Modeling and Reasoning with ProbLog: An Application in Recognizing Complex Activities

Timo Sztyler, Gabriele Civitarese, Heiner Stuckenschmidt

CoMoRea´18

19.03.2018

Content

1. Motivation
2. Model & System
3. Preliminary Results
4. Discussion
5. Ongoing Work

MOTIVATION

Scenario

We focus on recognizing complex activities like "watering plants" or "taking medicine".

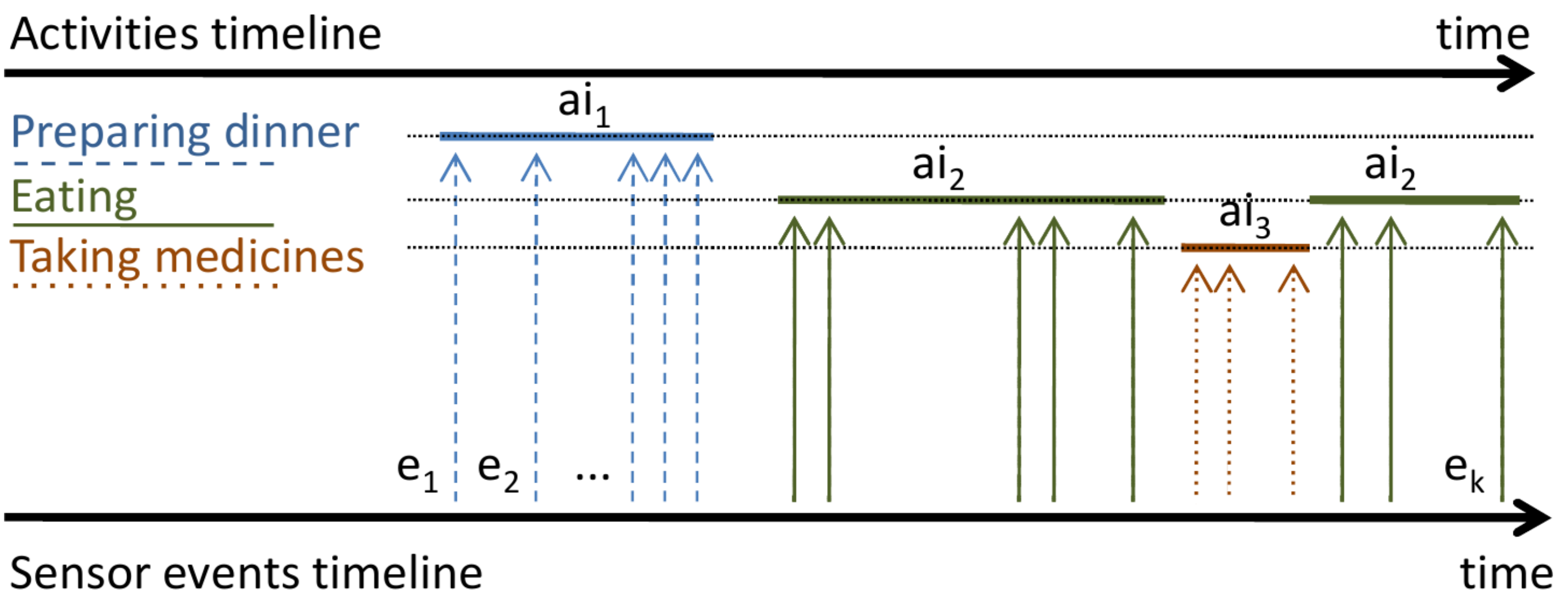
- So far we relied on...

➡ ... ontological and probabilistic reasoning (as it overcomes well-known limitations)

- What we are targeting right now ...

➡ ... online recognition and active learning

Scenario


Activities timeline
time
ai_1
Preparing dinner
Eating
Taking medicines
ai_2
ai_3
ai_2
e_1
e_2
...
e_k
Sensor events timeline
time

Motivation

Until now we used Markov Logic Networks (RockIt) to model our scenario, but ...

- ... weights are not this intuitive as probabilities
- ... much computational effort
- ... did not support time-constraints and marginal inferences (RockIt).

➡ We decided to investigate ProbLog in respect of Activity Recognition

Idea

ProbLog is a probabilistic extension of Prolog which allows to explicitly define probabilistic facts and rules.

- Segment sensor stream into windows

→ Build for each window a ProbLog program

  - Allows to query the user in almost real-time
  - "Marginal Inferences" allows to apply rules but also interpret the results across windows

Contribution

- We clarify the benefits of *ProbLog* for activity recognition

  ➡ We introduce a guidance on how to write *ProbLog* programs.

- We present a *ProbLog*-based method to recognize activities of daily living in an online fashion
- We show potential advantages of ProbLog's marginal inference

MODEL & SYSTEM

System

1. Extracting probabilities (sensor type ↔ activity)

➡ extracted from the dataset (supervised)

2. Segmentation

➡ Standard windowing approach with a fixed size

3. ProbLog Program Generation

➡ As soon as a window is finalized we generate the ProbLog program

4. Online Activity Recognition

➡ We choose the most likely activity

Building a ProbLog Program (1/4)

Event and Instance Clauses (Example)

1.0::**event**(e_1, water, 5).
1.0::**event**(e_2, absent, 6).

➡ This implies that these two clauses have to be part of each possible world (ground truth).

0.5::**instance**(ai_1, ac_1, 0, 7); 0.5::**instance**(ai_1, ac_2, 0, 7).
0.5::**instance** (ai_2, ac_1, 4, 10); 0.5::**instance**(ai_2, ac_2, 4, 10).

e.g. cleaning

XOR

predicate

Building a ProbLog Program (1/4)

Probabilistic Facts (Example)

0.9::**related**(X, watering) :- **event**(X, water, T)

➡ allows to incorporate mined probabilities

Temporal Constraints (Example)

closeAfter(T_1,T_2) :- T_1>T_2,T_3 is T_1-T_2, T_3<3.

0.6::**producedBy**(X_2,I) :- **event**(X_1,Y_1,T1), **event**(X_2,Y_2,T_2), **closeAfter**(T_2,T_1), **producedBy**(X_1,I)

➡ Intuitively, temporally close events more likely belong to the same activity instance.

Building a ProbLog Program (1/4)

Knowledge-based Facts (Example)

```
0.9::bond(Y, waterplants) :- event(X1, water, T1), event(X2,
can, T2), closeAfter(T1,T2), producedBy(X1,Y).
```

➡ We assume that "can" is only used for "waterplants"

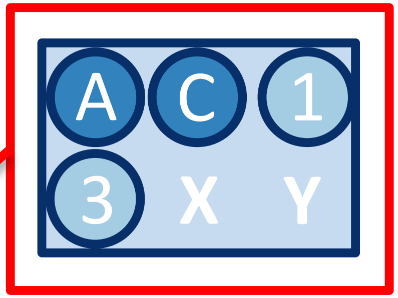
Domain Constraints (Example)

r1 :- **related**(e_1, ac_1), \+**related**(e_1, ac_2);
related(e_2, ac_1), \+**related**(e_2, ac_2);
evidence(r1,true).

➡ We want to be sure that each sensor event is assigned to exactly one activity.

Executing a ProbLog Program

An instance of our program



We have to formulate a query

A instance(ai_1,ac_1,0,7)

1 producedBy(e_1,ac_1)

X related(e_1, cleaning)

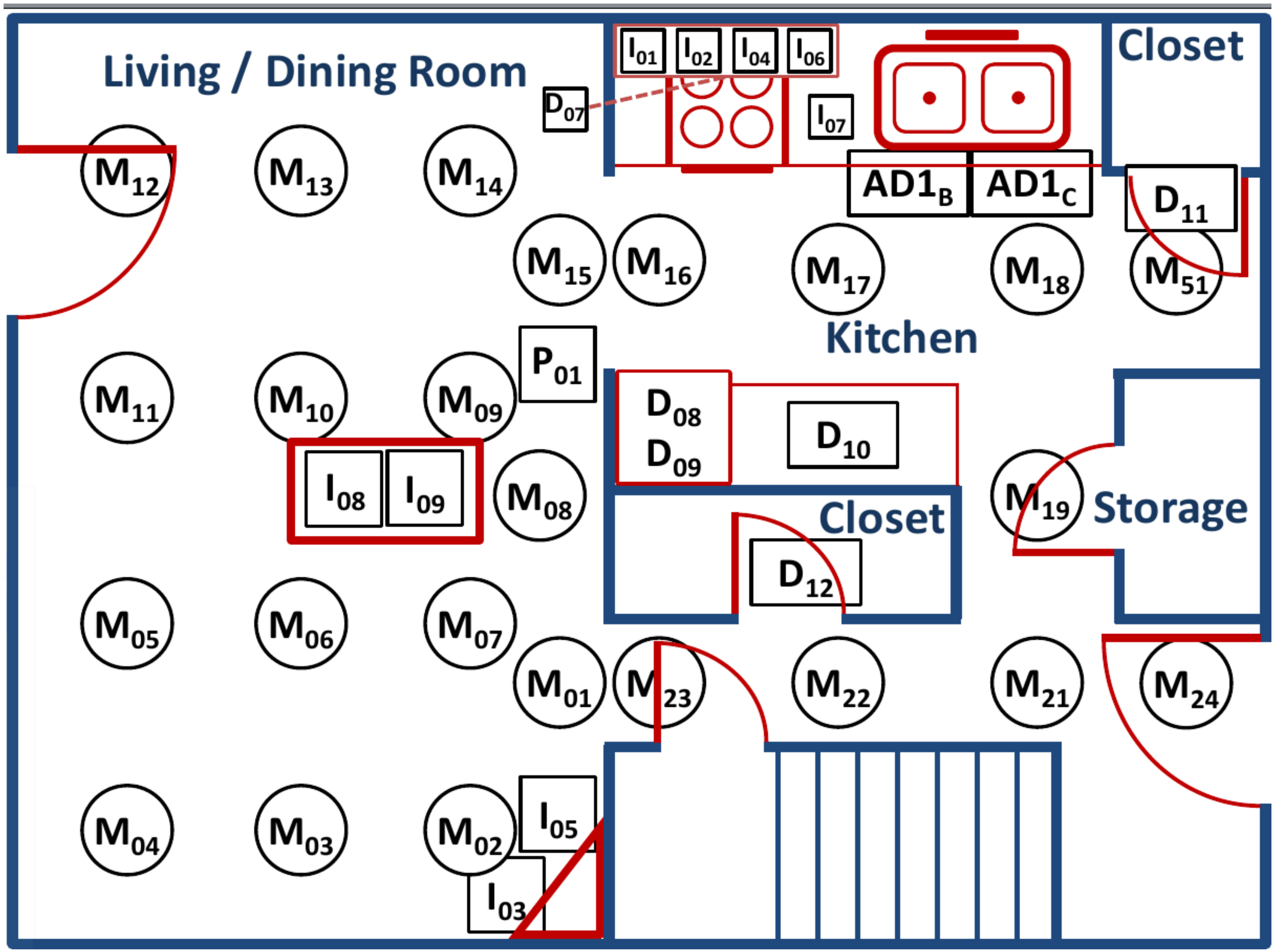
Each of them has a probability

Probability of a certain program instance

The probabilities of all instances answer our query

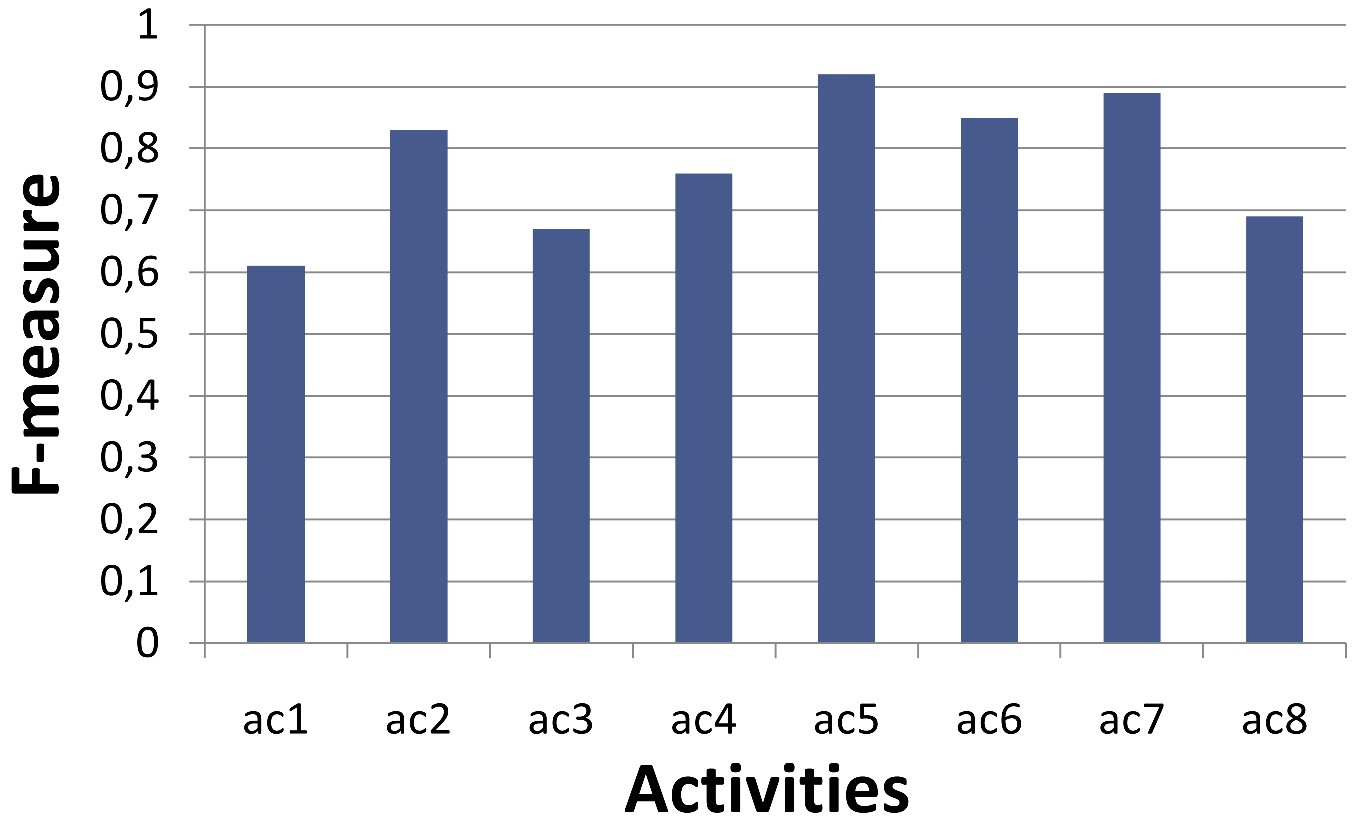
PRELIMINARY RESULTS

Dataset: CASAS



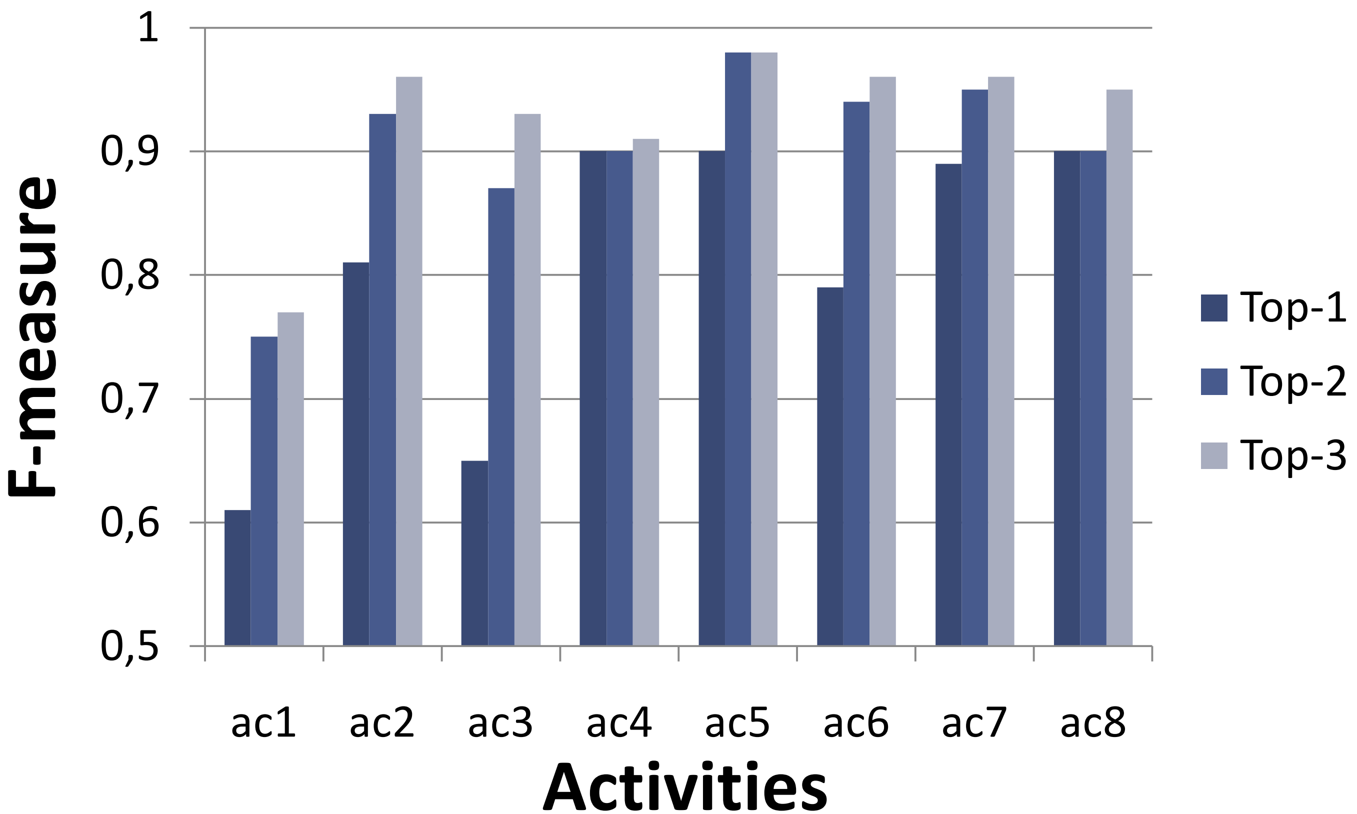
CASAS: A smart home in a box, D.J. Cook et al., 2013

Results: Recognition Rate



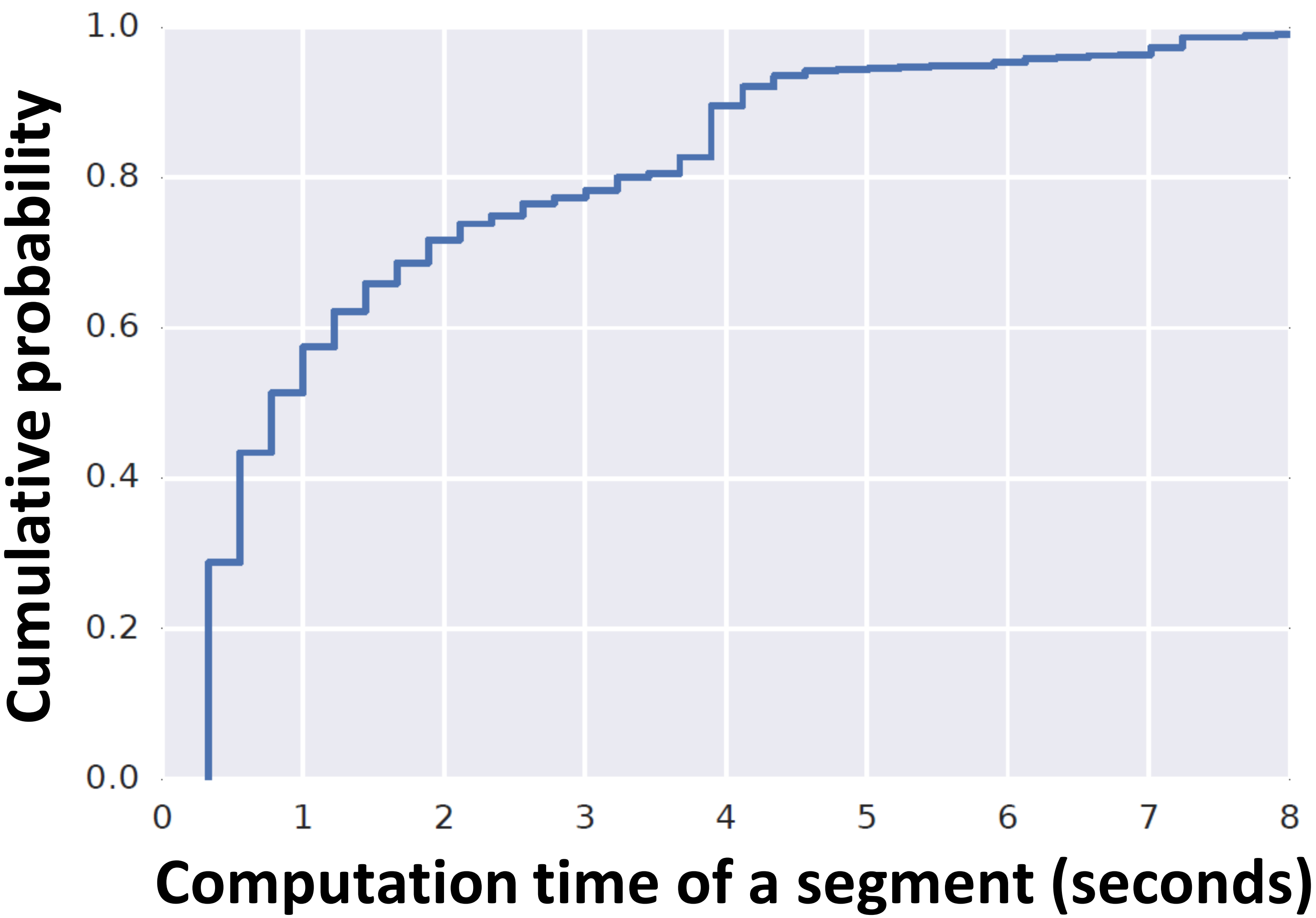
The activities are well recognized but there is a gap between the worst and best result (ac_1 vs. ac_5)

Results: Marginal Inference



This militates for the quality of the ranking but also for the reliability of the approach.

Runtime



1.0
0.8
0.6
0.4
0.2
0.0
Cumulative probability
0
1
2
3
4
5
6
7
8
Computation time of a segment (seconds)

DISCUSSION

Discussion

The results look promising but ...

- ... difficult to compare to existing works due to different setups
- ... only a standard windowing approach
- ... probabilities were extracted from the dataset
- ... Interleaved activities are not explicitly modeled

ONGOING WORK

Ongoing Work

Our next steps are ...

- ... fully exploit marginal inferences

➡ refine the classification of the windows

- ... investigating active learning for personalization (e.g. correcting/adapting probabilities)

➡ Currently, in another work, we focus on suitable online and active learning techniques

Thank you for your attention :)

CoMoRea´18