AgentSense: Virtual Sensor Data Generation Using LLM Agents in Simulated Home Environments

Zikang Leng[1*], Megha Thukral[1*], Yaqi Liu[1*], Hrudhai Rajasekhar[1], Shruthi K. Hiremath[1], Jiaman He[2], Thomas Plötz[1]

[1]School of Interactive Computing, Georgia Institute of Technology, Atlanta, USA
[2]RMIT University, Melbourne, Australia
{zleng7, mthukral3, yliu3387, hrajasekhar3, shiremath9, thomas.ploetz}@gatech.edu, jiaman.he@student.rmit.edu.au

Abstract

A major challenge in developing robust and generalizable Human Activity Recognition (HAR) systems for smart homes is the lack of large and diverse labeled datasets. Variations in home layouts, sensor configurations, and individual behaviors further exacerbate this issue. To address this, we leverage the idea of embodied AI agents—virtual agents that perceive and act within simulated environments guided by internal world models. We introduce AgentSense, a virtual data generation pipeline in which agents live out daily routines in simulated smart homes, with behavior guided by Large Language Models (LLMs). The LLM generates diverse synthetic personas and realistic routines grounded in the environment, which are then decomposed into fine-grained actions. These actions are executed in an extended version of the VirtualHome simulator, which we augment with virtual ambient sensors that record the agents' activities. Our approach produces rich, privacy-preserving sensor data that reflects real-world diversity. We evaluate AgentSense on five real HAR datasets. Models pretrained on the generated data consistently outperform baselines, especially in low-resource settings. Furthermore, combining the generated virtual sensor data with a small amount of real data achieves performance comparable to training on full real-world datasets. These results highlight the potential of using LLM-guided embodied agents for scalable and cost-effective sensor data generation in HAR.

Code — https://github.com/ZikangLeng/AgentSense

Introduction

In 1999, *The Matrix* created a simulated reality, one in which most humans lived unknowingly inside a computer-generated illusion. That world was created by intelligent machines, built not to serve humanity, but to control it. Today, we are building simulated environments of our own. But this time, the purpose is different: to understand, model, and support human life through intelligent systems. With the advent of large language models (LLMs) and rich, interactive simulations, we now have the tools to do so.

In 2018, Xia et al. (2018) introduced the Gibson Environment, a realistic 3D simulation platform designed for training and evaluating embodied agents—AI systems that perceive and act within physical or simulated environments. Since then, virtual embodied agents (VEAs) have become integral to a wide range of interactive and conversational tasks. These agents take on diverse forms, from virtual 2D or 3D avatars to physical robotic androids equipped with synthetic skin, expressive facial features, and motorized mechanisms for controlling facial expressions and lip movements (Fung et al. 2025). Recent advances have also explored the use of world model-based approaches, which allow agents to ground vision-language prompts within embodied domains and learn complex behaviors through imaginative simulation (Mazzaglia et al. 2024).

Building on the foundation of prior work in embodied AI, we shift the focus from simulating robotic behaviors or dialogue agents to simulating human lives within virtual smart home environments. Our aim is to translate the simulated lives into sensor-level data. Specifically, we focus on smart home based Human Activity Recognition (HAR), which rely on ambient sensors to monitor daily activities. HAR is crucial in domains such as healthcare, elder care, and assisted living (Alam, Reaz, and Ali 2012; Qi et al. 2018; Chernova et al. 2024). However, the development of effective HAR models is limited by the scarcity of large, diverse, and annotated sensor datasets (Liciotti et al. 2020; Bouchabou et al. 2021).

To address this challenge, we simulate diverse human personas and embed them as agents within virtual smart home environments. Each persona is generated using large language models (LLMs), which also direct the agent's behavior by producing realistic daily routines and corresponding actions grounded in the simulated home environment. These actions are executed within VirtualHome, a 3D simulation platform that we extend with ambient sensor capabilities—referred to as X-VirtualHome.

Each interaction with the environment—opening a door, turning on a light, walking between rooms—triggers corresponding virtual sensor signals, mimicking real-world smart home sensor data. This process allows us to generate large-scale, fully annotated, privacy-preserving datasets without the need for intrusive real-world data collection.

Crucially, our simulation is not solely aimed at generating large volumes of data, but at producing diverse and struc-

*These authors contributed equally.

tured sensor data that capture the heterogeneity of human behavior. By systematically varying personas, daily routines, home layouts, and sensor configurations, our approach generates datasets designed to help HAR models generalize across a wide range of real-world scenarios.

In our experiments, we simulate 18 distinct personas across 22 unique home layouts, generating a total of 250 days of virtual sensor data. We evaluate the resulting models on five real-world smart home datasets—Aruba, Milan, Kyoto7, and Cairo from the CASAS collection (Cook et al. 2012), as well as Orange4Home (Cumin et al. 2017)—and find that HAR models pretrained on our simulated data consistently outperform baseline methods. Notably, even when fine-tuned with only a small amount of real data, these models achieve performance comparable to those trained entirely on full real-world datasets. These results underscore the practical value of our approach for developing more efficient and scalable HAR systems.

To summarize, through our LLM-based agents in simulated home environments, we make the following contributions:

1. *An embodied agent framework for smart home data simulation:* We present AgentSense, where LLM-driven agents enact daily lives in virtual smart homes to generate structured, privacy-preserving ambient sensor datasets.
2. *Comprehensive evaluation of virtual sensor data for HAR:* We demonstrate the effectiveness of our approach through extensive experiments on five benchmark smart home datasets. Our results show that models pretrained on virtual data significantly improve HAR performance—especially in low-data regimes—and can match full-data baselines when combined with minimal real-world data.

Related Work

Embodied Agents and Simulation Platforms Embodied agents are AI systems that perceive, reason, and act within physical or simulated environments. This line of research has become foundational to robotic navigation, task planning, and interactive learning. Simulation platforms have been critical to this progress. The Gibson Environment (Xia et al. 2018) introduced a photorealistic 3D simulator for training and evaluating embodied agents in rich environments. Platforms such as AI2-THOR (Kolve et al. 2017), Habitat (Savva et al. 2019), and VirtualHome (Puig et al. 2018) further expanded agent capabilities, enabling object interactions, scripted activities, and complex navigation.

Recent work has broadened the scope of embodied agents beyond physical interaction. Agents now include expressive humanoid avatars and physically embodied androids capable of gaze, speech, and facial expression (Fung et al. 2025). Others integrate world models that support abstract reasoning and future-state simulation (Mazzaglia et al. 2024), improving planning and grounded decision-making.

Despite this progress, simulating human daily routines in smart home contexts for data generation remains underexplored. We build on this direction by generating structured, ambient sensor data from agents enacting daily lives in simulated homes. Rather than creating one-to-one digital twins (Grieves and Vickers 2016), we adopt the digital cousin approach (Dai et al. 2024), simulating diverse agents and environments to support diverse data generation.

A natural extension of this approach is to integrate LLMs to guide agent behavior.

Language Models for Behavior Simulation Recent advances in LLMs have enabled their integration into agent-based systems for reasoning, planning, and interaction. Several works have explored using LLMs to control autonomous agents in virtual environments. For instance, Voyager (Wang et al. 2023) uses GPT-4 to explore, plan, and act in Minecraft by generating code and updating a skill library. Generative Agents (Park et al. 2023) simulate human behaviors in a virtual town by assigning LLM-driven agents memories, goals, and interactions. CAMEL (Li et al. 2023) employs role-playing to facilitate multi-agent cooperation toward task completion.

These efforts demonstrate the potential of LLMs to produce structured, plausible agent behaviors. However, most focus on narrative, dialogue, or open-ended exploration rather than generating structured data for downstream tasks, though recent work has examined LLM–human equivalence in annotation behavior (He et al. 2025). Our work extends this line by using LLM-guided agents to simulate the daily lives of diverse synthetic personas in virtual smart homes, enabling ambient sensor data generation for training HAR models—a new application of LLM-embodied simulation.

Synthetic Data Generation for Human Activity Recognition Smart home-based Human Activity Recognition (HAR) systems rely on ambient sensors to passively and privacy-preservingly monitor daily activities (Cook et al. 2012; Cumin et al. 2017). However, building robust HAR models is challenging due to the scarcity of large, diverse, annotated datasets that capture variations in home layouts, sensor setups, and resident routines (Liciotti et al. 2020; Bouchabou et al. 2021).

To address data scarcity, recent work has explored synthetic data generation, particularly for wearable HAR—e.g., generating IMU data from video (Kwon et al. 2020), audio (Liang et al. 2022), and text (Leng, Kwon, and Ploetz 2023; Leng et al. 2024). However, these methods do not naturally extend to ambient sensors, which involve distinct spatial and triggering mechanisms.

Simulation environments like VirtualHome (Puig et al. 2018) have been used to model household activities. Some approaches generate routines from program sketches (Liao et al. 2019) or use LLMs for daily schedule generation (Yonekura et al. 2024) and action planning (Huang et al. 2022), but none produce ambient sensor data.

Our work bridges this gap by generating synthetic ambient sensor data from LLM-guided agents acting out daily routines in simulated homes. By varying personas, routines, and environments, we produce structured, diverse datasets that better reflect real-world variability.

Methodology

We present a system that uses LLMs and an extended version of the VirtualHome simulator—***X-VirtualHome***—to generate virtual ambient sensor data across diverse home environments and resident profiles (Figure 1). Our pipeline prompts LLMs to create personas and daily routines, which are decomposed into simulator-executable action sequences. We extend VirtualHome by adding virtual motion, appliance door, and device activation sensors, enabling the simulation of privacy-preserving ambient data for training activity recognition models.

Multi-Level Prompting of LLM Agents to Generate Daily Routines in Diverse Home Environments

We use a three-stage prompting pipeline to generate diverse, simulator-executable daily routines from LLMs. Starting from generated personas, we produce a high-level schedule, which is then decomposed into fine-grained action sequences compatible with X-VirtualHome.

Persona Generation We aim to capture behavioral diversity crucial for robust HAR models by generating a wide range of personality profiles. Real-world routines vary significantly by age, occupation, health, and lifestyle. Collecting such varied real-user data at scale is costly and time-intensive. Instead, we leverage LLMs to generate realistic personality descriptions, inspired by prior work on persona generation (Abbasiantaeb et al. 2024; Smrke et al. 2025; Serapio-García et al. 2023).

High-Level Daily Routine Generation We focus on simulating routines for a single resident. To generate a persona-specific daily schedule, we prompt the LLM with:

1. **Persona:** A generated profile including age, occupation, health status, and lifestyle. These attributes shape behavior and introduce meaningful variability for HAR modeling.
2. **Day of the Week:** Daily activities differ across the week (e.g., workdays vs. weekends). Specifying the day guides the model to generate contextually appropriate routines.
3. **Environment:** A list of rooms in the selected VirtualHome layout. To ensure physical plausibility, the LLM must know what rooms exist. Activities are tagged as "at home" or "outside," and only in-home activities are retained for simulation.
4. **Example Schedules:** We provide few-shot examples adapted from the Homer dataset (Patel and Chernova 2022), which reflects real human routines. We also instruct the LLM to avoid overly neat time slots (e.g., always ending in 0 or 5) to better mimic natural scheduling.

Decomposing High-Level Routines into Low-Level Actions We decompose each high-level activity into a sequence of simulator-executable actions. To do so, we prompt the LLM with the following inputs:

1. **Persona:** Generated personality profile, which includes age, occupation, lifestyle, and health status. These attributes affect how actions are realistically performed—e.g., a retiree may move slower or take more steps.
2. **Activity:** A scheduled high-level activity and its start/end time. Only in-home activities are decomposed for compatibility with X-VirtualHome.
3. **Environment:** The target room and a list of objects available within it. A separate LLM identifies the appropriate room, and we pass the full object list to ground the model's outputs in the actual environment.
4. **Actions:** A predefined set of 18 allowed simulator actions with descriptions (e.g., `[walk] <bedroom>`) to constrain output format and ensure simulator compatibility.
5. **Example Decompositions:** Reference examples demonstrate how to convert high-level tasks into action sequences. These clarify structure and formatting, improving consistency across generated outputs.

Converting LLM Output to Executable Actions in VirtualHome

To ensure LLM-generated routines are executable in X-VirtualHome, we convert them into simulator-compatible commands. Left unchecked, raw LLM output may include hallucinated or out-of-vocabulary tokens that cause execution failures. We address this through a five-step process that cleans and validates the output against VirtualHome's ontology.

1. **Output Cleaning.** We remove extraneous metadata (e.g., day labels, high-level activities), retaining only low-level actions annotated with location and timestamp (e.g., `[walk] <doorjamb> (06:42{06:42) (bedroom))`.
2. **Embedding the VirtualHome Vocabulary.** Define
   $$\mathcal{A} = \{\text{valid actions}\}, \quad \mathcal{O} = \{\text{valid objects}\}.$$
   Each $x \in \mathcal{A} \cup \mathcal{O}$ is embedded using OpenAI's `text-embedding-3-small`:
   $$\mathbf{e}(x) = \text{Embed}(x) \in \mathbb{R}^d.$$
   These are stored in FAISS indices—one for actions, one for objects (Douze et al. 2024).
3. **Nearest-Neighbor Retrieval.** For each LLM token g, we compute $\mathbf{e}(g)$ and retrieve
   $$x^* = \arg\max_{x \in \mathcal{V}} \cos(\mathbf{e}(g), \mathbf{e}(x)),$$
   where $\mathcal{V} = \mathcal{A}$ for actions or $\mathcal{V} = \mathcal{O}_r \subset \mathcal{O}$ for room-specific objects, and
   $$\cos(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\|\|\mathbf{v}\|}.$$
4. **Thresholding and Replacement.** Accept x^* only if
   $$\cos(\mathbf{e}(g), \mathbf{e}(x^*)) \geq \tau,$$
   where $\tau_{\text{act}} = 0.8$ for actions and $\tau_{\text{obj}} = 0.6$ for objects. Otherwise, the line is flagged as invalid and regenerated by the LLM using surrounding context. After a fixed number of retries, unfixable lines are discarded.

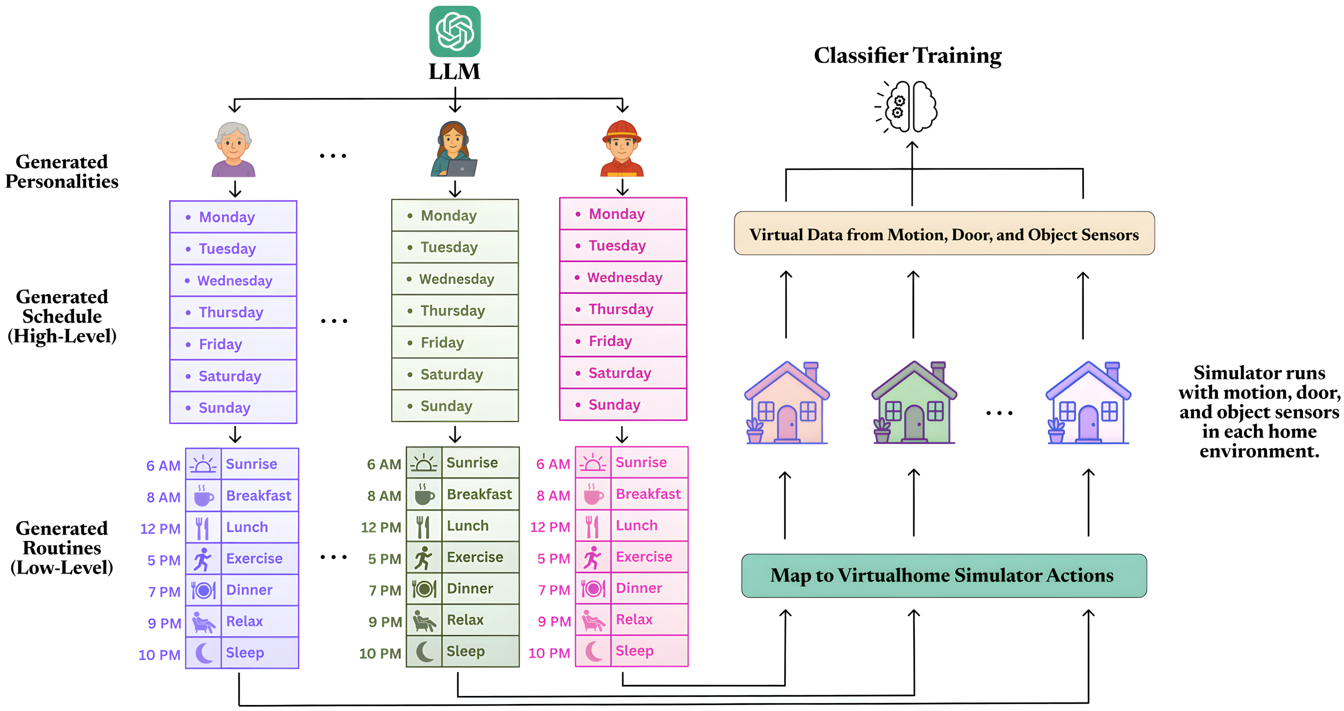

Figure 1: Overview of the framework. The LLM first generates diverse synthetic personas. For each persona, it then produces daily routines grounded in the context of a simulated environment. These routines are decomposed into fine-grained actions, which are executed in the X-VirtualHome simulator. The simulator, augmented with ambient sensors, captures virtual sensor data as the agent enacts its daily life.

5. **Final Command Assembly.** Validated tokens replace the originals, yielding simulator-ready commands:

```
[action*] <object*> (start--end) (room).
```

We implement this embedding and grounding workflow using LangChain (Chase and contributors 2023) with FAISS and OpenAI embeddings. This ensures semantic alignment with the simulator's ontology, eliminating hallucinations and preserving execution correctness.

Implementing Virtual Ambient Sensors in X-VirtualHome Simulator

We extend the VirtualHome simulator (Puig et al. 2018) to support ambient sensors—specifically motion, appliance door, and device activation sensors—aligned with LLM-generated routines. While VirtualHome allows for the simulation of scripted household activities and supports multimodal outputs (e.g., action logs, video, segmentation), it lacks support for privacy-preserving sensor simulation. To address this, we integrate virtual motion sensors that track character movement and simulate door/device activations by monitoring environment graph state changes (Puig et al. 2021).

Incorporating Motion Sensors in X-VirtualHome We augment each home environment by placing motion sensors based on room size. During simulation, character trajectories are tracked at fixed intervals, and motion detection events are recorded when the character enters a sensor's detection radius. Below is the detailed procedure:

1. **FindAllRooms**: All rooms in a home environment are detected using VirtualHome's `FindAllRooms`, which traverses the hierarchy to find active game objects tagged as `TYPE_ROOM`.
2. **DetermineSensorCount**: For each room, the area is computed from its bounding box. Sensor count is assigned as follows:
   - Small ($\leq 30\,m^2$): 1 sensor
   - Medium ($30 <$ area $\leq 60\,m^2$): 2 sensors
   - Large ($> 60\,m^2$): 3 sensors
3. **CalculateSensorPositions**: Sensors are placed near corners of the room, offset by 0.3 m horizontally and vertically (from floor and ceiling). Placement varies depending on the number of sensors required.
4. **CreateVirtualSensor**: Each sensor is instantiated with a unique ID, room name, and detection radius ($r = 5.0\,\text{m}$). A `Motion_sensor` component is attached, and the sensor is registered in `MotionSensorManager`.
5. **TrackCharacterPosition**: The character's position $\mathbf{p}(t)$ is logged every $\Delta t = 0.2\,\text{s}$ using Unity's `position` property. Motion detection is computed as:

$$\|\mathbf{p}(t) - \mathbf{s}_i\| \leq r$$

where $\mathbf{s}_i$ is the i-th sensor's 3D position.

6. **MotionDetectionEvent**: On detection, we log: $\{$*frame number*, *character ID*, *sensor ID*, *room name*, $(x, y, z)\}$. All events are saved for analysis.
7. **MotionTriggers**: Using recorded positions, we derive ON/OFF states. A segment $[t_{\text{on}}, t_{\text{off}}]$ is triggered if:

$$\|\mathbf{p}(t) - \mathbf{s}_i\| \leq r \quad \text{and} \quad \|\mathbf{p}(t) - \mathbf{p}(t - \Delta t)\| > \epsilon$$

with $\epsilon = 0.1\,\text{m}$ for distinguishing motion from jitter. These states are exported as virtual motion sensor readings.

Simulating Door and Device Activation Sensors

Environment Graph. The Environment Graph is a structured representation of the VirtualHome simulation environment. It encodes objects as nodes and their spatial or semantic relationships as edges. Formally, it is defined as a graph $G = (V, E)$, where:

- V is the set of nodes, each representing an object instance (e.g., *chair*, *table*, *toothbrush*).
- E is the set of directed edges representing object relations (e.g., *on top of*, *next to*). For example, the object *cup* may be "ON" the *kitchen counter*.

Each node maintains attributes including the object's class, associated room, and dynamic states (e.g., OPEN, ON), as well as static properties like CAN_OPEN and HAS_SWITCH.

We simulate two types of ambient sensors—door sensors and device activation sensors—by monitoring environment state transitions as recorded by the Environment Graph after each simulated action. Specifically:

- A door sensor event is triggered when an object with the property CAN_OPEN (e.g., doors, cabinets) transitions from CLOSED to OPEN.
- A device activation sensor event is triggered when an object with the property HAS_SWITCH (e.g., microwave, washing machine) transitions from OFF to ON.

These virtual sensor events are logged along with metadata such as *timestamp*, *object ID*, *room location*, and updated *object state*. This enables a temporally aligned ambient sensor stream suitable for privacy-preserving HAR model training.

Experiments

To evaluate the effectiveness of the virtual sensor data generated using our approach, we conduct experiments using established real-world smart home datasets. These benchmarks provide a controlled, reproducible setting for assessing model performance and allow us to demonstrate the practical utility of our virtual data generation—without deploying physical hardware. The following sections detail the datasets and classifier training setup.

Datasets

Real Datasets We conduct experiments on five publicly available datasets: Aruba, Milan, Kyoto7, and Cairo from the CASAS collection (Cook et al. 2012), and Orange4Home (Cumin et al. 2017) from the Amiqual4Home environment. Among the CASAS datasets, Aruba has the most data points and balanced sensor modalities (motion, door, temperature), with a floorplan similar to Milan, which shares the same sensor types. Kyoto7, Cairo, and Orange are multi-story homes with more diverse layouts and fewer samples. Cairo and Kyoto7 include two residents, adding behavioral variability, while Orange has one. Sensor types also vary: Cairo includes motion and temperature; Kyoto7 adds item usage, light switches, and device activations; Orange features 18+ modalities, including noise, voltage, and humidity.

Virtual Dataset We generated virtual sensor data for 18 distinct personas across 22 simulated home environments, yielding 250 days of activity data. The final dataset includes 3,266 activity windows, each containing between 3 and 393 sensor triggers (average: 36). All environments represent single-story homes with a single resident.

During data generation, LLM agents freely produced open-ended routines without restrictions on the activity space. To align these with real-world benchmarks, we mapped the LLM-generated activities to the label sets defined in the target HAR datasets. This was done by prompting the LLM with (1) the high-level activity name, (2) its decomposed sequence of low-level actions, and (3) the complete set of activity labels from the target dataset. The LLM then selected the most appropriate label, which we assigned to the corresponding virtual sensor data.

Since the simulation assumes single-resident settings, overlapping activities from multi-resident datasets were interpreted as being performed by one individual. Additionally, when real-world activity labels were not reflected in the virtual data—due to unconstrained routine generation—we assigned them the label "Other."

Classifier Training:

We adopt the TDOST-based HAR framework proposed by Thukral et al. (2025), which transfers across diverse home layouts and sensor configurations. Unlike transformer- or graph-based models that assume fixed topologies, TDOST is layout-agnostic, making it well-suited for cross-environment evaluation. This allows us to directly assess the impact of pretraining on virtual data.

We use a pre-segmented, activity-level windowing approach, selecting the first 100 sensor triggers from each activity window. Each trigger includes contextual metadata (e.g., sensor type, location, and timestamp), which we convert into natural language sentences using two TDOST variants:

- **TDOST-Basic:** Encodes sensor type and location. For example, *"Motion sensor in bedroom fired with value ON"*.
- **TDOST-Temporal:** Adds time information to the above, e.g., *"Motion sensor in bedroom fired with value ON at twelve hours six minutes PM"*.

These sentences are embedded using the `all-distilroberta-v1` model from Sentence-Transformers (Reimers and Gurevych 2019). The resulting

	Aruba	Milan	Cairo	Kyoto7	Orange
Accuracy					
Real (TDOST-Basic)	91.00 ± 0.53	90.07 ± 0.70	69.01 ± 2.16	70.31 ± 1.53	82.40 ± 0.64
Real+Virtual (TDOST-Basic)	**93.19 ± 0.22**	**91.97 ± 0.42**	**75.61 ± 1.93**	**70.31 ± 1.53**	**85.21 ± 0.82**
Real (TDOST-Temporal)	91.24 ± 0.41	86.58 ± 0.12	57.20 ± 1.00	48.09 ± 0.49	67.40 ± 0.39
Real+Virtual (TDOST-Temporal)	**93.67 ± 0.04**	**91.60 ± 0.70**	**67.10 ± 2.60**	**49.31 ± 0.49**	**67.60 ± 0.15**
Macro F1 Score					
Real (TDOST-Basic)	63.98 ± 0.66	70.81 ± 1.94	51.51 ± 1.58	52.48 ± 1.59	21.56 ± 3.75
Real+Virtual (TDOST-Basic)	**72.20 ± 0.62**	**74.44 ± 0.85**	**62.47 ± 1.95**	**56.07 ± 1.43**	**41.83 ± 2.58**
Real (TDOST-Temporal)	68.57 ± 1.21	57.20 ± 1.65	21.07 ± 2.28	29.51 ± 1.99	8.42 ± 2.58
Real+Virtual (TDOST-Temporal)	**77.36 ± 0.34**	**73.41 ± 0.93**	**46.49 ± 2.89**	**31.62 ± 2.16**	**10.25 ± 1.46**
Weighted F1 Score					
Real (TDOST-Basic)	89.81 ± 0.55	90.20 ± 0.69	66.79 ± 1.83	66.43 ± 1.48	75.91 ± 0.86
Real+Virtual (TDOST-Basic)	**92.41 ± 0.18**	**91.74 ± 0.30**	**74.65 ± 1.98**	**68.27 ± 1.02**	**83.42 ± 0.76**
Real (TDOST-Temporal)	89.57 ± 0.40	84.99 ± 0.33	44.45 ± 0.29	41.70 ± 1.46	61.68 ± 5.73
Real+Virtual (TDOST-Temporal)	**93.48 ± 0.12**	**91.23 ± 0.64**	**61.14 ± 2.85**	**43.96 ± 1.68**	**65.68 ± 0.43**

Table 1: Model performance (Accuracy, Weighted F1, Macro F1) comparing training on real data only versus pretraining on virtual data followed by finetuning on real data, using two TDOST embedding variants.

sequence of embeddings is passed into a bidirectional LSTM (Bi-LSTM) with 64 hidden units, following the architecture used by Thukral et al. (2025). Finally, a linear classification layer maps the encoded sequence to a probability distribution over activity classes.

Training Settings To evaluate the effectiveness of the generated virtual sensor data, we conduct two types of experiments across all real-world datasets: *Real* and *Real + Virtual*. In the *Real* setting, the model is trained and evaluated in a fully supervised manner on each dataset independently, serving as our baseline. In the *Real + Virtual* setting, we follow a two-stage training procedure inspired by Kwon et al. (2020): the model is first pretrained on virtual sensor data, then fine-tuned on real sensor data, with all weights updated. Final evaluation is performed on the real test split.

Results

From Table 1, we note that models pretrained with virtual data, Real + Virtual (TDOST-Basic) and Real + Virtual (TDOST-Temporal), consistently outperform their counterparts trained exclusively on real data across the benchmark HAR datasets.

For TDOST-Basic, the average accuracy increases across datasets, with notable gains such as 69.01% to 75.61% on Cairo, and 82.40% to 85.21% on Orange4Home. Similarly, TDOST-Temporal shows substantial improvements, especially on low-resource datasets, as such accuracy improves by 10% on Cairo and 5% on Milan. while performance on other datasets such as Kyoto7 and Orange4Home remains stable. Macro F1 Score shows substantial gains using our Virtual+Real supervised pipeline. For TDOST-Basic, the Macro F1 increases from 11% on Cairo and by approximately 20% on the Orange4Home dataset. Similarly, TDOST-Temporal shows a sharp improvement from 68.57% to 77.36% for Aruba, 51% to 73% for Milan. Weighted F1 scores follow a similar upward trend, with both TDOST-Basic and TDOST-Temporal showing significant improvements across datasets.

The consistency of improvements across all five datasets—including Orange4Home, which is a large-scale non-CASAS dataset with over 18 different types of sensors—further reinforces the generalizability of pretraining and robustness of our virtual sensor data across different sensor types, activity sets, and home environments. To sum up, substantial improvements in downstream HAR performance validate our virtual data-generation approach as an effective way to reduce reliance on costly real-world data collection, especially for complex or low-resource environments.

Ablation Studies

Varying the Amount of Real Data Used for Finetuning We analyze how varying the amount of real data used during fine-tuning affects downstream HAR performance. This experiment aims to identify the minimum quantity of real-world data needed to achieve competitive performance when models are pretrained on virtual data.

For each dataset, we first pretrain the model using all available virtual data, then fine-tune it using randomly sampled subsets of real data. The number of fine-tuning samples varies per dataset, as illustrated in Figure 2. To ensure consistency, the sampled subsets maintain the original distribution of activity classes. All experiments are conducted using the TDOST-Basic variant.

Across all five benchmark datasets, we observe that the *Real+Virtual* models consistently outperform those trained solely on real data, regardless of the amount used. This indicates that pretraining on virtual data provides a strong initialization, enabling robust performance even in low-data regimes. Notably, using only 5% to 10% of real data, pretraining on virtual data yields substantial gains over training

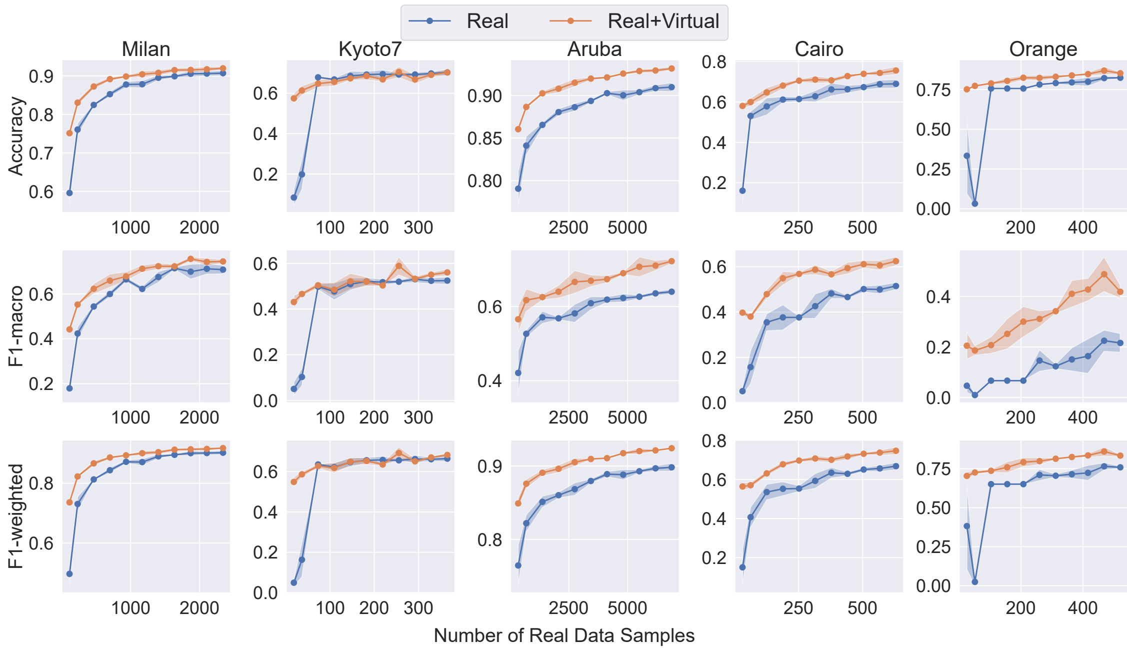

Figure 2: TDOST Basic model performance when different amount of real data are used for training. The amount of virtual data stays the same.

using only the real data. For example, we see a ∼10% improvement in macro F1 for Aruba and up to a 45% increase for Kyoto7 when using just 5% of real data. These findings support our hypothesis that virtual data can significantly reduce the need for large-scale real-world data collection.

In several cases, such as Orange4Home and Cairo, *Real+Virtual* performance approaches the upper bound of fully supervised training. On Cairo, for instance, models reach ∼80% accuracy and ∼60% macro F1 using just 200 real samples. Similarly, fewer than 200 samples are sufficient to achieve comparable results on Orange. This further underscores the practical value of virtual pretraining for efficient HAR model development.

Env	Days	Personas	Acc	Macro F1	Weighted F1
			92.72	68.35	91.66
✓			92.76	70.69	91.84
✓	✓		93.22	71.01	92.38
✓	✓	✓	93.19	72.20	92.41

Table 2: Effect of environment diversity, weekly routine coverage, and persona variation on HAR performance

Effectiveness of Individual Components We evaluate the impact of different generation settings—namely, personas, routines from multiple days of the week, and diverse home environments—on the effectiveness of virtual sensor data. All experiments are conducted on the Aruba dataset using the TDOST-Basic variant. As shown in Table 2, when virtual data is generated using a single persona, a single day's routine, and a single environment, the downstream classifier achieves a macro F1 score of 68.35%. Introducing diversity in environments (22 homes) increases performance to 70.69%. Adding routines from all seven days of the week further improves macro F1 to 71.01%. Finally, incorporating multiple personas leads to the highest performance of 72.20% macro F1.

These results indicate that each component—environment, daily routine, and persona—contributes positively to the performance of the downstream classifier. We attribute this improvement to increased diversity in the generated virtual sensor data, which enhances the generalizability of the downstream classifier. Notably, the total volume of virtual sensor data remains constant across settings, achieved through repeated generation.

Conclusion

We introduced *AgentSense*, a data generation framework that leverages LLM-guided embodied agents to simulate human lives in virtual smart home environments. By generating diverse synthetic personas and their daily routines grounded in varied home layouts, *AgentSense* produces structured, richly annotated ambient sensor data designed to reflect the heterogeneity of real-world settings. We extended the VirtualHome simulator with virtual ambient sensors, enabling the conversion of LLM-generated action sequences into sensor data.

Our experiments across five real-world smart home datasets demonstrate that models pretrained on this virtual data consistently improve downstream HAR performance—particularly in low-resource scenarios. Even with limited real data, these models approach the performance of those trained on full datasets, highlighting the practical value of our approach. This work illustrates a promising new direction for scalable, privacy-preserving HAR development: using LLM-guided embodied simulation to reduce dependence on costly real-world data collection.

Acknowledgments

This work was partially supported by NSF IIS-2112633 and the NSF Research Fellowship under Grant No. DGE-2039655.

Any opinion, findings, and conclusions or recommendations expressed in this material are those of the authors(s) and do not necessarily reflect the views of the National Science Foundation.

References

Abbasiantaeb, Z.; Yuan, Y.; Kanoulas, E.; and Aliannejadi, M. 2024. Let the LLMs Talk: Simulating Human-to-Human Conversational QA via Zero-Shot LLM-to-LLM Interactions. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, 8–17. Merida Mexico: ACM. ISBN 979-8-4007-0371-3.

Alam, M. R.; Reaz, M. B. I.; and Ali, M. A. M. 2012. A review of smart homes—Past, present, and future. *IEEE transactions on systems, man, and cybernetics, part C (applications and reviews)*, 42(6): 1190–1203.

Bouchabou, D.; Nguyen, S. M.; Lohr, C.; LeDuc, B.; and Kanellos, I. 2021. A survey of human activity recognition in smart homes based on IoT sensors algorithms: Taxonomies, challenges, and opportunities with deep learning. *Sensors*, 21(18): 6037.

Chase, H.; and contributors. 2023. LangChain. https://github.com/langchain-ai/langchain. Accessed: 2025-05-01.

Chernova, S.; Mynatt, E.; Rozga, A.; Simmons, R.; and Yanco, H. 2024. AI-CARING: National AI Institute for Collaborative Assistance and Responsive Interaction for Networked Groups. *AI Magazine*, 45(1): 124–130.

Cook, D. J.; Crandall, A. S.; Thomas, B. L.; and Krishnan, N. C. 2012. CASAS: A smart home in a box. *Computer*, 46(7): 62–69.

Cumin, J.; Lefebvre, G.; Ramparany, F.; and Crowley, J. L. 2017. A dataset of routine daily activities in an instrumented home. In *Ubiquitous Computing and Ambient Intelligence: 11th International Conference, UCAmI 2017, Philadelphia, PA, USA, November 7–10, 2017, Proceedings*, 413–425. Springer.

Dai, T.; Wong, J.; Jiang, Y.; Wang, C.; Gokmen, C.; Zhang, R.; Wu, J.; and Fei-Fei, L. 2024. Automated Creation of Digital Cousins for Robust Policy Learning. In *Conference on Robot Learning (CoRL)*.

Douze, M.; Guzhva, A.; Deng, C.; Johnson, J.; Szilvasy, G.; Mazaré, P.-E.; Lomeli, M.; Hosseini, L.; and Jégou, H. 2024. The faiss library. *arXiv preprint arXiv:2401.08281*.

Fung, P.; Bachrach, Y.; Celikyilmaz, A.; Chaudhuri, K.; Chen, D.; Chung, W.; Dupoux, E.; Jégou, H.; Lazaric, A.; Majumdar, A.; et al. 2025. Embodied AI Agents: Modeling the World. *arXiv preprint arXiv:2506.22355*.

Grieves, M.; and Vickers, J. 2016. Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In *Transdisciplinary perspectives on complex systems: New findings and approaches*, 85–113. Springer.

He, J.; Leng, Z.; McKay, D.; Spina, D.; and Trippas, J. R. 2025. Can We Hide Machines in the Crowd? Quantifying Equivalence in LLM-in-the-loop Annotation Tasks. *arXiv preprint arXiv:2510.06658*.

Huang, W.; Abbeel, P.; Pathak, D.; and Mordatch, I. 2022. Language Models as Zero-Shot Planners: Extracting Actionable Knowledge for Embodied Agents. In *Proceedings of the 39th International Conference on Machine Learning*, 9118–9147. PMLR. ISSN: 2640-3498.

Kolve, E.; Mottaghi, R.; Han, W.; VanderBilt, E.; Weihs, L.; Herrasti, A.; Gordon, D.; Zhu, Y.; Gupta, A.; and Farhadi, A. 2017. AI2-THOR: An Interactive 3D Environment for Visual AI. *arXiv*.

Kwon, H.; Tong, C.; Haresamudram, H.; Gao, Y.; Abowd, G. D.; Lane, N. D.; and Ploetz, T. 2020. IMUTube: Automatic extraction of virtual on-body accelerometry from video for human activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 4(3): 1–29.

Leng, Z.; Bhattacharjee, A.; Rajasekhar, H.; Zhang, L.; Bruda, E.; Kwon, H.; and Plötz, T. 2024. IMUGPT 2.0: Language-Based Cross Modality Transfer for Sensor-Based Human Activity Recognition.

Leng, Z.; Kwon, H.; and Ploetz, T. 2023. Generating Virtual On-Body Accelerometer Data from Virtual Textual Descriptions for Human Activity Recognition. In *Proceedings of the 2023 ACM International Symposium on Wearable Computers*. New York, NY, USA: Association for Computing Machinery. ISBN 9798400701993.

Li, G.; Hammoud, H. A. A. K.; Itani, H.; Khizbullin, D.; and Ghanem, B. 2023. CAMEL: Communicative Agents for "Mind" Exploration of Large Language Model Society. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Liang, D.; Li, G.; Adaimi, R.; Marculescu, R.; and Thomaz, E. 2022. Audioimu: Enhancing inertial sensing-based activity recognition with acoustic models. In *Proceedings of the 2022 ACM International Symposium on Wearable Computers*, 44–48.

Liao, Y.-H.; Puig, X.; Boben, M.; Torralba, A.; and Fidler, S. 2019. Synthesizing Environment-Aware Activities via Activity Sketches. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 6284–6292. Long Beach, CA, USA: IEEE. ISBN 978-1-7281-3293-8.

Liciotti, D.; Bernardini, M.; Romeo, L.; and Frontoni, E. 2020. A sequential deep learning application for recognising human activities in smart homes. *Neurocomputing*, 396: 501–513.

Mazzaglia, P.; Verbelen, T.; Dhoedt, B.; Courville, A.; and Rajeswar, S. 2024. GenRL: Multimodal-foundation world models for generalization in embodied agents. *Advances in neural information processing systems*, 37: 27529–27555.

Park, J. S.; O'Brien, J. C.; Cai, C. J.; Morris, M. R.; Liang, P.; and Bernstein, M. S. 2023. Generative Agents: Interactive Simulacra of Human Behavior. ArXiv:2304.03442 [cs].

Patel, M.; and Chernova, S. 2022. Proactive robot assistance via spatio-temporal object modeling. *arXiv preprint arXiv:2211.15501*.

Puig, X.; Ra, K.; Boben, M.; Li, J.; Wang, T.; Fidler, S.; and Torralba, A. 2018. VirtualHome: Simulating Household Activities via Programs. ArXiv:1806.07011 [cs].

Puig, X.; Shu, T.; Li, S.; Wang, Z.; Liao, Y.-H.; Tenenbaum, J. B.; Fidler, S.; and Torralba, A. 2021. Watch-And-Help: A Challenge for Social Perception and Human-AI Collaboration. ArXiv:2010.09890 [cs].

Qi, J.; Yang, P.; Waraich, A.; Deng, Z.; Zhao, Y.; and Yang, Y. 2018. Examining sensor-based physical activity recognition and monitoring for healthcare using Internet of Things: A systematic review. *Journal of biomedical informatics*, 87: 138–153.

Reimers, N.; and Gurevych, I. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. arXiv:1908.10084.

Savva, M.; Kadian, A.; Maksymets, O.; Zhao, Y.; Wijmans, E.; Jain, B.; Straub, J.; Liu, J.; Koltun, V.; Malik, J.; et al. 2019. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9339–9347.

Serapio-García, G.; Safdari, M.; Crepy, C.; Sun, L.; Fitz, S.; Romero, P.; Abdulhai, M.; Faust, A.; and Matarić, M. 2023. Personality Traits in Large Language Models. ArXiv:2307.00184.

Smrke, U.; Rehberger, A.; Plohl, N.; and Mlakar, I. 2025. Exploring the Feasibility of Generative AI in Persona Research: A Comparative Analysis of Large Language Model-Generated and Human-Crafted Personas in Obesity Research. *Applied Sciences*, 15(4).

Thukral, M.; Dhekane, S. G.; Hiremath, S. K.; Haresamudram, H.; and Ploetz, T. 2025. Layout-Agnostic Human Activity Recognition in Smart Homes through Textual Descriptions Of Sensor Triggers (TDOST). *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 9(1): 1–38.

Wang, G.; Xie, Y.; Jiang, Y.; Mandlekar, A.; Xiao, C.; Zhu, Y.; Fan, L.; and Anandkumar, A. 2023. Voyager: An Open-Ended Embodied Agent with Large Language Models. ArXiv:2305.16291 [cs].

Xia, F.; Zamir, A. R.; He, Z.; Sax, A.; Malik, J.; and Savarese, S. 2018. Gibson env: Real-world perception for embodied agents. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 9068–9079.

Yonekura, H.; Tanaka, F.; Mizumoto, T.; and Yamaguchi, H. 2024. Generating Human Daily Activities with LLM for Smart Home Simulator Agents. In *2024 International Conference on Intelligent Environments (IE)*, 93–96. Ljubljana, Slovenia: IEEE. ISBN 979-8-3503-8679-0.