSADS		Avg.	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1
Gemini 2.0 Flash	83.00	82.71	92.50	92.33	96.19	96.19	85.42	85.94	71.67	71.60	85.83	85.51	78.12	77.30	90.53	90.44	85.41	85.25
Gemini 2.5 Flash	92.50	92.51	95.42	95.38	100.00	100.00	85.83	86.01	74.58	74.60	90.42	89.33	79.51	78.54	93.68	93.05	88.99	88.68
Gemini 3.0 Flash	<u>90.50</u>	<u>90.62</u>	94.17	94.09	100.00	100.00	89.17	<u>89.49</u>	82.50	82.01	<u>89.58</u>	89.45	80.90	80.27	94.74	94.79	90.20	90.09
DeepSeek R1	80.50	80.28	94.12	<u>94.17</u>	96.67	96.66	85.00	85.22	71.25	71.50	78.33	77.80	<u>79.51</u>	78.66	92.63	91.98	84.75	84.53
Grok 4	85.50	85.41	92.50	92.44	<u>99.52</u>	<u>99.52</u>	<u>87.50</u>	87.90	72.50	72.41	85.00	84.58	78.82	77.83	93.15	92.78	86.81	86.61
GPT-5	89.50	89.65	92.50	92.42	100.00	100.00	89.17	89.56	<u>77.50</u>	<u>77.43</u>	89.17	88.97	78.47	77.33	94.74	<u>94.54</u>	88.88	<u>88.74</u>

TABLE VI
PER-QUERY EFFICIENCY WITH GEMINI 2.5 FLASH. C/W DENOTES CHANNELS/WINDOW; RET. IS LOCAL RETRIEVAL TIME; TOKENS (K) ARE PROMPT/COMPLETION/TOTAL, WITH THE TOTAL COMPUTED BEFORE ROUNDING; COST IS IN USD.

Dataset (C/W)	Ret. (s)	LLM (s)	Tokens (K)	Cost
DSADS (30/125)	0.057	1.39	1.91/0.35/2.25	0.001442
Mhealth (15/100)	0.036	1.67	1.29/0.31/1.60	0.001170
Opportunity (30/60)	0.077	2.59	2.71/0.50/3.20	0.002055
PAMAP2 (18/200)	0.054	2.52	2.31/0.45/2.76	0.001822
Shoaib (30/100)	0.092	1.48	1.61/0.47/2.08	0.001655
UCI-HAR (6/128)	0.019	4.96	1.03/0.18/1.22	0.000768
USC-HAD (6/200)	0.022	3.99	1.33/0.22/1.55	0.000945
WISDM (6/200)	0.041	4.72	1.00/0.26/1.25	0.000947

The six evaluation activities are standing, sitting, lying, walking, upstairs, and downstairs. Labels with common or closely corresponding meanings are mapped to this output space, but the underlying windows come from different sensor placements and collection protocols. Keeping those placements rather than filtering to a single location makes the task a joint test of activity semantics and sensing heterogeneity. In particular, a label match across datasets does not imply that its raw channel values or nearest-neighbor geometry match. The ontology supplies a common sequence of local label decisions, while the embedding and retrieved examples still have to carry the signal-level evidence for each query.

Table IV shows 48.70 accuracy and 42.11 macro-F1 for OntoAR-HAR with Gemini 3.0, versus 43.0/35.8 for UniMTS. OntoAR-HAR leads on four of five datasets, indicating that local semantic decisions transfer across mixed sensing configurations; Mhealth also shows that representation quality and source coverage still affect retrieved evidence.

D. Backbone and Reasoning Effects

Table V is a fixed-subset backbone comparison, not a replacement for Table II’s ten-sample estimates. Gemini-family average Acc/F1 increases from 85.41/85.25 to 88.99/88.68 and 90.20/90.09; DeepSeek R1, Grok 4, and GPT-5 confirm compatibility but also backbone sensitivity.

Figure 5 isolates the Gemini 2.5 Flash decision stage. Mean accuracy rises from 81.51 to 84.40, showing a contribution beyond similarity aggregation, although this test does not separate general reasoning from model-specific evidence integration.

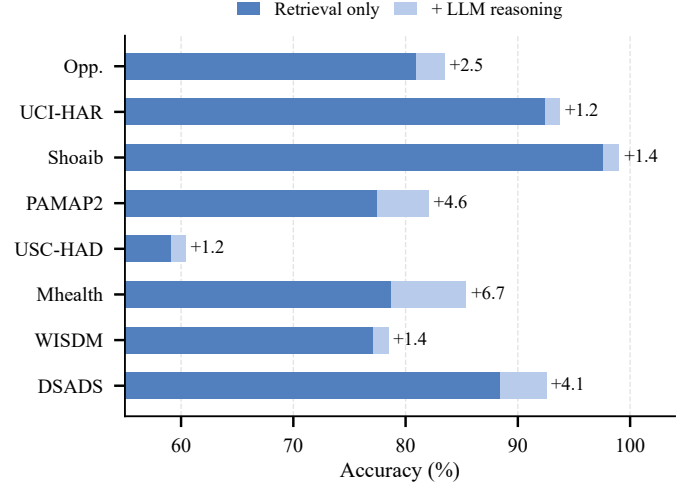


Fig. 5. Retrieval-only accuracy and the change obtained after LLM reasoning with the same retrieved evidence. Positive changes occur on all eight datasets.

This comparison is especially relevant to the feature construction stage. The retrieved list is not only a vote over stored labels: it supplies class-conditioned examples against which executable signal features can be evaluated. The LLM receives the resulting values and local candidate descriptions, then selects a branch. The positive change across the eight datasets indicates that this additional decision step uses information beyond the retrieval-only label ranking. Because the backbone is fixed in this comparison, the observed difference belongs to the evidence-to-label decision stage rather than a change in the underlying sensor encoder.

E. Inference Cost and Reasoning Example

Table VI reports Gemini 2.5 Flash inference cost. Retrieval takes 0.019–0.092 seconds per query, versus 1.39–4.96 seconds for each LLM decision; prompts contain 1.00K–2.71K tokens and cost USD 0.000768–0.002055. Sequential LLM calls therefore dominate latency, while the ontology limits each call to the current node’s children.

Figure 4 illustrates the evidence flow. Static retrieval initially favors *elevator down* (92.114) over *sitting* (90.872), but personalized retrieval supplies a previously accepted *sitting* episode (98.573). Subsequent global and local feature comparisons separate *sitting* from *standing* and *sleeping*. The final

path is thus supported by inspectable population examples, user-specific episodes, and numerical feature checks.

The case shows why nearest-neighbor similarity need not be the final activity decision. The static store makes an elevator-related label plausible, but the personal example provides stronger evidence for this user’s sitting pattern. A feature comparison then checks that interpretation against other stationary labels. Each source of information has a visible role: the static examples prevent the personalized match from being considered in isolation, the personal record captures a recurring user pattern, and the node features address a narrower motion distinction. This interpretation is limited to the illustrated query; the dataset-level evidence for the components comes from the ablation and cross-subject tables.

F. Discussion

The ontology and personalized store act at complementary levels: the former restricts each comparison through label relations, while the latter changes its evidence using current-user observations without replacing the source reference. Executable features add signal-level evidence where needed. The LLM therefore integrates bounded candidates, retrieved examples, and feature statistics rather than raw arrays or an unconstrained label list, leaving an inspectable decision path.

V. CONCLUSION AND FUTURE WORK

OntoAR-HAR combines an activity ontology with static population and personalized knowledge stores for sensor-based HAR. The ontology represents inclusion relations among labels and restricts retrieval and feature reasoning to local sibling decisions. Personalized episodes complement static population evidence without parameter updates. Across eight datasets, the hierarchy and memory provide additive gains, while the heterogeneous-dataset experiment indicates that the same structure can help under mixed sensing conditions. The framework thus connects label-level semantics, source-domain sensor evidence, and observations from the current user in a single recursive inference process.

Future work will automate and scale ontology construction beyond the current manual, confusion-guided procedure, including the generation and validation of intermediate concepts for larger activity vocabularies. We will also develop more complete memory gates that model confidence and long-term drift, and improve feature selection with broader libraries, automated validation, and robust fallback mechanisms. These extensions will reduce manual ontology design and make personalization more reliable over longer user streams.

REFERENCES

- [1] S. Ji, X. Zheng, and C. Wu, “Hargpt: Are llms zero-shot human activity recognizers?” in *2024 IEEE International Workshop on Foundation Models for Cyber-Physical Systems & Internet of Things (FMSys)*. IEEE, 2024, pp. 38–43, doi: 10.1109/FMSys62467.2024.00011.
- [2] P. Lewis *et al.*, “Retrieval-augmented generation for knowledge-intensive nlp tasks,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 9459–9474, 2020.
- [3] N. Sivaroopan, H. Karunarathna, C. Madarasingha, A. Jayasumana, and K. Thilakarathna, “RAG-HAR: Retrieval augmented generation-based human activity recognition,” in *2026 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, 2026, pp. 1–12, doi: 10.1109/PerCom67906.2026.11524560.
- [4] Z. Li, B. Chen, H. Xue, and F. D. Salim, “Zara: Training-free motion time-series reasoning via evidence-grounded LLM agents,” in *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2026, pp. 14 986–15 008, doi: 10.18653/v1/2026.acl-long.684.
- [5] F. Ordóñez and D. Roggen, “Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition,” *Sensors*, vol. 16, no. 1, p. 115, Jan. 2016, doi: 10.3390/s16010115.
- [6] A. van den Oord, Y. Li, and O. Vinyals, “Representation Learning with Contrastive Predictive Coding,” 2018, doi: 10.48550/arXiv.1807.03748.
- [7] H. Xu, P. Zhou, R. Tan, M. Li, and G. Shen, “LIMU-BERT: Unleashing the Potential of Unlabeled Data for IMU Sensing Applications,” in *Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys ’21. New York, NY, USA: Association for Computing Machinery, Nov. 2021, pp. 220–233, doi: 10.1145/3485730.3485937.
- [8] Z. Yue *et al.*, “Ts2vec: Towards universal representation of time series,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 8, 2022, pp. 8980–8987, doi: 10.1609/aaai.v36i8.20881.
- [9] V. Feofanov *et al.*, “Mantis: Lightweight foundation model for time series classification,” in *Forty-third International Conference on Machine Learning*, 2026. [Online]. Available: <https://openreview.net/forum?id=gbJMAjXLZ4>
- [10] Y. Ganin *et al.*, “Domain-adversarial training of neural networks,” *Journal of Machine Learning Research*, vol. 17, no. 59, pp. 1–35, 2016.
- [11] S. Miao and L. Chen, “GOAT: A Generalized Cross-Dataset Activity Recognition Framework with Natural Language Supervision,” *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 8, no. 4, pp. 1–28, Nov. 2024, doi: 10.1145/3699736.
- [12] H. Yan, H. Tan, Y. Ding, P. Zhou, V. Nambodiri, and Y. Yang, “Large language model-guided semantic alignment for human activity recognition,” *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 9, no. 4, pp. 1–25, 2025, doi: 10.1145/3770652.
- [13] X. Zhang *et al.*, “Unimts: Unified pre-training for motion time series,” *Advances in Neural Information Processing Systems*, vol. 37, pp. 107 469–107 493, 2024, doi: 10.52202/079017-3414.
- [14] A. Abedin, M. Ehsanpour, Q. Shi, H. Rezatofighi, and D. C. Ranasinghe, “Attend and Discriminate: Beyond the State-of-the-Art for Human Activity Recognition Using Wearable Sensors,” *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 5, no. 1, pp. 1–22, Mar. 2021, doi: 10.1145/3448083.
- [15] H. Yuan *et al.*, “Self-supervised learning for human activity recognition using 700,000 person-days of wearable data,” *NPJ digital medicine*, vol. 7, no. 1, p. 91, 2024, doi: 10.1038/s41746-024-01062-3.
- [16] Q. Wei, J. Huang, Y. Gao, and W. Dong, “One Model to Fit Them All: Universal IMU-based Human Activity Recognition with LLM-assisted Cross-dataset Representation,” *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 9, no. 3, pp. 1–22, Sep. 2025, doi: 10.1145/3749509.
- [17] M. Xue *et al.*, “MobHAR: Source-free Knowledge Transfer for Human Activity Recognition on Mobile Devices,” *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 9, no. 1, pp. 1–24, Mar. 2025, doi: 10.1145/3712620.
- [18] Z. Li, S. Deldari, L. Chen, H. Xue, and F. D. Salim, “Sensorllm: Aligning large language models with motion sensors for human activity recognition,” in *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 2025, pp. 354–379, doi: 10.18653/v1/2025.emnlp-main.19.
- [19] Y. Zhang *et al.*, “SensorLM: Learning the language of wearable sensors,” in *Advances in Neural Information Processing Systems*, vol. 38, 2025, pp. 52 274–52 310, doi: 10.52202/085713-1559.
- [20] L. Arrotta, G. Civitarese, and C. Bettini, “Semantic loss: A new neuro-symbolic approach for context-aware human activity recognition,” *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 7, no. 4, Dec. 2023, art. no. 147, doi: 10.1145/3631407.
- [21] S. Moon, A. Madotto, Z. Lin, A. Saraf, A. Bearman, and B. Damavandi, “IMU2CLIP: Language-grounded Motion Sensor Translation with Multimodal Contrastive Learning,” in *Findings of the Association for Computational Linguistics: EMNLP 2023*. Singapore: Association for Computational Linguistics, 2023, pp. 13 246–13 253, doi: 10.18653/v1/2023.findings-emnlp.883.

- [22] B. Chen, W. Wongso, Z. Li, Y. Khaokaew, H. Xue, and F. D. Salim, "CO-MODO: Cross-modal video-to-IMU distillation for efficient egocentric human activity recognition," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 10, no. 2, pp. 1–29, 2026, doi: 10.1145/3810218.
- [23] R. R. Chowdhury *et al.*, "Zerohar: Sensor context augments zero-shot wearable action recognition," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 15, 2025, pp. 16 046–16 054, doi: 10.1609/aaai.v39i15.33762.
- [24] D. Wang, E. Shelhamer, S. Liu, B. Olshausen, and T. Darrell, "Tent: Fully test-time adaptation by entropy minimization," in *International Conference on Learning Representations*, 2021. [Online]. Available: <https://openreview.net/forum?id=uXl3bZLkr3c>
- [25] Q. Wang, O. Fink, L. V. Gool, and D. Dai, "Continual test-time domain adaptation," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7191–7201, 2022, doi: 10.1109/CVPR52688.2022.00706. [Online]. Available: <https://api.semanticscholar.org/CorpusID:247748613>
- [26] G. V. Cormack, C. L. A. Clarke, and S. Buettcher, "Reciprocal rank fusion outperforms condorcet and individual rank learning methods," in *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '09. New York, NY, USA: Association for Computing Machinery, Jul. 2009, pp. 758–759, doi: 10.1145/1571941.1572114.
- [27] D. Roggen *et al.*, "Collecting complex activity datasets in highly rich networked sensor environments," in *2010 Seventh International Conference on Networked Sensing Systems (INSS)*. Kassel, Germany: IEEE, Jun. 2010, pp. 233–240, doi: 10.1109/INSS.2010.5573462.
- [28] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones," in *Proc. European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)*, 2013, pp. 437–442. [Online]. Available: <https://www.esann.org/sites/default/files/proceedings/legacy/es2013-84.pdf>
- [29] M. Shoaib, S. Bosch, O. Incel, H. Scholten, and P. Havinga, "Fusion of Smartphone Motion Sensors for Physical Activity Recognition," *Sensors*, vol. 14, no. 6, pp. 10 146–10 176, Jun. 2014, doi: 10.3390/s140610146.
- [30] A. Reiss and D. Stricker, "Introducing a New Benchmarked Dataset for Activity Monitoring," in *2012 16th International Symposium on Wearable Computers*. Newcastle, United Kingdom: IEEE, Jun. 2012, pp. 108–109, doi: 10.1109/ISWC.2012.13.
- [31] M. Zhang and A. A. Sawchuk, "USC-HAD: A daily activity dataset for ubiquitous activity recognition using wearable sensors," in *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, ser. UbiComp '12. New York, NY, USA: Association for Computing Machinery, Sep. 2012, pp. 1036–1043, doi: 10.1145/2370216.2370438.
- [32] O. Banos *et al.*, "Design, implementation and validation of a novel open framework for agile development of mobile health applications," *BioMedical Engineering OnLine*, vol. 14, no. Suppl 2, p. S6, 2015, doi: 10.1186/1475-925X-14-S2-S6.
- [33] G. M. Weiss, "WISDM smartphone and smartwatch activity and biometrics dataset," UCI Machine Learning Repository, Dataset, 2019, doi: 10.24432/C5HK59.
- [34] K. Altun, B. Barshan, and O. Tunçel, "Comparative study on classifying human activities with miniature inertial and magnetic sensors," *Pattern Recognition*, vol. 43, no. 10, pp. 3605–3620, Oct. 2010, doi: 10.1016/j.patcog.2010.04.019.
- [35] R. Girdhar *et al.*, "Imagebind: One embedding space to bind them all," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 15 180–15 190, doi: 10.1109/CVPR52729.2023.01457.
- [36] G. Zerveas, S. Jayaraman, D. Patel, A. Bhamidipaty, and C. Eickhoff, "A transformer-based framework for multivariate time series representation learning," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021, pp. 2114–2124, doi: 10.1145/3447548.3467401.
- [37] R. R. Chowdhury, X. Zhang, J. Shang, R. K. Gupta, and D. Hong, "Tarnet: Task-aware reconstruction for time-series transformer," in *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data Mining*, 2022, pp. 212–220, doi: 10.1145/3534678.3539329.
- [38] B. Li, W. Cui, W. Wang, L. Zhang, Z. Chen, and M. Wu, "Two-stream convolution augmented transformer for human activity recognition," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 1, 2021, pp. 286–293, doi: 10.1609/aaai.v35i1.16103.
- [39] Z. Leng *et al.*, "IMUGPT 2.0: Language-Based Cross Modality Transfer for Sensor-Based Human Activity Recognition," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 8, no. 3, pp. 1–32, Aug. 2024, doi: 10.1145/3678545.
- [40] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long, "TimesNet: Temporal 2D-variation modeling for general time series analysis," in *International Conference on Learning Representations*, 2023. [Online]. Available: <https://mlanthology.org/iclr/2023/wu2023iclr-timesnet/>
- [41] T. Zhou, P. Niu, X. Wang, L. Sun, and R. Jin, "One fits all: Power general time series analysis by pretrained LM," *Advances in Neural Information Processing Systems*, vol. 36, pp. 43 322–43 355, 2023, doi: 10.52202/075280-1877.
- [42] X. Zhang, R. R. Chowdhury, J. Zhang, D. Hong, R. K. Gupta, and J. Shang, "Unleashing the power of shared label structures for human activity recognition," in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, 2023, pp. 3340–3350, doi: 10.1145/3583780.3615101.