Unseen Activity Recognitions: A Hierarchical Active Transfer Learning Approach

Mohammad Arif Ul Alam[1], Nirmalya Roy[1]
[1]Department of Information Systems, University of Maryland Baltimore County
alam4@umbc.edu, nroy@umbc.edu

***Abstract*—Human activity recognition (AR) is an essential element for user-centric and context-aware applications. While previous studies showed promising results using various machine learning algorithms, most of them can only recognize the activities that were previously seen in the training data. We investigate the challenges of improving the recognition of unseen daily activities in smart home environment, by better exploiting the hierarchical taxonomy of complex daily activities. We first (a) design a hierarchical representation of complex activity taxonomy in terms of human-readable semantic attributes, and (b) develop a hierarchy of classifiers which incorporates a cluster tree built on the domain knowledge from training samples. Though this model is rich in recognizing complex activities that are previously seen in training data, it is not well versed to recognize unseen complex activities without new training samples. To tackle this challenge, we extend Hierarchical Active Transfer Learning (HATL) approach that exploits semantic attribute cluster structure of complex activities shared between seen (source) and unseen (target) activity domains. Our approach employs transfer and active learning to help label target domain unlabeled data by spawning the most effective queries. We evaluated our approach with two real-time smart home systems (IRB #HP-00064387) which corroborates radical improvements in recognizing unseen complex activities.**

I. Introduction

AR techniques have improved over the past few years significantly in terms of accuracy and reliability [1], [3], [5]. Now it is a prime time to develop AR for community or a population and therefore, its generalizability and scalability are of utmost importance. Though, there has been extensive research on complex AR using a multitude of sensors and various machine learning algorithms [7], [8], [11], existing approaches cannot recognize a previously *unseen* activity, if there were no properly labeled training samples. Labeled examples are often time consuming and expensive to obtain, as they require a lot of efforts from test subjects and domain experts. However, labeling previously unseen activities is more expensive than the seen ones, because unlike the seen activities, labeling unseen activities needs definition of activities and start/end points on each attempt. For example, we experienced two outcomes from a real experience of data collection performed in our previous work [2]: (i) Data annotation is 4 times more costly than the data collection for *seen* activities in terms of duration; and (ii) Data annotation is 6 times more costly than the data collection for *unseen* activities. A recent survey on daily activity events showed that there are at least 462 different activities that people do in their daily lives [4]. If we consider the diversity of people and cultures that were not covered by the survey, the actual number of daily activities is more likely even larger which eventually transmutes the existing classification as an unseen AR problem. These fundamental problems signify the importance of extending existing AR model to discover new activities (*unseen*) and generalize the AR to achieve robustness and scalability.

Unseen AR has been explored recently for low-level daily activities (micro activities such as ‘walking’, ‘sitting’, ‘standing’ etc.) using active learning [13], Multi-class Positive-Unlabeled Learning [14], hierarchical classifier [12] and transfer learning [27]. Though all of these proposed methods promoted significant performances in unseen AR in terms of labeling cost and recognition accuracy, the problem becomes more complex and error prone in the case of complex daily activities (macro activities that consist of sequence of low-level activities such as ‘cooking’, ‘watching tv’ etc.) where these are subject to proper representation of low-level activity components [13]. This fundamental problem of generalization necessitates a structured representation of complex activities to aid the application of machine learning methods that aim to reduce labeling costs (active learning and transfer learning) and classify unseen AR problem. A significant number of researchers showed that hierarchical representation can provide a more robust solution for complex AR where complex activities were considered be related to lower level activities in a hierarchical tree structure [9], [12], [13]. However, extending the framework to recognize hundreds of different human activity classes (either *seen* or *unseen*) using the available labeled datasets needs more rigorous modeling addressing the underpinning scalability challenges to reach maturity.

In light of existing problems, there are two major research questions we aimed to answer:

- Can we represent any complex activity in a hierarchical tree structure? If so, how can we take advantages of available well-labeled activity datasets to learn hierarchical tree structure of complex activities and transfer that knowledge to label *unseen* new activities?
- If opportunities are available to probe users for labeling task, how to reinforce the query with maximal informativeness and minimal intervention using active learning in conjunction with transfer learning by percolating the hierarchical activity knowledge source?

AR models need to generalize over variations in environmental and resident characteristics. However, robust machine learning approaches need a significant amount of labeled training dataset. In this paper, we hypothesize that clustering activities into a hierarchical activity taxonomy can facilitate analysis of activity patterns from multiple similar data sources and such a taxonomy can also be used to scale AR. In this regard, we design a hierarchy of classifiers, each of which helps distinguish between child nodes at a particular location in the hierarchy. The activity taxonomy generated by our approach is employed in the context of transfer learning across generalized settings to tackle the challenges of learning an activity model in a new setting. To do this, we first determine the specific positions of the activity labels in the hierarchy using few labeled samples, which are then combined with labeled samples of similar activities to initiate the taxonomy of learning for the new activity model. To scale the hierarchical transfer learning [28] based human AR model towards *unseen* AR, we combine hierarchical sampling for active learning (HSAL) [35]. More specifically, we make the following **key contributions**:

- We first design and represent human activities using semantic attributes. We then postulate a data-driven approach that calculates the similarity of predefined and *unseen* activities and helps generate complex activity taxonomy by creating a hierarchical cluster of activity labels generated from available datasets.
- We employ this postulated hierarchical complex activity taxonomy, build a hierarchical classification tree and propose an uncertainty metric (contextual informativeness) to distinguish unseen activities from previously seen ones. Employing the uncertainty metric, we augment transfer learning and active learning to minimize the user intervention and maximize the AR performance gain.
- We evaluate the scalability, generalizability, adaptability, and statistical significance of our proposed taxonomy and uncertainty metric assisted hierarchical active transfer learning approach in two real-time AR datasets with diverse settings and populations.

II. Related Works

This paper builds on previous works on complex daily AR using machine learning, such as active and transfer learning approaches. Here we compare and contrast our contributions with the most relevant existing literature.

A. Complex Activity Recognition

A significant amount of research has been performed on recognizing complex human activities in a smart home setting equipped with distinct types of sensors such as cameras [16], RFID tags [18], passive infrared sensors [11] or combination of multiple sensors [1]). In terms of theories, most of the researches focused on solving complex graphical models to address specific AR problem domains [11]. In past, we used coupled hidden markov model [7] and later proposed coupled hierarchical dynamic bayesian network to improve complex AR performance in presence of multiple occupants [1]. We also proposed Beacon radio signal based light-weight ensemble learning techniques to recognize complex activities [1] in past. Imitation learning has been used to reduce the activity learning framework into simple regression learning model to predict complex activities [15]. In this paper, we propose a hierarchical clustering based AR approach that aggregates the available labeled data sources to generate a hierarchical taxonomy of complex activity patterns. Researchers have also proposed a hierarchical taxonomy based complex AR before [12]. These includes hierarchical activity pattern clustering based single classifier (temporal artificial neural network framework [19]) or multiple hierarchical classifiers [12]. Our framework generates multiple hierarchical clustered classifiers that facilitate robust complex AR irrespective of hierarchical activity tree structure or sub-tree position.

B. Active Transfer Learning in AR

Active learning has been used in activity prediction domain by many researchers before. [20] proposed a novel active learning technique that exploits the informativeness of the individual activity instances but also utilizes their contextual information during the query selection process in a continous video stream. [13] proposed a new semantic attribute representation of complex daily activities and developed a two layer zero-shot learning algorithm for wearable sensor based AR. A linear k-means clustering based active learning technique to recognize passive infrared sensors assisted smart home activities in presence of single occupant has been proposed in [21]. A bayesian active learning has been explored to recognize complex smart home activity patterns in [25]. It is well known that the hierarchical Bayesian framework can be adapted to sequential decision problems and it has been shown more recently that it provides a natural formalization of transfer (reinforcement) learning [26]. Transfer learning has also been investigated in complex AR domains in past. [23] proposed a heterogeneous transfer learning for AR using heuristic search techniques. Recently researchers have shown that active learning combined with transfer learning can improve the performance significantly and investigated active transfer learning for a cross-system recommendation, since a newly launched system has a cold-start problem, where existing rating information is available [28].

C. Unseen Activity Recognition

There have been very few researches focused on *unseen* AR problem before. NuActiv proposed an outlier-aware hierarchical probabilistic framework using active learning techniques to solve the problem of *unseen* low-level AR problem [13]. mPUL proposed a Multi-class Positive and Unlabeled Learning approach that significantly reduces the false positives of simple smoking activities and *unseen* gesture recognition [14]. Active learning has been used to reduce the required number of user annotations by asking users to label only “informative” instances of unlabeled low-level activity datasets [29], [30]. FE-AT (Feature-based and Attribute-based learning) approach leveraged the relationship between existing and new activities to compensate for the shortage of the labeled data and applied collaborative probabilistic method to solve *seen-unseen* AR problem [32]. Although above works proposed different methods to solve *unseen* low-level AR problem, *unseen* complex AR has never been investigated. We focus on building hierarchical taxonomy of data from binary sensor assisted available smart home activity data sources. Additionally we build a semantic attribute representation based hierarchical cluster tree pattern of complex activities and incorporated hierarchical sampling for active learning combining with transfer learning. While this model is rich in terms of previously *seen* AR, our *unseen* activity pattern discovery method improves the *unseen* AR technique significantly.

III. System Overview

We extend our previously proposed frameworks [1], [21] in several distinct ways.

A. Problem Scenarios

Towards the generalizability of supervised AR framework, we approach two different types of activities: *seen* and *unseen*. *Seen* activities are those that are available densely in the well labeled training samples. On the other hand, *unseen* activities are those that are available with sparsely in the training samples. The number of required amount of well-labeled training samples is called budget of that particular activity class. We build an uncertainty heuristic (contextual informativeness) using the estimated budget (b) such a way that active learning followed by transfer learning will be chosen if and only if *unseen* activities have been discovered in the test data points. Finally, we design and extended a hierarchical active transfer learning framework that can identify an *unseen* activity from the unlabeled or poorly (zero or wrongly) labeled dataset and take advantages of existing dataset knowledge to label *unseen* human activities with the minimum labeling requirement.

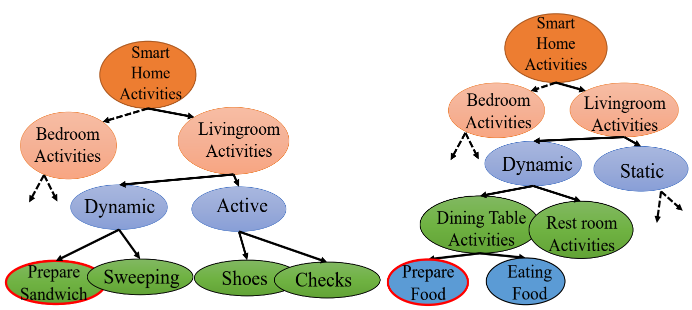

Fig. 1. Hierarchical tree structure of livingroom activities in a home scenario generated from our previous works (left one from [2] and right one from [1])

B. Design Considerations

Complex daily activities can be represented as hierarchical tree structure which can be learned using hierarchical clustering of smart home sensor events and low-level activity attributes [12], [13]. For example, Fig. 1 shows two partially represented hierarchical living room activity trees generated from activity dataset collected in our previous works ([1], [2]). Here, 'prepare sandwich' and 'prepare food' complex activities are most likely the same activity according to the tree structure. We see a tiny difference between these two hierarchical tree structures: one extra layer has been found (dining table activities and rest of the room activities). If the first one has been trained in the source dataset and the other one is found in the target dataset, it will be considered as unseen AR problem. If we can map these similarity measures into a middle-ware model (transfer learner), we can use existing well-labeled dataset to strengthen the poorly labeled data points. However, the tree structures not necessarily be always similar i.e., location may not always be the children of root as we are creating the tree from randomly sampled dataset.

In this paper, we consider of having small amount of labeled activity instances and a huge number of unlabeled activity instances as source and target dataset respectively. We then represent source dataset into hierarchical cluster tree (similar to Fig. 1) and train a hierarchical transfer learning classifier mapping with target hierarchical cluster tree generated from target dataset. We then design Contextual Informativeness (CI) that decides how to choose transfer and active learner based on the labeling nature (poorly or strongly labeled).

C. System Architectural Overview

Fig 2 explains the basic steps in our overall *unseen* AR pipeline: (1) 'Pre-processing' gathers and extract multi-modal sensor data features from the object, wearable, and ambient motion sensors associated with our smart home; (2) 'Hierarchical activity classifier creation' performs semantic attribute detection [13], knowledge-based hierarchical clustering, hierarchical cluster tree generation and hierarchical classification model creation; (3) 'Inference engine' estimates the hierarchical classifier's performance uncertainty and based on that it either infers complex activity (for *seen* activities) or activates hierarchical active transfer learning (for *unseen* activities).

IV. System Design and Algorithms

In this section, we describe the design of semantic attribute detection based hierarchical clustering, hierarchical transfer learning and active learning for hierarchical sampling.

A. Semantic Attribute Classification

Researchers showed that though ambient sensor readings and postural activities are good enough to recognize complex activities, the addition of hand gesture improves complex AR performance significantly [1], [10]. Inspired by this, we follow three layered architecture to detect semantic attribute: (a) Hand gesture recognition [33]; (b) Postural AR [36], and (c) Semantic attribute detection [13].

Hand Gestural and Postural AR: For hand gesture recognition, we define 18 standard hand gestures and build hand gesture dictionary [33]. First, we apply segmentation (2 seconds window with 50% overlap) and filtering (low band-pass filter with 0.02Hz cut-off) on wrist-worn accelerometer (ACC) to eliminate postural activity effects. Then we extract 57 statistical features from ACC sensor axises (x, y, z), apply correlation feature selection technique to select top 12 features,

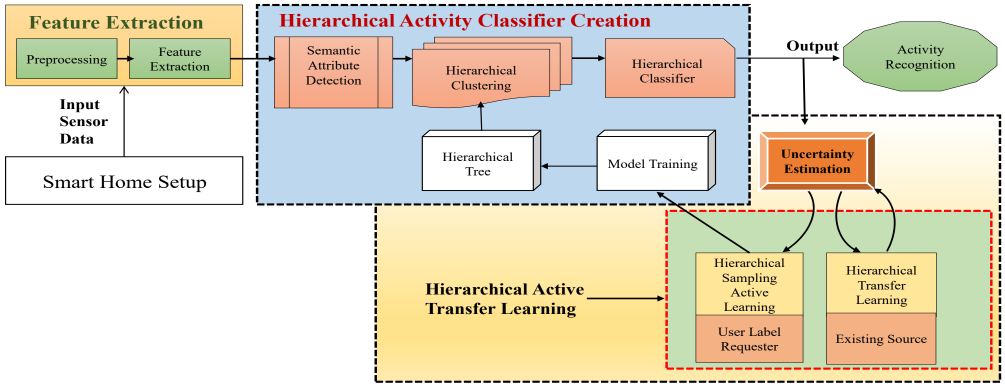

Fig. 2. An Overview of our proposed Unseen Activity Recognition Framework Comprising of 3 core modules with several sub-modules.

train a Support Vector Machine based SMO algorithm [22] (optimal one) and validate model performance with a 10-fold cross-validation [33]. Similarly, for postural AR, we first define 5 postural activities (sitting, standing, walking, lying, cycling and running), apply segmentation (4 seconds window with 50% overlap) and filtering (high band-pass filter with 0.02Hz cut-off) to eliminate hand gestural activity effects. Then, similar to gestural case, we extract 12 features, train a Random-Forest ensemble learning classification algorithm (optimal one) and validate with 10-fold cross-validation [36].

Semantic Attribute Detection: Semantic attributes are microscopic components of complex daily activities. For example, 'sweeping room' complex activity consists of three semantic attributes: user retrieves broom from supply closet, sweeps the floor and returns broom to supply closet. At first we define M gestural, N postural and O semantic attributes for our entire dataset that create an $M \times N \times O$ attribute-activity matrix where each element a_{ijk} represents the transition line from i gesture to k attribute via j posture. We define each element as binary value indicating whether such association exists or not. We define the attribute-activity matrix manually. We feed the training samples into Support Vector Machine (SVM) classifier comprises with ambient sensor features which can help detect instant semantic attributes on the test dataset. The idea is, first transform low-level features x into a vector of semantic attributes a in the attribute space A. Each attribute corresponds to a complex activity which are obtained from attribute-activity matrix. We map every complex activity Y in the attribute space A called transition line and train SVM model with the extracted mapped features for semantic attribute detection in training phase.

B. Designing Hierarchical Classification Tree Model

It has been proven that insights about complex activities can be exploited and models can be learned effectively by organizing complex activities in a cluster hierarchy [12]. In the hierarchy, each leaf node represents a unit of complex activity, and internal nodes represent unions of the activities that reside in the sub-tree rooted at the node. Each leaf-level complex activity composes of semantic-attributes and ambient sensors (object and motion sensors) features as feature set.

1) Training: The basic strategy is to divide the data into two subsets at every hierarchical level. We use a bottom-up or agglomerative method for creating hierarchical clusters [37] where we merged every two lower nodes into one upper node cluster using the following distance function:

$$d_{ave} = \frac{1}{|V_1||V_2|} \sum_{v_1 \in V_1} \sum_{v_2 \in V_2} d(v_1, v_2) \quad (1)$$

Here v_1 and v_2 are the two lower level activity nodes and $d(v_1, v_2)$ represents the Euclidean distance. After several iterations, we finally generate hierarchical cluster tree for both source dataset. We aim to train a binary classifier at every internal node of the cluster tree to differentiate between the node's children towards creating a hierarchical classifier. In the classifier, the data points for a parent node are obtained by combining the instances of its children. We employ an under-sampling method in our hierarchical classifiers to avoid class imbalance situation [41] where we selectively sample instances from the children instead of combining all instances from child nodes into the parent node. To explain that, let consider an internal node V with two children, v_1 and v_2, with n_1 and n_2 numbers of data points, respectively. Our hierarchical classifier selects $max(n_1, n_2)$ data points to assign to parent node V and the remaining half are those which are closest to this decision boundary. Samples that are farthest from the decision boundary represent the unique characteristics of the individual children. In contrast, points that are closest to the decision boundary represent the characteristics of the combined dataset [42]. Selecting the data points for the parent in this manner helps the algorithm to partially preserve the data distribution of its children. We use support vector machine (SVM) as our base classifier that generates a complete multi-class Hierarchical Support Vector Machine (HSVM) model [6].

2) Testing: At the beginning, all of the classes are considered to be true class. At every node, we apply decision function to test the input and eliminate the nominees that do not exist in the region (positive or negative). Following the branches that indicate the same labels as the result of the decisions, we end up with the predicted class.

C. Designing Extended Hierarchical Active Transfer Learning

We incorporate semantic attributes into the classification tree and imposed a new heuristic method (i.e., contextual informativeness) to excel the existing HATL method in selecting active and transfer learning.

Problem Formulation: We first consider X as data points from D distribution which consists of properly labeled and poorly labeled data points. Now, let f be a labeling function thus the label of a point x is $f(x)$. We will view learning as searching for a hypothesis $h \in H$. T denotes the binary tree representing a hierarchical clustering of the data points in X. For any node (or cluster) $v \in T$, T_v denotes the hierarchy of nodes (or subtree) rooted at v ($T = T_{root}$). By descending down T_v, we finally reach a leaf node x that represents semantic attribute. We consider the set of points $X(v)$ is associated with arbitrary node v. For each node v, $X(v) = X(u_1) \cup X(u_2)$ where u_1 and u_2 are two child nodes of v. Trivially, $X(x) = x$ for leaf nodes. P is a pruning of the complete tree T. Now, let consider pruning P_v is a subset of non-overlapping nodes of T_v that contains all points associated with $v : X(v) = \bigcup_{u \in P_v} X(u)$. A labeling function $L(v)$ is a mapping of a node v to a label (e.g., $L(v) : v \mapsto \pm 1$).

Contextual Informativeness (CI): Contextual Informativeness is the measure of the overall confidence of the HSVM classification tree. This measure has been calculated by assigning a confidence score (probability) to each path at each node in the hierarchy, all nodes, and all paths are searched, and the test input is assigned to the class with the highest probability at the bottom node. We pre-determined two threshold values (τ_t, τ_a) using trial-and-error method on target dataset to choose transfer learner and active learner respectively in the heuristic. CI works in three steps:

- *Seen activity detection*: After the HSVM classification, CI has been used to decide whether the amounts of training samples are sufficient for the particular class label prediction i.e., to detect whether the class label is previously seen or not. It first calculates CI for the class label prediction. If $CI > \tau_t$, then target data point does not go for transfer learner to label class and finalize y_T as target data point (Line 4 at Algorithm 1).
- *Transfer Learner Selection*: If $CI < \tau_t$, then our framework estimates the target data point as *unseen* or poorly labeled and selects hierarchical transfer learning to update node statistics. The $UpdateNodeStatistics$ provide a newly updated label y_T that has been used to estimate CI again and repeat whether to choose transfer learner again based on $CI < \tau_t$ for $\mathcal{N}$ iteration. Now if $CI > \tau_t$, we choose transfer learner predicted y_T as target data point label (Line 4 at Algorithm 1).
- *Active Learner Selection*: After $\mathcal{N}$ iteration, if transfer learner fails to satisfy $CI > \tau_t$ condition, we decide that the input instant is completely new activity that needs to be labeled by the experts. Now, for HSAL, if $CI < \tau_a$ or the target label budget $B > 0$, we select active learner to label target data point which selectively asks experts to label data point (Line 13 at Algorithm 1).

Hierarchical Active Transfer Learning: We extend HATL incorporating CI uncertainty measures [28] (as shown in Algorithm 1). It takes a cluster tree as an input built on source and target data and uses cluster structure and label information from both domains to impute labels on the full data set (source and target). At the beginning, it relies heavily on source dataset but gradually incorporates feedback from target label inquiries based on CI measure to refine both its clustering and label imputation. This accelerates the learning process and mitigates the cold start problem.

At first, we have semantic attribute labeled data X_S from the source domain, in addition to unlabeled target data X_T. Our proposed extended HATL framework can leverage X_S and a limited number of queried target labels to help impute labels for the full X_T, which we can then use to train an accurate classifier for the target domain. We begin with cluster tree T over $X_S \cup X_T$, a label budget B, and batch size b, and target label $oracle$. On line 2, we initialize P to the $root$ of T and L to be an arbitrary label (positive class). Then, in lines 6-8, we update the label proportion estimates for all nodes, based on labeled source points. The $UpdateNodeStatistics(x, y, v)$ subroutine performs this update for all nodes along the path from x to v in T_v. On line 9, we update P and L using the $GetPruningAndLabeling(T_v)$ subroutine, which recursively splits nodes in T_v that have high label disagreement. If the budget $B > 0$ or $R < CI$, we run $HSAL$ on the mixture of source and target data but using the updated P and L (lines 16-18). Finally, we impute labels for all source and target points in lines 19-21 and output the fully labeled data sets. The $UpdateNodeStatistics$ and $GetPruningAndLabeling$ subroutines are implemented as in HSAL.

Hierarchical Sampling For Active Learning: We use HSAL for sampling in our active learner [35]. HSAL begins with a cluster tree T over N points in X and a label query budget of B. At all times, it maintains a current P and L for T , with initial values of $P = root$ and $L(root) = +1$. Each iteration of HSAL consists of four main steps: (i) It queries labels for a batch of b unlabeled points, (ii) It estimates the label proportions in each $v \in P$, (iii) it updates P by replacing any node v with its children if it has a high label disagreement and finally (iv) It updates L by letting $L(v)$ equal the estimated majority label for each v. Label queries are made by choosing first a cluster v from P and then an unlabeled point x from $X(v)$.

Upon termination, HSAL produces a fully labeled data set $Y = (x, \hat{y}(x)) : \forall x \in X$ by assigning $\hat{y}(x) = L(v)$ to each $x \in X(v)$. Y can be used to train HSVM. This *label imputation* step helps avoid sample selection bias suffered by other active learning algorithms [35]. Here we have to estimate P and L by minimizing the following error function:

$$\epsilon(P, L) = (1/N) \sum_{v \in P} \sum_{x \in X(v)} (L(v) \neq f(x)) \qquad (2)$$

Algorithm 1 Extended Hierarchical Active Transfer Learning

procedure HATL(T, X_T, X_S, B, b, O) ▷ X_T = Unlabeled target data points, T = Hierarchical cluster tree of X_T, X_S = Labeled source data, B = Target label budget, b = Batch size, O = Target label oracle, τ_t,τ_a
  $P \leftarrow \{root\}, L(root) \leftarrow +1$
  $(CI, y_T) \leftarrow HSVM(P, L, X_S, X_T)$ ▷ HSVM classification with contextual informativeness CI
  while $CI < \tau_t$ **do** and $n = 1 : \mathcal{N}$ ▷ Selection of Transfer Learner
    $\mathcal{T} \leftarrow Init(CI, y_T, P, L)$ ▷ Initialize Transfer Learner
    for all (x, y) where $x \in X_S \& y = f_s(x)$ **do**
      UpdateNodeStatistics($x, y, root$) ▷ counts, estimates and bounds, admissibility, score for transfer classifier
    end for
    $(P, L) \leftarrow GetPruningAndLabeling(T_v)$
    $(CI, y_T) \leftarrow HSVM(P, L, X_S, X_T)$ ▷ Recalculate Contextual Informativeness
  end while

  if $B > 0 \mid\mid CI < \tau_a$ **then** ▷ Select Active Learner
    $(P, L) \leftarrow HSAL(T, \{X_T, X_S\}, P, L, B, oracle)$
  end if
  for all $v \in P$ **do**
    $\hat{y} \leftarrow L(v)$ for each $x \in X(v)$
  end for
    $y_S = \{(x, \hat{y}(x)) : \forall x \in X_S\}$
    $y_T = \{(x, \hat{y}(x)) : \forall x \in X_T\}$
  end procedure

Where $\epsilon(P, L)$ is the minimum possible label imputation error with maximum label queries B. Algorithm 2 shows the basic steps of HSAL.

V. Experimental Results

We present experimental results and detail comparison analysis using two real-time activity data traces.

A. Retirement Community Center (RCC) Dataset

The first dataset we have used is collected in a retirement community center (IRB #HP-00064387).

1) Smart Home Setup: We used our previously developed real testbed smart home system, *PogoPlug* [1], [2], where we customized Cloud Engine PogoPlug Mobile [34] base station firmware to integrate with WiFi (connect ambient sensors) and Bluetooth (connect wrist-band) protocol. The smart home components are as follows (as shown Fig. 3): PogoPlug base server with a continuous power supply, 3 binary PIR sensors in three different rooms (kitchen, livingroom and bedroom), 7 binary object sensors attached with closet door, entry door, telephone, broom, laundry basket, trash can and trash box, three IP cameras in the appropriate positions to collect the ground truth data and an Empatica E4 [17] wrist

Algorithm 2 Hierarchical Sampling Active Learning

procedure HSAL(T, X, P, L, B, b, O) ▷ T = Hierarchical cluster tree, X = Data points, P = Initial pruning, L = Initial labeling, B = Active learning budget, b = Batch size, O = Oracle
  $q \leftarrow 0$ ▷ current number of queries
  while $q \leq B$ **do**
    $Q \leftarrow \{\}$ ▷ list of queried nodes
    for $i = 1$ *to* b **do**
      $(v, x) \leftarrow ChooseNextQuery(P)$
      $y \leftarrow O(x), Q \leftarrow Q \cup \{v\}$
      for all $x \in X(u)$ where $u \in T$ **do**
        UpdateNodeStatistics(u, y, u) ▷ counts, estimates and bounds, admissibility, score
      end for
    end for
    for all $v \in Q$ **do**
      $(P_v, L_v) \leftarrow GetPruningAndLabeling(T_v)$
      $P \leftarrow (P \setminus \{v\}) \cup P_v$ ▷ replace v with $v \in P_v$
      $L(u) \leftarrow L_v(u)$ for each $u \in P_v$
    end for
  end while
  for all $v \in P$ **do**
    $\hat{y}(x) \leftarrow L(v)$ for each $x \in X(v)$
  end for
end procedure

band (integrated with a triaxial accelerometer at 32Hz) on the participant's dominating hand.

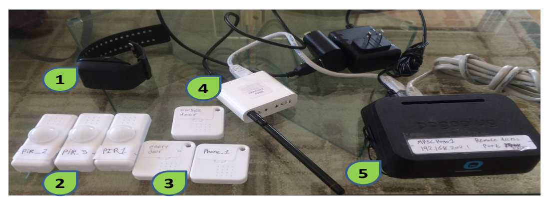

Fig. 3. Pogoplug [1], [2] smart home system devices: (1) Empatica E4 Wristband, (2) Wireless Sensor Tag Passive Infrared Sensors, (3) Wireless Sensor Tag Object Sensors, (4) Ethernet Tag Manager, (5) Cloud Engine Pogoplug Mobile

2) Data Collection: We recruited 22 participants for this study (19 females and 3 males) with age range from 77-93 (mean 85.5, std 3.92) from a continuing care retirement community with the appropriate institutional IRB approval and signed consent. The gender diversity in the recruited participants reflects the gender distribution (85% female and 15% male) in the retirement community facility. We performed a site survey in the retirement community center where we investigated the nature of the apartments occupied by the participants to generalize our setup and asked them about their daily performed activities. Then we listed all activities and narrowed down to 13 common activities. On the data collection day, participants are given a wrist band to wear on their dominant hand, and concurrently another trained IT graduate student have the PogoPlug smart home setup in participants'

TABLE I
SCRIPTED AND UNSCRIPTED ACTIVITIES

Scripted activities
(1) wearing shoes, (2) wearing jacket, (3) combing hair, (4) wash hands, (5) tooth brushing, (6) receive phone call, (7) making phone call, (8) answer door, (9) sweeping, (10) taking out trash, (11) pay bills, (12) fold laundry and (13) prepare sandwich.
Unscripted activities
(1) sleeping, (2) watching TV, (3) reading books, (4) eating, (5) watering plants and (6) playing with pets.

own living environment (smart home setup time 15-30 minutes with mean 20.5 minutes). The participants are instructed to perform pre-defined 13 *scripted ADLs* with the presence of only the evaluator. Then, the evaluator also leaves the apartment with video recording turn on for a couple of more hours for collecting participants' natural *unscripted activities* (as shown in Table. I). Two trained and IRB recruitment approved graduate students are engaged to annotate activities (postural, gestural and scripted ADLs) on scripted and unscripted activity sessions. Two more graduate students are engaged to validate the annotations on the videos. In overall, we are able to annotate 13 scripted activities (291 samples) labeling for each participant; 18 hand gestures (43561 samples) and 4 postural activities (43561 samples) labeling on scripted and unscripted datasets; and 6 *unseen* complex ADLs (689 samples) from unscripted dataset. We previously named *unseen* activities as 'Random' type [1]–[3]. We named our dataset as RCC Dataset (Retirement Community Center). We have designed 87 semantic attributes for RCC Dataset, labeled them well (13590 samples) and generate the Activity-Attribute Matrix*.
.

B. TU Darmstadt Dataset

In our second case, we used publicly available real-time activity dataset, TU Darmstadt dataset [24] which includes 34 daily life activity classes collected from one subject for seven days. The sensor data were collected using a wearable sensor platform with a three-axis accelerometer (ADXL330) worn on the wrist and the hip of the subject with a sampling rate of 100Hz. We defined 17 semantic attributes and generated Activity-Attribute Matrix**. .

C. Evaluation Methodologies

For overall performance across all classes, the accuracy is computed as the number of correctly recognized samples divided by the number of all samples in the test set. We use 10-fold cross-validation method to detect semantic attributes and supervised AR i.e., we randomly partition the entire dataset into 10-equal sized subsamples. Of the 10 samples, a single sub-sample is retained as the validation data for testing the model, and the remaining 9 sub-samples are used as training data. The cross-validation process is then repeated 10 times (the folds), with each of the 10 sub-samples used exactly once as the validation data. We consider following baseline frameworks in evaluating the performance of our framework in similar scenarios:

- L-Active: Linear K-means clustering based active learning with Query [21].
- HBATL: Hierarchical Bayesian Active Transfer Learning [25].
- HSAL: Hierarchical Sampling for Active Learning [35].
- JOTAL: Joint transfer and batch-mode active learning [40].
- Our approach: Our Approach with semantic attribute detector and contextual informativeness.

We implement our proposed method and all baseline methods in MATLAB (Package and dataset will be publicly released).

D. Case Study I: RCC Dataset

In the case of RCC Dataset, we extract wrist-worn ACC sensor signal to detect hand gesture and postural activities, extract features of the object and ambient sensor signals and impute them into our framework.

1) Supervised Classification Performance: For hand gesture, postural activity and semantic attribute recognition frameworks evaluations, we use 10-fold cross-validation as stated before to test performances. We achieve 91.8% (FP rate 6.8%), 75.9% (FP rate 12.9%) and 87.3% (FP rate 8.5%) accuracies for hand gesture, postural activity and semantic attribute recognition model, respectively. For supervised complex AR, we consider supervised parts of each baseline framework, i.e., L-Active (linear K-Means clustering), HBATL (Hierarchical Bayesian Network), JOTAL (joint probability of classification decisions) and our approach (hierarchical SVM classification). Fig 4 shows that supervised complex AR accuracy for the supervised version of our baseline activities. We notice that our HSVM classification accuracy outperforms ($\approx$ 6% more accuracy than a nearby framework) other frameworks achieving 91% accuracy (FP rate 2.5%) for all activities (both scripted and unscripted). We also notice that scripted activity classification performances are higher (95% for our method) than unscripted activities (86%).

2) Unseen Activity Classification Experiments: To evaluate *unseen* AR performance on our entire dataset, we combine scripted and unscripted activity sessions into one dataset and follow leave-two-class cross-validation method to evaluate overall *unseen* activity performance which is mostly used for *unseen* class classification domain [38], [39]. The validation scheme is used for recognizing *unseen* classes that do not have any sample in the training set. The traditional 10-fold cross-validation is not applicable to *unseen* class recognition because it does not leave out all samples of certain *unseen* classes in the training step so that every class will have some samples in the training set. Fig 6 illustrates the accuracy measure for all baseline frameworks accuracy in terms of leave-two-out cross-validation on our entire dataset which clearly depicts that

* see a full list of the activity-attribute matrix for RCC Dataset http://userpages.umbc.edu/~alam4/icdcs/rcc_matrix.pdf

** see a full list of the activity-attribute matrix for TU Darmstadt Dataset http://userpages.umbc.edu/~alam4/icdcs/tu_matrix.pdf

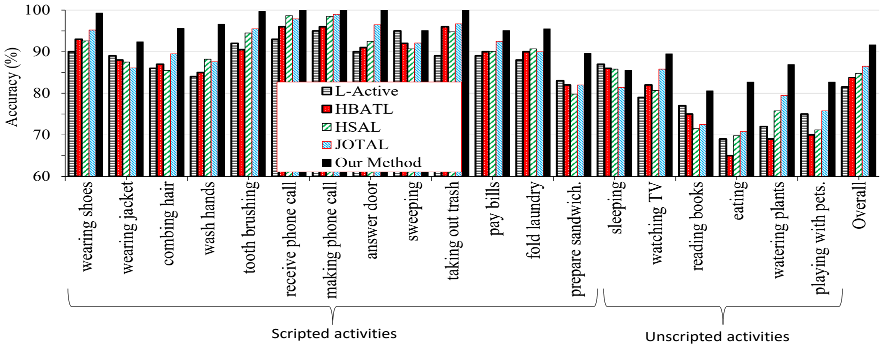

Fig. 4. RCC Dataset: Supervised classification accuracy for scripted and unscripted activities comparison with the baseline methods' supervised parts where our framework's outperforms in each activity case

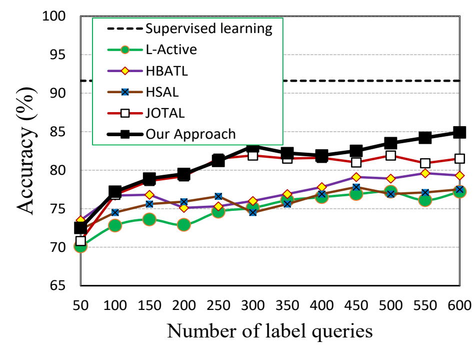

Fig. 5. RCC Dataset: Test set AR accuracy comparisons with baseline through 600 queries

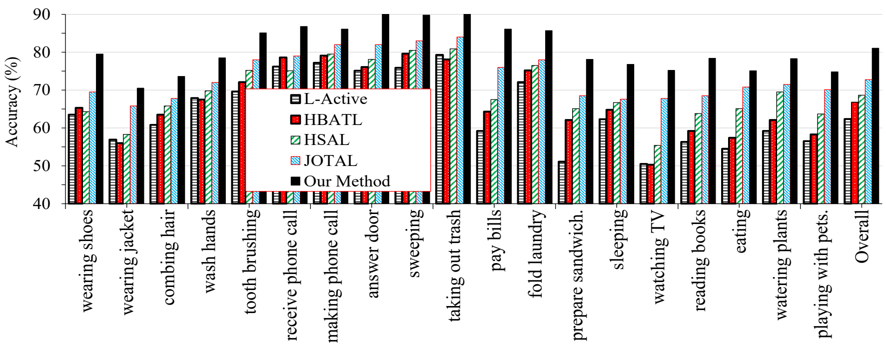

Fig. 6. RCC Dataset: Unseen Activity Classification Accuracy with leave-two-out cross-validation method and comparison with baseline methods.

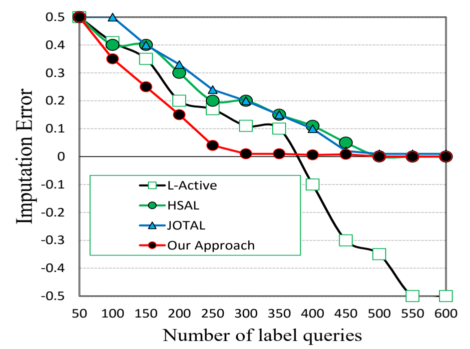

Fig. 7. RCC Dataset: Test set label imputation error comparisons with baseline through 600 queries

our framework outperforms all other frameworks in significant magnitude (9% more accurate that nearby baseline frameworks) achieving an overall accuracy of 81%. It should be noted that we did not train our model until the label confidence reaches 85% in the above evaluation (the standard used in [13]).

Fig 5 and Fig 7 illustrate the accuracy measures and label imputation error respectively for baseline frameworks in terms of increasing number of label queries which clearly depict that our framework outperforms all baseline frameworks in both measures. Fig 11 shows that AR accuracy gradually degrades as the number of unseen classes in the testing data increases (i.e. the number of seen classes in the training data decreases). This is in accordance with the expectation, because it gradually becomes difficult for the system to generalize to a large number of *unseen* activity classes based on only a few seen classes. However, our framework also outperforms other baseline frameworks in this case as well. We can see that all of the complex activities which are integrated with object sensors provide impressive recognition performance even with the baseline methods for both in supervised (Fig. 4) and unseen (Fig. 6) AR. For instance, 'making phone calls' (telephone sensor) 'answer the door' (door sensor), 'sweeping' (sensor attached to broom), 'fold laundry' (sensor attached to laundry basket) etc.

E. Case Study II: TU Darmstadt

In case of TU Darmstadt, we extract wrist-worn ACC sensor signal and hip-worn ACC sensor signal generating a feature set of 24 features (12-ACC sensor features from each node) to detect our defined semantic attributes.

1) Supervised Classification Performance: Similar to RCC dataset, we use 10-fold cross-validation method to detect semantic attributes and supervised AR. We achieve 83.4% (FP rate 10.8%) accuracy for our 17-classes semantic attribute recognition model and achieve and 80% (FP rate 6.2%) accuracy for 34-classes daily AR. For supervised complex AR, we consider supervised parts of each baseline frameworks.

2) Unseen Activity Classification Experiments: To evaluate *unseen* AR performance on TU Darmstadt dataset, we follow leave-two-class cross-validation method to evaluate overall *unseen* activity performance. Fig 8 illustrates the accuracy measure for all baseline frameworks in terms of leave-two-out cross-validation on TU Darmstadt dataset which clearly depicts that our framework outperforms all other frameworks in significant magnitude achieving an overall accuracy of 79%. We did not train our model until the label confidence reaches 85% in the above evaluation.

Fig 9 illustrates the accuracy measures for baseline frameworks which clearly depicts that our framework outperforms all baseline frameworks in terms of number of label queries. Fig 10 shows the accuracy graphs of baseline frameworks in

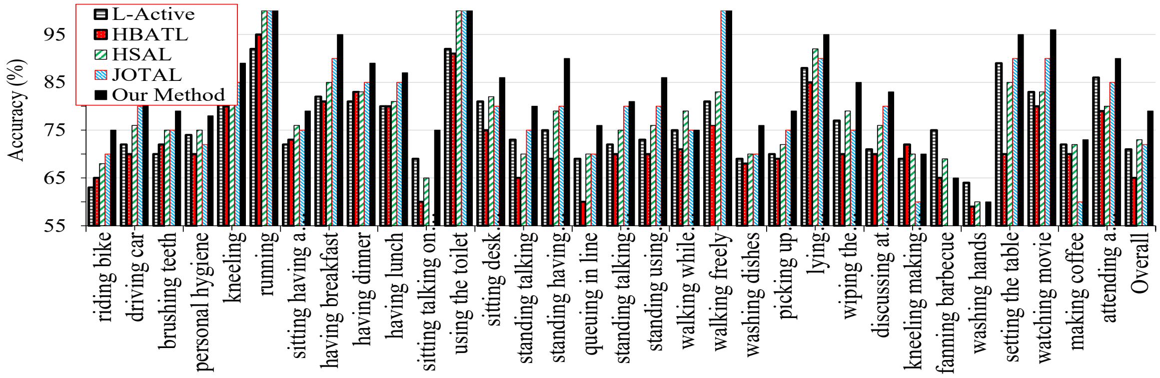

Fig. 8. TU Darmstadt Dataset: Unseen Activity Classification Accuracy with leave-two-out cross-validation method and comparison with baseline methods.

terms of the number of seen classes in the training data which also depicts that our framework outperforms other baseline frameworks in this case as well.

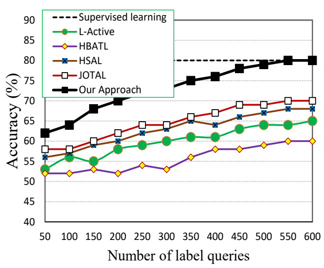

Fig. 9. TU Darmstadt Dataset AR accuracy comparisons with baseline through 600 label queries

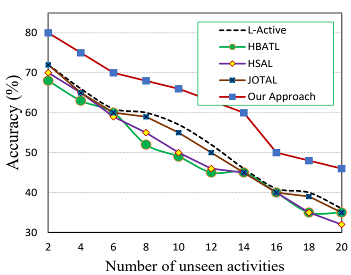

Fig. 10. TU Darmstadt Dataset AR accuracy comparisons through number of unseen activities

F. Importance of Hierarchical Clustering

Hierarchical clustering and efficient pruning are two important steps in our framework. We experiment on how this hierarchical clustering helps improve our AR performance using RCC Dataset. To measure our clustering performance, we apply cluster *purity* evaluation metric. To compute purity, each cluster is assigned to the class which is most frequent in the cluster, and then the accuracy of this assignment is measured by counting the number of correctly assigned classes and dividing by N. Formally,

$$purity(\Omega, \mathcal{C}) = \frac{1}{N} \sum_k \max_j |\omega_k \cap c_j| \tag{3}$$

where $\Omega = \{\omega_1, \omega_2, \ldots, \omega_K\}$ is the set of clusters and $\mathcal{C} = \{c_1, c_2, \ldots, c_J\}$ is the set of classes. Bad clusterings have purity values close to 0 while a perfect clustering has a purity of 1. Fig 12 shows our hierarchical clustering purity outperforms other clustering methods used in the baseline frameworks.

G. Importance of Semantic Attribute Detector

Semantic attribute based hierarchical clustering has been proven as an optimal way to get higher accuracy in the active learning based AR domain [13]. In this paper, we use semantic attribute detection method in our extended HATL framework. Fig 13 shows that inclusion of the semantic attribute detector increases the overall *unseen* AR accuracy (6% improvement) over the number of label queries requested by the system.

H. Importance of Contextual Informativeness

We introduce a contextual informativeness (CI) based *seen-unseen* AR technique to select active learner and transfer learner. CI checks for sufficient confidence measure of activity classification based on currently labeled data points (i) before transfer learning; (ii) after transfer learning, and (iii) finally after active learning which continues in a loop until desired confidence has been achieved. This life cycle of CI confirms that our framework provides maximum fine-grained granularity of activity labeling with minimum possible computational time (instance selection time). Fig 13 shows inclusion of CI improves the accuracy significantly (4% improvement). On the other hand, Fig 15 shows significant reduction of instance selection time which can be achieved if CI has been included in our framework. CI also solves the problem of similar activities but in different location problem (‘sweeping’ in living room or bedroom) in the transfer learning phase too.

I. Importance of Semantic Attribute Learning

Semantic attribute based hierarchical active transfer learning enables a new way of representing the data points that is understandable by experts even without watching the video. For example, consider a new activity data point ‘cooking’ (‘cooking’ is not part of our entire dataset) has been found in the test dataset which is neither seen nor poorly labeled in the source or target dataset. Fig 14 shows ‘cooking’ complex activity representation in terms of only low-level ($\langle postural, location \rangle$) and semantic attribute context format. It can be depicted that semantic attribute representation of complex activities can visualize complex activities more clearly than low-level activity context representation. However, defining and training samples with semantic attribute is an extra burden on the data labeling which is a costly task too. Our efficient extended hierarchical active transfer learning proves that the cost can be minimized significantly for both *seen* and *unseen* activities.

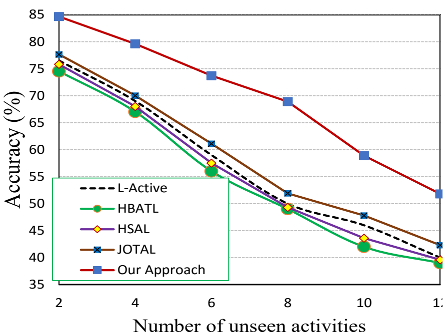

Fig. 11. RCC Dataset: Activity recognition accuracy comparisons through number of unseen activities

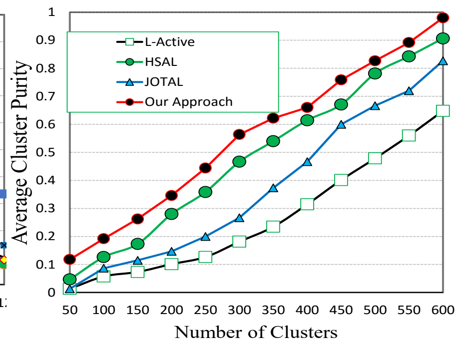

Fig. 12. RCC Dataset: Average cluster purity comparison as a function of increasing the number of clusters

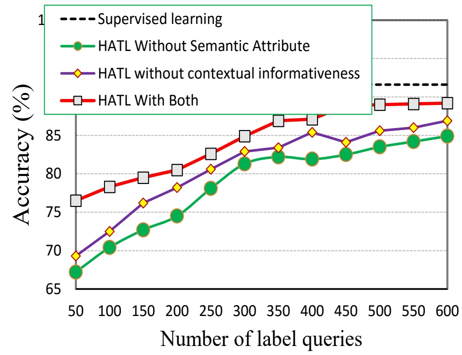

Fig. 13. RCC Dataset: Test set AR accuracy through 600 queries for our approach with and without inclusion of semantic attribute detector

As our semantic attribute detection performance is not that satisfactory, new algorithms can be explored to improve the performance.

J. Importance of Hierarchical Cluster Classification

Our hierarchical cluster based HSVM classification algorithm applies a pruning technique to reduce the number of samples required to be labeled in the HSAL algorithm (Algorithm 2 Line 13). The pruning costs incur extra computation complexity to our overall framework. However, as shown in Fig 16, this pruning can reduce the number of clusters significantly while running the core active transfer learning query label request.

Posture Location	walk kitchen	stand kitchen	walk living	stand living	walk kitchen	stand kitchen	stand kitchen	stand kitchen
Semantic Attribute	Washing dishes		Retrieve utensils		chopping		Blending spices	

Fig. 14. Representation of complex activity 'cooking' as $< postural, location >$ and $< semanticattribute >$ hierarchicaltuples

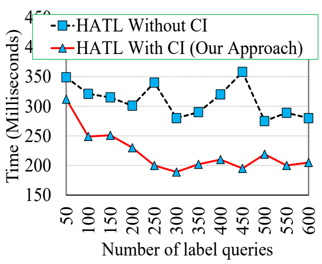

Fig. 15. RCC Dataset: Importance of Contextual Informativeness Testing in terms of instance time selection. Significant reduction of instance selection time achieved

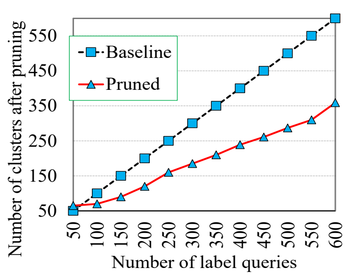

Fig. 16. RCC Dataset: Importance of cluster pruning testing. Significant reduction of number of clusters while label query requested by the system

VI. Discussion

In our framework, we start with some labeled dataset and gradually update both source and target dataset. If the number of *unseen* activities in the target dataset is much less than the total number of dissimilar activities between source and target dataset (source dataset has more unknown activities than target dataset), then there is a possibility of negative transfer learning which can not be solved in next stage of active learning. This scenario can be visible in case of large scale dataset. For example, in Fig. 10 shows accuracy drops drastically (below 50%) when unseen activities are more than half of total classes. This framework is also limited to a similar setup of the smart home. The semantic attribute detection is one of the most important parts in our system which did not provide satisfactory recognition accuracy. We also did not compare unsupervised method performances with our proposed framework in the current version.

VII. Conclusion

We have presented the design, implementation, and evaluation of a new AR system that helps reinforce the scalability, generalizability, adaptability, and the statistical importance of the unseen AR process exploiting its underlying taxonomical structure. While prior unseen AR techniques rely on accuracy costing computational complexity and label imputation errors, our hierarchical activity taxonomy learning guided active transfer learning framework judiciously mitigates the costing metrics with significant performance improvement. In addition, our efficient active learning assisted contextual informativeness aware learner selection method helps achieve the optimal performance gain with minimum costs on activity datasets comprising both *seen* and *unseen* activities. Moreover, we are the first of the kind extensively explored different active transfer learning methods in unseen activity recognition domain. Our proposed semantic attribute based generic representation of complex activities also opens up a research avenue bridging crowdsourcing with active transfer learning application domain.

Acknowledgment

The work is supported by the University of Maryland Baltimore-University of Maryland Baltimore County Research and Innovation Partnership grant.

REFERENCES

[1] Mohammad Arif Ul Alam, Nirmalya Roy, Archan Misra, Joseph Taylor, CACE: Exploiting Behavioral Interactions for Improved Activity Recognition in Multi-Inhabitant Smart Homes, in Proc. 36th IEEE International Conference on Distributed Computing Systems, ICDCS, 2016.

[2] Mohammad Arif Ul Alam, Nirmalya Roy, Sarah Holmes, Aryya Gangopadhyay, Elizabeth Galik, Automated Functional and Behavioral Health Assessment of Older Adults with Dementia, in Proc. First IEEE Conference on Connected Health: Applications, Systems and Engineering Technologies, CHASE, 2016.

[3] Mohammad Arif Ul Alam, Nilavra Pathak, and Nirmalya Roy. Mobeacon: An iBeacon-Assisted Smartphone-based Real Time Activity Recognition Framework, in Proc. of the 12th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, MobiQuitous, 2015.

[4] U.S. Bureau of Labor Statistics. American time use survey activity lexicon. American Time Use Survey, 2011.

[5] Jiahui Wen, Jadwiga Indulska, Mingyang Zhong: Adaptive activity learning with dynamically available context. PerCom 2016.

[6] Yangchi Chen, Melba M. Crawford, Joydeep Ghosh: Integrating support vector machines in a hierarchical output space decomposition framework. IGARSS 2004: 949-952.

[7] N. Roy, A. Misra, D. Cook, Infrastructure-assisted smartphone-based ADL recognition in multi-inhabitant smart environments. PerCom 2013.

[8] N. Lane, L. Pengyu, L. Zhou, F. Zhao, Connecting personal-scale sensing and networked community behavior to infer human activities. UbiComp 2014.

[9] L Peng, L Chen, X Wu, H Guo, G Chen, Hierarchical Complex Activity Representation and Recognition using Topic Model and Classifier Level Fusion, IEEE Transactions on Biomedical Engineering, vol PP, issue 99, Aug 31, 2016.

[10] C. Zhu, W. Sheng, Realtime Recognition of Complex Human Daily Activities Using Human Motion and Location Data. IEEE Trans. Biomed. Engineering 59(9), 2012.

[11] D. Cook and N. Krishnan. Activity Learning from Sensor Data. Wiley, 2015.

[12] Narayanan Chatapuram Krishnan, Diane J. Cook, Zachary Wemlinger: Learning a taxonomy of predefined and discovered activity patterns. JAISE 5(6), 2013.

[13] Heng-Tze Cheng, Feng-Tso Sun, Martin L. Griss, Paul Davis, Jianguo Li, Di You: NuActiv: recognizing unseen new activities using semantic attribute-based learning. MobiSys 2013.

[14] Le T. Nguyen, Ming Zeng, Patrick Tague, Joy Zhang: I did not smoke 100 cigarettes today!: avoiding false positives in real-world activity recognition. UbiComp 2015

[15] B. Minor, D. Cook, and J. Doppa. Data-driven activity prediction: Algorithms, evaluation methodology, and applications. ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2015.

[16] JK Aggarwal and Lu Xia. Human activity recognition from 3d data: A review. Elsevier Pattern Recognition Letters, 2014.

[17] Empatica E4 Wristband: https://www.empatica.com/e4-wristband

[18] Dany Fortin-Simard, Jean-Sbastien Bilodeau, Kevin Bouchard, Sbastien Gaboury, Bruno Bouchard, Abdenour Bouzouane: Exploiting Passive RFID Technology for Activity Recognition in Smart Homes. IEEE Intelligent Systems 30(4): 7-15, 2015.

[19] Bourobou STM, Yoo Y. User Activity Recognition in Smart Homes Using Pattern Clustering Applied to Temporal ANN Algorithm. Puliafito A, ed. Sensors (Basel, Switzerland). 2015.

[20] Mahmudul Hasan, Amit K. Roy-Chowdhury: Context Aware Active Learning of Activity Recognition Models. ICCV 2015.

[21] H. M. Sajjad Hossain, Nirmalya Roy, Md Abdullah Al Hafiz Khan: Active learning enabled activity recognition. PerCom 2016.

[22] S. L. Cao, S. Keerthi, C. J. Ong, P. Uvaraj, X. J. Fu, H. P. Lee, Developing parallel sequential minimal optimization for fast training support vector machine, Neurocomputing, 2006.

[23] Kyle Dillon Feuz , Diane J. Cook , Heterogeneous transfer learning for activity recognition using heuristic search techniques, International Journal of Pervasive Computing and Communications, 2014, Vol. 10 Iss: 4, pp.393 - 418.

[24] M. Stikic, D. Larlus, S. Ebert, and B. Schiele. Weakly supervised recognition of daily life activities with wearable sensors. IEEE Trans. Pattern Analysis and Machine Intelligence, 33(12):25212537, 2011.

[25] Diethe, Tom, Niall Twomey, and Peter Flach, Bayesian active transfer learning in smart homes. ICML Active Learning Workshop, 2015.

[26] A Wilson, A Fern, and P Tadepalli. Transfer learning in sequential decision problems: A hierarchical Bayesian approach. In ICML, pages 217–227, 2012.

[27] Derek Hao Hu, Vincent Wenchen Zheng, Qiang Yang: Cross-domain activity recognition via transfer learning. Pervasive and Mobile Computing 7(3): 344-358, 2011.

[28] David C. Kale, Marjan Ghazvininejad, Anil Ramakrishna, Jingrui He, Yan Liu: Hierarchical Active Transfer Learning. SDM 2015: 514-522.

[29] Nguyen, L. T., Zeng, M., Tague, P., and Zhang, J. Superad: Supervised activity discovery. In International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication, ACM, 2015.

[30] Stikic, M., Van Laerhoven, K., and Schiele, B. Exploring semi-supervised and active learning for activity recognition. In 12th IEEE International Symposium on Wearable Computers, 2008. ISWC 2008, IEEE (2008), 8188.

[31] Zheng, Li, and Tao Li, Semi-supervised hierarchical clustering. In 2011 IEEE 11th International Conference on Data Mining, pp. 982-991. IEEE, 2011.

[32] Le T. Nguyen, Ming Zeng, Patrick Tague, Joy Zhang: Recognizing new activities with limited training data. ISWC 2015: 67-74.

[33] A. Akl, C. Feng, S. Valaee, A Novel Accelerometer-Based Gesture Recognition System. IEEE Transactions on Signal Processing, 2011.

[34] PogoPlug: https://pogoplug.com/

[35] S. Dasgupta and D. Hsu. Hierarchical sampling for active learning. In Proceedings of the 25th International Conference on Machine Learning (ICML), pages 208–215, 2008.

[36] Gjoreski, Martin, et al. How accurately can your wrist device recognize daily activities and detect falls?. Sensors 16.6 (2016): 800.

[37] D.H. Fisher, Knowledge acquisition via incremental conceptual clustering. Machine Learning, 2(2):139-172, 1987.

[38] H. Larochelle, D. Erhan, and Y. Bengio. Zero-data learning of new tasks. In Proc. Conf. on Artificial intelligence, AAAI'08, pages 646–651, 2008.

[39] M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell. Zero-shot learning with semantic output codes. In Proceedings of the Neural Information Processing Systems (NIPS), pages 1410–1418, 2009.

[40] R. Chattopadhyay, W. Fan, I. Davidson, S. Panchanathan, and J. Ye. Joint transfer and batchmode active learning. In Proceedings of the 30th International Conference on Machine Learning (ICML), pages 253–261, 2013.

[41] H. He and E.A. Garcia, Learning from imbalanced data. IEEE Transactions on Knowledge and Data Engineering, 21:1263-1284, 2009.

[42] X.Y. Liu, J. Wu, and Z.-H. Zhou, Exploratory under-sampling for class-imbalance learning. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 39(2):535-550, 2009.