2011 11th IEEE International Conference on Data Mining

# Helix: Unsupervised Grammar Induction for Structured Activity Recognition

Huan-Kai Peng, Pang Wu, Jiang Zhu, and Joy Ying Zhang
*Carnegie Mellon University, Moffett Field, CA, USA*
{*huankai.peng, pang.wu, jiang.zhu, joy.zhang*}@*sv.cmu.edu*

***Abstract*—The omnipresence of mobile sensors has brought tremendous opportunities to ubiquitous computing systems. In many natural settings, however, their broader applications are hindered by three main challenges: rarity of labels, uncertainty of activity granularities, and the difficulty of multi-dimensional sensor fusion. In this paper, we propose building a grammar to address all these challenges using a language-based approach. The proposed algorithm, called Helix, first generates an initial vocabulary using unlabeled sensor readings, followed by iteratively combining statistically collocated sub-activities across sensor dimensions and grouping similar activities together to discover higher level activities. The experiments using a 20-minute ping-pong game demonstrate favorable results compared to a Hierarchical Hidden Markov Model (HHMM) baseline. Closer investigations to the learned grammar also shows that the learned grammar captures the natural structure of the underlying activities.**

***Keywords*-Ubiquitous Knowledge Discovery; Heterogeneous Sensor Fusion; Unsupervised Grammar Induction.**

## I. Introduction

The penetration of smartphones with various embedded sensors makes available logging human activities with sensory data in large scale. Such a data quantity brings opportunities for Ubiquitous Computing Systems (UCS) to mine human activities. Three types of UCS applications depend upon accurate and flexible activity recognition:

- Context-aware mobile computing utilizes users' current activity, e,g., whether the user is "driving", "meeting" or "jogging" can help determine if an incoming call should be answered.
- Behavior-ware mobile computing utilizes users' behavior pattern, e.g., turning on the home heater before the user returns home based on his/her commuting pattern, or allocating the wireless bandwidth based on the user's mobility pattern in an office building.
- Activity monitoring uses sensors embedded in the building/furniture or worn on-body that allows caregivers to monitor elderlies who choose to age in place. In particular, detection of abnormal activities (e.g., accidental falls) can be used for alerting the caregiver for rapid response.

Empirically, we observe two additional challenges in sensor-based activity recognition compared to standard sequence classification. They are **heterogeneous sensor fusion** and **unsupervised structural activity recognition**.

In standard sequence classification, input data are usually one-dimension sequences, such as words or DNA nucleotides. For UCS applications, activities are captured by various sensors such as accelerometers, gyroscopes, GPS receiver, WiFi receiver, microphone, and so on. These sensors are heterogeneous in data format (e.g., real number value or location coordinates), sampling rate, and semantic meaning. Past research has shown that utilizing multiple sensors can improve activity recognition.

Another practical challenge is the need for unsupervised structural activity recognition. Supervised methods use samples with predefined labels such as "walking" or "running". Practically, however, not only it is unrealistic to assume large amounts of labeled data, but also it's limited to recognize only the predefined activities. In many applications, it is more useful to recognize activities at high-level, or more generally, in a structured manner. Such an approach will allow a UCS to identify high-level activities like "playing a tennis game", lower-level ones like "a play", or even breaking down to "step" or "swing", etc.

In this work, we propose a novel approach called *Helix* to induce the underlying grammar of human activities. Inspired by grammar induction in natural language processing, Helix is a greedy method that iteratively does two things. First, it identifies high-level activities through bracketing two statistically collocated sub-activities. For example, if "swing backward" and "swing forward" occur together much more than random, we merge them into "swing backward then forward". Second, it generalizes formally or semantically similar activities into yet higher-level activities. For example, if "swing backward then forward" and "swing backward then tip" looks similar and both occur mostly between "moving back" and "hit the ball", then we generalize them into a higher-level activity "returning the ball". We evaluate Helix in our Lifelogger system, compare its performance to Hierarchical Hidden Markov Model (HHMM), and find favorable results in both recognition accuracy and computational cost.

## II. Unsupervised Activity Structure Learning using a Language-based Approach

With illustrations in Figure 1, we first define the problem as follows: given $d$-dimensional sensor readings $S = \{S_1, \ldots, S_{|d|}\}$ with some underlying activities, we define the multi-level *Activity Text* as $A = \{A^1, \ldots, A^L\}$ of $L$ levels, where $A^l = \{a^l_1, \ldots, a^l_{|A^l|}\}$ in which each $a^l_i$ represents


DOI 10.1109/ICDM.2011.74

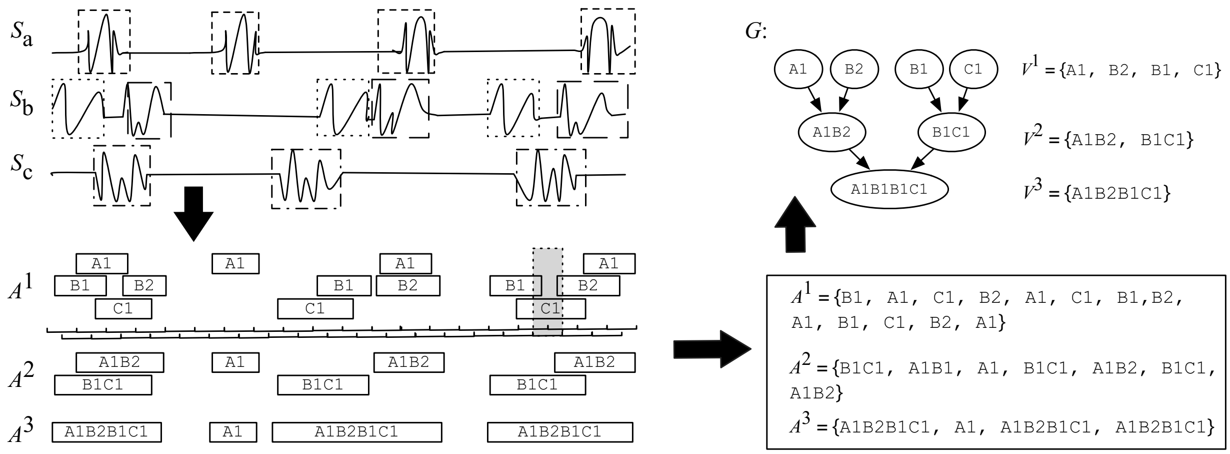



Figure 1. Grammar induction using Helix. $A$, $V$, and $G$ denote the multi-level activities, vocabulary, and induced grammar, respectively.

the $i$-th activity in level $l$. Here $A^1$ denotes lowest-level activities (e.g., "twisting the waist") while $A^L$ denotes the highest (e.g., "playing a game"). Subject to $A$, we define an *Activity Grammar* $G = \{V, R\}$, where $V$ denotes a multi-level *Activity Vocabulary* of $L$ levels and $R$ denotes the *Relations* specifying the structure of the grammar. Specifically, $V = \{V^1, \ldots, V^L\}$, where $V^l$ represents the activity vocabulary of the $l$-th semantic level, such that each $l$-th level activity $a_i^l \in \{V^1 \cup \cdots \cup V^l\}$. Accordingly, any directed relationship $(v_i^{l_1} \rightarrow v_j^{l_2}) \in R$ iff $l_2 > l_1$ and $v_j^{l_2}$ is a *super-activity* or a *generalized activity* of $v_i^{l_1}$.

*Definition.* **Unsupervised Activity Structure Learning**. Subject to a set of unlabeled multi-dimensional sensor readings $S$ generated by some activities, the *Unsupervised Activity Structure Learning Problem* is defined as finding a grammar $G = \{V, R\}$ that best describes the hierarchical activity structure embedded in $S$.

We then describe how to find such a grammar $G$ using an unsupervised algorithm consisting of three main steps: (1) vocabulary initialization, (2) collocation discovery, and (3) vocabulary generalization.

### A. Vocabulary Initialization using Time-series Motifs

An established approach to build a vocabulary from time-series is by finding *time-series motifs* [8], the recurring patterns of similar subsequences. As in the top-left of Figure 1, we extend [8] to extract similar subsequences. They are then assigned labels to form the initial vocabulary. In the example, $V^1 = \{\texttt{A1}, \texttt{B1}, \texttt{B2}, \texttt{C1}\}$ and $A^1 = \{\texttt{B1}, \texttt{A1}, \texttt{C1}, \texttt{B2}, \texttt{C1}, \texttt{B1}, \texttt{B2}, \texttt{A1}, \texttt{B1}, \texttt{C1}, \texttt{B2}, \texttt{A1}\}$.

### B. Super-Activity Discovery by Statistical Collocation

In Statistical Natural Language Processing (SNLP), *Collocation* [7] is the "expression of two or more words that correspond to some conventional way of saying things". In the context of super-activity discovery, collocation is the combination of two activities where their joint-occurrence frequency is significantly larger than what could be resulted randomly from their marginal frequencies.

To discover super-activities using statistical collocation, we first assume that the activity pairs occurring jointly within a window may be components of one super-activity, like an intertwined double helix. For example, the grey dotted box in Figure 1 includes 3 activity pairs ([B1,B2], [B1,C1], and [B2,C1]). Note that the order within the pair matters: two sub-activities from the same dimension are ordered by time ([B1,B2]), whereas two sub-activities from different dimensions are ordered by their dimension rank ([B2,C1]). The rationale is that for a super-activity, the ordering between its sub-activities in the same dimension usually matters (e.g., leaning forward then stand up), but not so much for those from different dimensions (doing an exam while feeling pressure).

By sliding the grey window along the current activity text ($A^1$), we accumulate the joint frequencies of all activity pairs and record them in a *Joint Frequency Table* (upper-left of Figure 2). The *Marginal Frequency Table* (upper-right of Figure 2) is also accumulated similarly. For example, the pair [B1,C1] will account for one [B1,-] and one [-,C1]. Although the content of the two tables depends on the length and the step size of the sliding window, we found that the collocation discovery results are not sensitive to them, because as the size of a super-activity grows, the region within which other activities can co-occur with it also stretches. Empirically, setting the sliding window's step size to an atomic activity length (e.g., 0.5s) and the window size to a multiple of it (e.g., 5s) gives good results.

With the joint and marginal frequency tables, we test each activity pair for its statistical collocation significance. We do so by constructing the contingency table for each activity pair and calculating the $\chi^2$ (chi-square) statistics [7]:

$$\chi^2 = \sum_{i,j} \frac{(O_{ij} - E_{ij})^2}{E_{ij}} \tag{1}$$

**Joint Freq. Table**

| w1 | w2 | Freq. |
|---|---|---|
| A1 | B1 | 5 |
| A1 | B2 | 12 |
| A1 | C1 | 6 |
| B1 | B2 | 3 |
| B1 | C1 | 10 |
| B2 | B2 | 6 |

**Marginal Freq. Table**

| w1 | w2 | Freq. |
|---|---|---|
| A1 | – | 23 |
| B1 | – | 13 |
| B2 | – | 6 |
| – | B1 | 5 |
| – | B2 | 15 |
| – | C1 | 22 |

**Contingency Table of** [A1,B2]

| | A1 | ¬ A1 | |
|---|---|---|---|
| B2 | 12 | 11 | 23 |
| ¬ B2 | 3 | 16 | 19 |
| | 15 | 27 | 42 |

$$\chi^2 = \frac{42 \times (12 \times 16 - 11 \times 3)}{23 \times 19 \times 15 \times 27} = 5.99$$

$$\geq \alpha_{0.95} = 0.3841$$

Figure 2. Joint and marginal frequencies of activity pairs, with the contingency table of the pair (A1,B2).

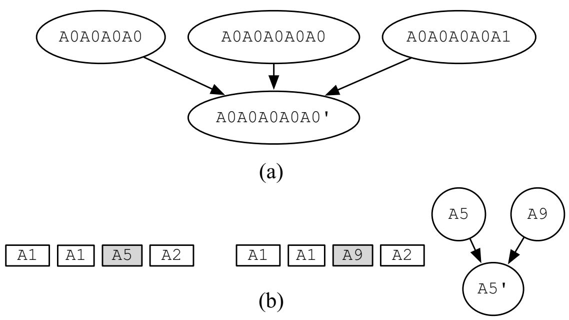



Figure 3. (a) Content-based and (b) context-based generalizations.

where $O_{ij}$ denotes the $(i, j)$-th entry of the contingency table and $E_{ij}$ denotes the expectation of the entry derived from their marginals. For [A1,B2] in the Figure 2, $\chi^2 = 5.99 \geq \chi^2_{0.05}(1) = 3.841$. Therefore, we can merge [A1,B2] to form a super-activity with 95% confidence level, here we use $\alpha_{TH} = \chi^2_{0.05}(1)$ as the threshold parameter controlling the merging strength. Repeating this step for three iterations, we obtain the grammar in the right-bottom of Figure 1.

### C. Vocabulary Generalization

The collocation discovery works when there are repeating sequences of sub-activities. For real human activities, however, repetitions are unlikely to be exact. For example, when one is playing ping-pong, he may perform a cut, a lift, or a regular play, which can have various sequence combinations. In order to detect higher-level activities in such a condition, we need to *generalize* similar activities before constructing the joint and marginal frequency tables. Such a clustering is done according to two similarity measures: (1) Content Similarity ($\phi_e$) and (2) Context Similarity ($\phi_x$).

*1) Content similarity ($\phi_e$):* As in Figure 3(a), the first generalization accounts for activities similar in content, including two main cases. The first case is when multiple activities are both repetitions of the same sub-activity, e.g., A0A0A0A0, and A0A0A0A0A0 (left two nodes at the top). The second case is when two activities differ in only minor portion in their composition, e.g., A0A0A0A0A0, and A0A0A0A0A1 (right two nodes).

To consider both of these cases of two similar phrases (activities) $v_1$ and $v_2$, we first calculate their $n$-gram precision and recall. W.l.o.g., we assume $|v_1| \leq |v_2|$ and $n = 2$. For the example in Figure 3(a), $v_1$ = A0A0A0A0 and $v_2$ = A0A0A0A0A1, such that $nGram(A)$ consists of 3 A0A0's and whereas $nGram(B)$ consists of 3 A0A0's and 1 A0A1. Accordingly, we have:

$$\begin{aligned} Precision &= \frac{|nGram(v_1) \cap nGram(v_2)|}{|nGram(v_1)|} = 1 \\ Recall &= \frac{|nGram(v_1) \cap nGram(v_2)|}{|nGram(v_2)|} = 0.75 \\ \phi_e &= 2 \times \frac{Precision \times \sqrt{Recall}}{Precision + \sqrt{Recall}} = 0.93 \end{aligned} \quad (2)$$

The square root of the recall term is to award that case when two phrases are very similar in content but differs in length.

*2) Context similarity ($\phi_x$):* The second generalization accounts for activities occurring within similar context. For example, A5, and A9 in Figure 3(b) are both preceded by 2 A1's and followed by A2, suggesting that these two activities are likely to be semantically similar. We represent the context of an activity as $\vec{c} = (n_1, \ldots, n_{|V|})$, representing the number of times every activity co-occurs with the activity. Then the context similarity $\phi_x$ between two activities is the cosine distance between their context vectors:

$$\phi_x = \frac{\vec{c_1} \cdot \vec{c_2}}{|\vec{c_1}||\vec{c_2}|} \quad (3)$$

Before clustering, we need to aggregate the content and context similarities into one single similarity measure $\phi$. Intuitively, the arithmetic, geometric, and harmonic means are all reasonable candidates. However, in order to make each of $\phi_e$ and $\phi_x$ sufficient for generalization (instead of requiring both), we use the arithmetic mean. This aggregate $\phi$ is then used in a complete-link clustering algorithm where a link exists between two activities only if their similarity $\phi$ is larger than a threshold $\phi_{TH}$. Consequently, all activities grouped together in a cluster are all highly similar to each other. Algorithm 1 summarizes the overall Helix algorithm.

## III. EXPERIMENTS

We implemented Helix as the activity structure discovery engine of our Lifelogger system [11] as in Figure 4. The grammar learning for a 20-minute pingpong session takes less than 1 minute where the peak memory consumption is less than 10 MB. For the dataset, we collect all availible sensor readings from a smartphone plus a Electroencephalography (EEG) to record the electrical activity of the participant's brain. During the experiment, the participant first does three kinds of serves (the cut, the lift, and the regular) for 5 times each. Then he does 3 practice rounds with an opponent, followed by two matches for 11 and 6 points, respectively. These activities are annotated with three semantic levels of ground truth as depicted in Table I. The first level ground truth $T_1$ is based on a play, e.g., Cut, Lift, and Regular. The second level ground truth $T_2$ is based on

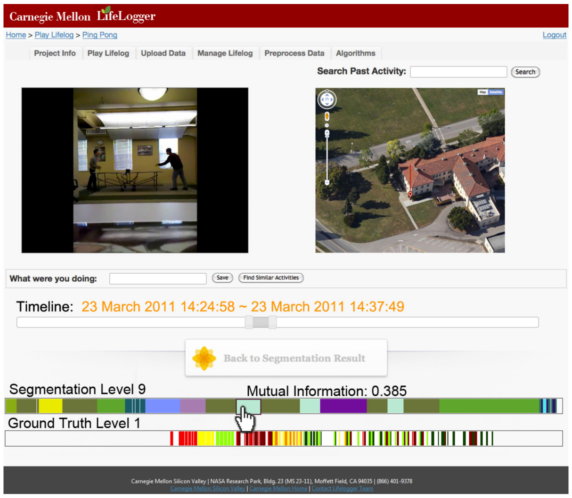



Figure 4. Helix implementation in CMU's Lifelogger system.

individual tests (5 to 10 plays), e.g., `CutServeTest` and `CutReplyTest`. The third level $T_3$ is based on groups of tests, e.g., `ServeTest`, `ReplyTest`, and `GameTest`. We take 9-dimensional raw sensor reading data including 3-axis accelerometer (X-, Y-, and Z-axises represented as Sensor `A` $\sim$ `C`) and EEG (high-pass and low-pass signals for Alpha, Beta, and Gamma waves, represented as Sensor `D` $\sim$ `I`).

As a baseline, we implemented an Hierarchical HMM (HHMM) with the Matlab Bayes Net Toolbox [9]. To have a reasonably good baseline, we set the HHMM structure according to ground truth characteristics: the HHMM height is set to 3 and the number of states at level 1, 2 and 3 are set to 3, 5, and 10, respectively. We also tried to set the HHMM to have a large height, but that makes the model too complicated and consumes >16GB memory for a height of 6. We therefore stay with the setting based on ground truth.

**Algorithm 1** Hierarchical Activity Structure Discovery

**Input:** $S = \{S_1, \ldots, S_d\}$ : $d$-dimensional sensor readings
**Input:** $\alpha_{TH}$ : merging threshold parameter
**Input:** $\delta_{TH}$ : generalization threshold parameter
**Output:** $G = \{V, R\}$ : discovered hierarchical grammar
**Output:** $A$ : hierarchical activities labeled using $V$

```
(A^1, V^1) = initialize the vocabulary by motif discovery
(A', V') = (A^1, V^1)
l = 1
while TRUE do
    l = l + 1
    (A^l, V^l) = discover collocations from (A', V')
    break if |V^l| == 0
    for all v_i ∈ V^l do
        add edges (v_i, v_j) into R for all collocations
    end for
    (A', V') = generalize the vocabulary from (A^l, V^l)
end while
V = {V^1, ..., V^{l-1}}
return G = {V, R}, A
```

Table I
THE *pingpong* DATASET'S GROUND-TRUTH LABELS

| Semantic Lv. | Label | Qty. |
|---|---|---|
| Level 1 ($T_1$) | `Cut` | 14 |
| | `Lift` | 14 |
| | `Regular` | 23 |
| | `CutReply` | 7 |
| | `LiftReply` | 18 |
| | `RegularReply` | 30 |
| Level 2 ($T_2$) | `CutServeTest` | 2 |
| | `LiftServeTest` | 2 |
| | `RegularServeTest` | 2 |
| | `CutReplyTest` | 2 |
| | `LiftReplyTest` | 2 |
| | `RegularReplyTest` | 2 |
| | `GameTest` | 2 |
| Level 3 ($T_3$) | `ServeTest` | 1 |
| | `ReplyTest` | 1 |
| | `MatchTest` | 1 |

### A. *Evaluation of the Learned Activity Grammar*

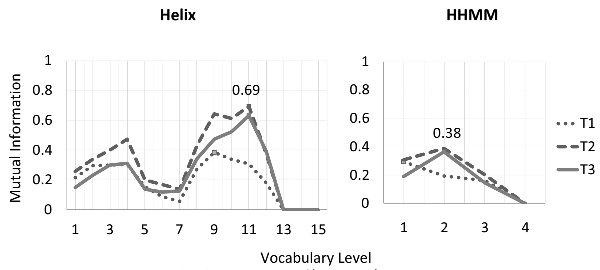
We evaluate the learned grammar by the mutual information of its vocabulary with the ground truth. We first split the whole pingpong dataset into units of 1ms. In each unit $i$, there is a ground-truth label $g_i$ using one level of ground-truth labels $T^k$, with a corresponding learned label $p_i$ using a level of learned vocabulary $V^l$. We then measure the similarity of the two labelings by:

$$\mathrm{MI}(T^k, V^l) = \mathrm{H}_p(T^k) + \mathrm{H}_p(V^l) - \mathrm{H}_p(T^k, V^l) \quad (4)$$

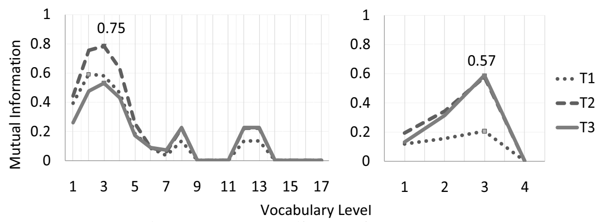
where $\mathrm{MI}(T^k, V^l)$ denotes the mutual information between the two labelings, $\mathrm{H}_p(T^k)$ and $\mathrm{H}_p(V^l)$ denote the entropy of the ground-truth and the learned labeling, respectively, and $\mathrm{H}_p(T^k, V^l)$ denotes the joint entropy of the two labelings.

Figure 5 compares the results between Helix and HHMM with three levels of ground truth using different sensor dimension combinations. The $y$-axis denotes mutual information; the $x$-axis denotes vocabulary level. Several peaks are marked with their y-axis value, indicating the vocabulary levels that best describe the underlying activity.

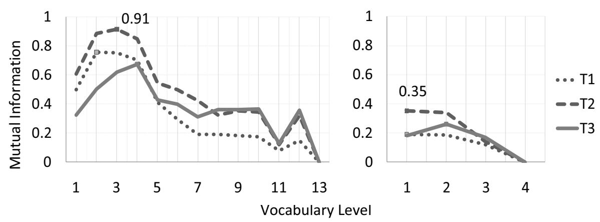
Accordingly, three observations are made. (1) Using all sensor combinations, Helix's results generally have two peak regions (the first at vocabulary level 2$\sim$5 and the second at level 11$\sim$17). Since such peak regions occurs in similar vocabulary levels across all ground-truth levels, this suggests that the activity structure underlying the pingpong dataset reflects these two-levels by nature, which are naturally correlated to human-annotated ground-truth at all semantic levels. In contrast, HHMM's results have only 1 peaking region. This reveals the first advantage of Helix over HHMM: by offering more levels of vocabulary (with $< 10MB$ memory), it is more flexible in capturing the underlying activities in multiple granularities. This advantage is confirmed by the constantly better global peak mutual information of Helix (0.69, 0.78, 0.91, and 0.91) over that of the classic



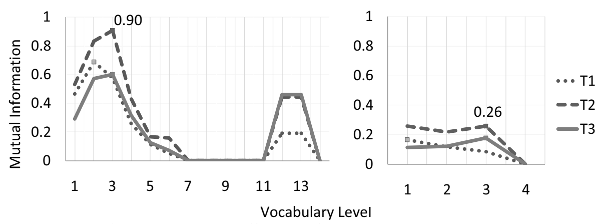

(a) 1 sensor dimensions: `A`




(b) 3 sensors dimensions: `ABC`




(c) 6 sensors dimensions: `DEFGHI`




(d) 9 sensors dimensions: `ABCDEFGHI`

Figure 5. Mutual information result comparison between Helix and HHMM with three levels of ground truth using different sensor dimension combinations: (a) accelerometer x-axis (`A`), (b) accelerometer's 3 axises (`ABC`), (c) 6 EEG dimensions (`DEFGHI`), and (d) all accelerometer and EEG sensors (`ABCDEFGHI`).

HHMM (0.38, 0.58, 0.34, 0.25) using various sensor combinations, respectively. (2) A closer look at the two local peak regions shows that the peaks of lower-level ground-truth often occur at slightly-lower vocabulary levels. For different sensor combinations from Figure 5(a) to Figure 5(d), the peaks for $T_1$ occur at vocabulary levels 9, 2, 2, and 2; the peaks for $T_2$ occur at vocabulary levels 11, 3, 3, and 3; the peaks for $T_3$ occur at vocabulary levels 11, 3, 4, and 3. This further confirms that the learned grammar properly reflects the human-annotated ground truth. (3) For all sensor combinations, the results of $T_2$ and $T_3$ are very similar in higher vocabulary levels. This can be explained by the fact that $T_2$ and $T_3$ reflect very similar semantic levels themselves. Plus, looking at Table I, each label of $T_3$ has only 1 instance. Such a non-repeating pattern is difficult to discover, and thus its best corresponding grammar

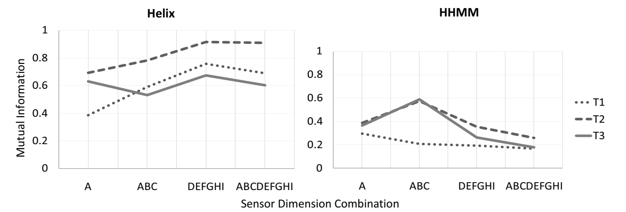



Figure 6. Peak mutual information result for Helix and HHMM with three levels of ground truth using different sensor dimension combinations.

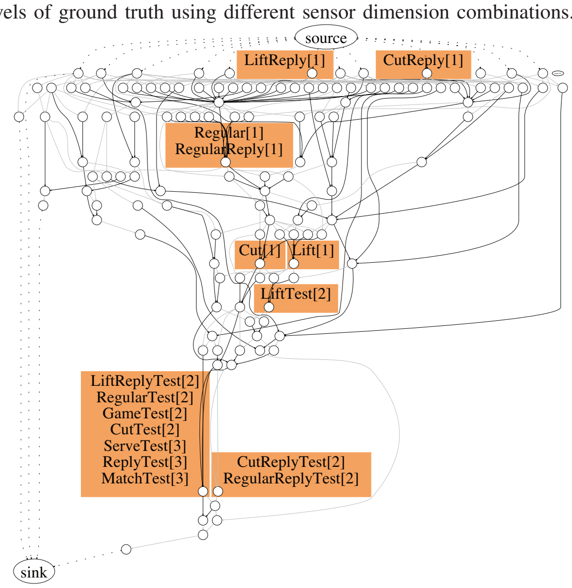



Figure 7. The learned grammar using Helix.

level could actually match the one of $T_2$, given $T_2$ and $T_3$ represent the same underlying activities.

To see how the effects associated with multiple sensors, we further plot the global peak mutual information values separately in Figure 6. As more (1, 3, 6, and 9) sensors are incorporated, the performance of HHMM degrades sharply, due to the known difficulty of learning patterns from heterogeneous sensors. In contrast, the performance of Helix shows little degradation. This shows the second advantage of Helix: it combines only statistically relevant sensors to form higher-level activities, which can adapt to each activity *dynamically*. Therefore, as additional sensors are provided, Helix can harvest the additional information while being less confused by the associated noise.

### B. Microscope Analysis of Grammar Structure

We visualize the learned activity grammar in Figure 7. To keep a reasonable number of nodes for visual clarity, we only use sensor `A` with the parameters of peak mutual information. The source and the sink are virtual nodes; the node level immediately below the source represents the initial vocabulary; the one immediately above the sink node represents the highest level activity. We annotate all ground-truth labels to the respective nodes having the highest mutual information, where the labels' semantic levels are appended in the brackets. Four observations can be made: (1) from top to bottom, the vocabulary size, i.e., the number of nodes at each level, grows and shrinks iteratively. This corresponds to the fact that merging (gray edges) usually increases the vocabulary size whereas generalization decreases it. This maintains a reasonable vocabulary size in the learning process, and explains why Helix can yield more granularity levels with less memory consumption ($<$10MB for 15 levels) compared to HHMM (16GB for 6 levels). (2) There are cases that several labels share the same node of the highest mutual information. Since the learning is completely data-driven, this suggests that from the data's perspective, theses labels may reflect to very similar activities. (3) All $T_3$ labels matches to the same node that is also shared with some $T_2$ labels. This confirms our previous conjecture that the non-repeating patterns in $T_3$ correspond to the same higher-level vocabularies as that of $T_2$ in the learned grammar. (4) Even though there is no one-to-one mapping between the learned grammar and human-annotated labels, the labels with higher semantic levels still correspond to higher-level nodes closer to the figure's bottom, and vice versa. This shows the learned grammar and human-annotated labels capture the same underlying activity structure.

## IV. Related Work

There are only few unsupervised activity recognition works compared to supervised ones. In [10], the authors combine Hidden Markov Model (HMM) and Gaussian Mixture Model (GMM) to cluster accelerometer data into groups.Also, [3], [2] use K-means to discover low-level activity clusters while applying topic modeling to discover high-level activities.For sensor selection and fusion, [6], [5], [4] compare the predictive performances of various sensors.Also, [1], [12] applies static fusion to multiple sensors.

## V. Conclusion

In this paper, we propose an unsupervised algorithm, Helix, for building an activity grammar to address the three key challenges for ubiquitous computing systems in natural settings, namely, rare labels, unknown activity granularities, and varying sets of activity-defining sensors. The experiments using a 20-minute ping-pong dataset demonstrate favorable results compared to a Hierarchical Hidden Markov Model (HHMM) baseline. Helix's advantage is three-fold: (1) It is an unsupervised algorithm, which still reserves the flexibility to utilize partial or later-available labels. (2) It is built from bottom up, considering all activity granularities. (3) It fuses heterogenous sensors dynamically, and is more resilient to the noise introduced by multi-dimensional sensors. Accordingly, this work contributes to the ubiquitous computing community by being the first work to our knowledge that considers all the three key challenges to understand human activities in natural settings. In the future, we plan to: (1) improve the grammar quality, (2) maintain an online grammar, and (3) do multi-level recognition.


## VI. Acknowledgement

This work was supported by CyLab at Carnegie Mellon under grants DAAD19-02-1-0389, W911NF-09-1-0273 from the Army Research Office, Google research award on "Social Behavior Sensing and Reality Mining" and the Cisco research award on "Behavior Modeling for Human Network". We would also like to thank Ross Grieshaber for proofreading the manuscript.


## References


[1] U. Blanke, B. Schiele, M. Kreil, P. Lukowicz, B. Sick, and T. Gruber. All for one or one for all? Combining Heterogeneous Features for Activity Spotting. In *International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*. Ieee, Mar. 2010.

[2] K. Farrahi and D. Gatica-perez. Discovering Routines from Large-Scale Human Locations using Probabilistic Topic Models. *ACM Transactions on Intelligent Systems*, 2010.

[3] T. Huynh, M. Fritz, and B. Schiele. Discovery of activity patterns using topic models. *International conference on Ubiquitous computing - UbiComp '08*, page 10, 2008.

[4] T. Huynh and B. Schiele. Analyzing features for activity recognition. *Joint conference on Smart objects and ambient intelligence innovative context-aware services: usages and technologies (sOc-EUSAI)*, 2005.

[5] J. Lester, T. Choudhury, N. Kern, G. Borriello, and B. Hannaford. A Hybrid Discriminative / Generative Approach for Modeling Human Activities. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2005.

[6] B. Logan, J. Healey, M. Philipose, E. Tapia, and S. Intille. A long-term evaluation of sensing modalities for activity recognition. In *International conference on Ubiquitous computing*, pages 483–500, 2007.

[7] C. D. Manning and H. Schuetze. *Foundations of Statistical Natural Language Processing*. The MIT Press, June 1999.

[8] A. Mueen, E. Keogh, Q. Zhu, and S. Cash. Exact discovery of time series motifs. *Proceedings of the SIAM*, 2009.

[9] K. Murphy and M. Paskin. Linear Time Inference in Hierarchical HMMs. In *Advances in neural information processing systems*, 2002.

[10] A. Nguyen, D. Moore, and I. McCowan. Unsupervised clustering of free-living human activities using ambulatory accelerometry. In *International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS)*, Lyon, France, 2007.

[11] P. Wu, H.-K. Peng, J. Zhu, and Y. Zhang. SensCare: Semi-Automatic Activity Summarization System for Elderly Care. In *International Conference on Mobile Computing, Applications, and Services (MobiCASE)*, 2011.

[12] A. Zinnen, U. Blanke, and B. Schiele. An Analysis of Sensor-Oriented vs. Model-Based Activity Recognition. *International Symposium on Wearable Computers*, Sept. 2009.