NECTAR: Knowledge-based Collaborative Active Learning for Activity Recognition

Gabriele Civitarese  
Univ. of Milano  
Italy

Claudio Bettini  
Univ. of Milano  
Italy

Timo Sztyler  
Univ. of Mannheim  
Germany

Daniele Riboni  
Univ. of Cagliari  
Italy

Heiner Stuckenschmidt  
Univ. of Mannheim  
Germany
Scenario

Recognizing activities of daily living in a smart-home to support healthcare, home automation, a more independent life, ...

We rely on unobtrusive sensors ...
State of the Art and Open Issues

Most activity recognition systems rely on …

… supervised-based approaches:
- acquire expensive labeled data sets
- often user/environment-specific

… knowledge-based approaches:
- require a significant effort in knowledge engineering
- not flexible
- questionable if such models could cover different environments and modes of execution
Our solution: NECTAR

*kKnowledge-basEd Collaborative acTive learning for Activity Recognition*

It overcomes drawbacks of supervised-based approach:
- not user/environment-specific, no expensive data set, …

It relies on semantic correlations:
- probabilistic dependencies (activities↔ events)
- derived from a possibly incomplete ontology

It exploits collaborative active learning:
- ...to refine rough correlations inferred by the ontology
MODEL AND SYSTEM
NECTAR’s architecture

1. Probabilistic and Ontological Activity Recognition
2. Query decision (entropy-based)
3. Collaborative Feedback Aggregation

Continuous stream of Sensor Events
1. Probabilistic/ontological activity recognition

We rely on ontological reasoning to pre-compute in an offline phase **semantic correlations**

they define probabilistic dependencies between home infrastructure and sensor events

A MLN combines those semantic correlations and sensor events to infer the most likely executed activities
Semantic Correlation Reasoner

Ontology / Axioms

OWL2 Reasoner infers

\{\text{turn on stove}\} \text{ is a predictive sensor event type for } \{\text{Prepare hot meal}\} \text{ and } \{\text{Prepare tea}\}

\begin{tabular}{|l|c|c|c|}
\hline
\textbf{SC Matrix} & \textbf{stove} & \textbf{silverware\_drawer} & \textbf{freezer} \\
\hline
\text{Hot meal} & 0.5 & 0.33 & 0.5 \\
\hline
\text{Cold meal} & 0.0 & 0.33 & 0.5 \\
\hline
\text{Tea} & 0.5 & 0.33 & 0.0 \\
\hline
\end{tabular}
Issues of this approach

Semantic correlations are computed based on an ontology written by knowledge engineers (humans)

- it is very likely that the ontology is incomplete
- it is hence questionable if it can cover different environments/mode of execution

Our goal is to refine and improve semantic correlations thanks to collaborative active learning!
2. Query decision

- Continuous Stream of Sensor Events
  - Online rule-based segmentation
    - Sensor events
  - Segment
    - Query decision (entropy-based)
      - Query
      - Feedback
      - Labeled segment
      - Semantic correlations
      - Collaborative Feedback Aggregation
        - Personalized update
        - ...
Online rule-based segmentation

We continuously segment the stream of sensor events based on knowledge-base conditions (e.g., interaction with objects, time gaps, changes of room).

Those conditions aim to generate segments which cover at most one activity instance.
Query decision

For each segment we derive a probability distribution over activities by **mining semantic correlations**

segments with high entropy values are queried to the inhabitant

\[
H(S) = \sum_{ac \in A} P(X = ac \mid S) \cdot \log \left( \frac{1}{P(X = ac \mid S)} \right)
\]

When \( H(S) \) is over a certain threshold we ask to the inhabitant the actual label of the segment \( S \)
3. Collaborative Feedback Aggregation

Labeled segments are transmitted to a cloud service by the participating homes.

- It stores feedback items: correspondence between sensor event types and activities.

Periodically, a personalized update is transmitted to each home.

- It contains reliable feedback items provided by similar environments.
Personalized update

To include only reliable feedback items in an update, we consider only whose \textit{support} is larger than a threshold \textit{support} is a value which indicates how many times the feedback was provided from different similar homes

We associate to each feedback item in an update:

- its \textbf{predictiveness}: computed as the normalization of support values
- its \textbf{estimated similarity}: the median value of similarity between origin/target environments
Semantic Correlation Updater

Each home receives periodically a set of **personalized feedback items**

- *predictiveness* is used to provide a semantic correlation to those event types for which the original ontology did not provide a starting correlation

- *estimated similarity* is used to scale semantic correlations of an event type which were originally computed by the ontology
EXPERIMENTS
Data Set

We consider a well-known data set …

CASAS

- Interleaved ADLs of twenty-one subjects
- Sensors: movement, water, interaction, door, phone
- Activities: fill medications dispenser, watch DVD, water plants, answer the phone, clean, choose outfit, …

We apply *leave-one-subject-out* cross validation:

- in each fold we collect feedback from 20 subjects to update semantic correlations for the remaining one
Recognition results (F1 score)

- Supervised approach (Random Forest)
- Image-based activity mining (unsupervised)
- NECTAR (without Active Learning)
- NECTAR (with Active Learning)
Improvement of collaborative active learning
Entropy threshold VS number of queries

![Bar chart showing the relationship between entropy threshold and average number of queries (0 to 10). The x-axis represents entropy threshold (0.6 to 1.2), and the y-axis represents the average number of queries.]
DISCUSSION / FUTURE WORK
Discussion

Results with a well-known dataset were positive, but...

- … contextual aspects should be taken in account to evaluate whether to ask a feedback
  - e.g., number of queries already been asked, current mood, availability

- … user interfaces need to be designed
  - e.g., vocal interfaces

- … knowledge engineering is still required (build starting ontology)
  - existing smart-home ontologies can be reused
Future Work

Data outsourced to the cloud service is sensitive …

… we will investigate solutions based on homomorphic encryption or secure multi-party computation

We also aim to extend our system …

… learning semantic correlations also for *temporal patterns*
THANKS FOR YOUR ATTENTION!
Entropy threshold VS F1 score
MLN / MAP Inference

Observed predicates

- Sensor Event: Freezer
  - 0.5: hot meal
  - 0.5: cold meal
  - 0.0: tea

- Sensor Event: Stove
  - 0.5: hot meal
  - 0.0: cold meal
  - 0.5: tea

ADL

- Event 1: opens freezer (1:00pm)
- Event 2: turns on stove (1:02pm)

- hot meal?
- cold meal?
- tea?

Hidden predicates

- Sensor Event: Freezer & Sensor Event: Stove
  - belong to ADL

- Hot meal
MLN Model (detailed)

PPM Matrix
  *SemanticCorrelation

Statistical analysis of events
  *InstanceCandidate / *Event

Ontological constraints
  time-aware inference
  temporal
  knowledge-based

Observed predicates
  *SemanticCorrelation
    (SenEvent, ADL, ActivClass, p)

Hidden predicates
  OccursIn
    (SenEvent, ADL)
  InstanceClass
    (ActivClass, ADL)

*Event
  (SenEvent, EventType, Time)

*InstanceCandidate
  (ADL, Start, Stop)
Semantic Integration Layer

- collects events data from a sensor network
- applies preprocessing rules to detect operations

Example

fridge door sensor signaled “1”

→

the operation is “opening the fridge”

<Event(se₁, et₁, t₁), …, Event(seₖ, etₖ, tₖ)>
**Statistical Analysis of Events**

**Input:** PPM matrix and temporally ordered events

- Infers most probable activity class for each event
- Allows to define activity boundaries (activity instance candidate)

Temporal extension of MLN ($MLN_{NC}$) Knowledge Base