Tool Support for Activity Recognition with Computational Causal Behaviour Models

Kristina Yordanova, Frank Krüger, and Thomas Kirste

University of Rostock, Institute of Computer Science, MMIS Group, Germany
{kristina.yordanova,frank.krueger2,thomas.kirste}@uni-rostock.de
http://mmis.informatik.uni-rostock.de

Abstract. Context-aware activity recognition plays an important role in different types of assistive systems and the approaches with which the context information is represented is a topic of various current projects. Here we present a tool support for activity recognition using computational causal behaviour models that allow the combination of symbolic causal model representation and probabilistic inference. The aim of the tool is to provide a flexible way of generating probabilistic inference engines from prior knowledge which reduces the need for collecting expensive training data.

Keywords: Activity Recognition, Context-Awareness, Causal Models

1 Introduction and Motivation

The area of activity recognition is expanding rapidly in the resent years which results in the need for powerful and reliable tools for building models for activity recognition and for inferring the user actions and intentions. In this paper we present such tool support that allows the building of Computational Causal Behaviour Models (CCBM) and their usage for recognising user activities. CCBM are human behaviour models that use symbolic causal representation to describe activities and which are compiled into probabilistic inference machines. Symbolic human behaviour models are well known to the activity recognition community as they allow the representation of user actions and the reasoning over them in order to infer not only the current user actions but also to what more complex activity it belongs [3, 1, 4].

On the other hand, as we are dealing with observations produced from unreliable sensors, the symbolic representation is not able to cope with the implications of the noisy readings. Thus, probabilistic models are often preferred in situations where the inference is applied under some uncertainty [6].

To cross the bridge between these two approaches, here we present our tool that compiles causal human behaviour models into a probabilistic inference machine, taking advantage of the features of both methods. The tool is aimed at providing an estimation of the trajectories of dynamic systems from observation data in domains where prior knowledge and context information can be used to reason over the causality of the user behaviour.

Furthermore, CCBM synthesises probabilistic models from prior knowledge aiming at reducing the need for training data. This is done by substituting the training data with

S. Wölfl (Ed.): Poster and Demo Track of the 35th German Conference on Artificial Intelligence (KI-2012), pp. 108-112, 2012.

context information based on room topology, types of sensors, typical actions durations and functionality of a given setting.

2 CCBM Features and Goals

The functionality and feature set of the CCBM tool has been designed with the aim of providing reliable and flexible system for activity recognition that is able to cope with real world problems. During the tool design, the following goals were kept in mind. (1) allow to infer goals not only as labels but also as desired states that would allow the system to plan corresponding assistance strategies; (2) allow to recognise individual actions and to infer plans that lead to specific goals, i.e. intentions; (3) allow to recognise environmental states and predict actions and intentions from these states; (4) ensure mechanism for coping with noisy or unreliable sensor data, i.e. probabilistic sensor models; (5) reduce the need for training data by employing context information, i.e. a-priori models; (6) enable the use of symbolic causal modelling paradigms for building a-priori models; (7) allow the development of reusable inference models; (8) allow the inference in large state-spaces; (9) allow the inference of actions in multi-agent scenarios and the recognition not only of the team behaviour but also of the individual agents; (10) allow probabilistic duration densities that can cope with variations in the expected actions durations; (11) allow real-time inference; (12) employ probabilistic inference, i.e. Bayesian filtering, smoothing, prediction, parameter estimation; (13) allow the usage of different or combined heuristics for action selection in order to adequately represent the dynamics of human behaviour

3 CCBM System Components

Based on the features and goals described in the previous section, the CCBM tool consists of different components that provide the tool functionality.

Computational Causal Behaviour Models: The CCBMs consist of symbolic causal human behaviour model described in a PDDL-like notation and an observation model that are translated into a probabilistic inference system. The symbolic model consists of two parts – domain and problem descriptions. The domain description containing the available user actions represented as precondition-effect operators, the object types used and the domain constants. A formal definition of the domain file can be seen in Fig. 1 – a typical domain entry consists of declaration of the types, declarations of the constants invariant for different problems, declaration of the observations that connect the described states to the observations, and declaration of the actions described as precondition effect formula.

On the other hand, the problem description contains the problem constants, the initial world state and the goal. The formal definition of the problem file can be seen in Fig. 2, where every typical problem entry contains the domain name, the problem specific constants (or objects), the initial state to which also a duration could be assigned, and the goal state. The third component needed for compiling the model into a probabilistic filter is the observation model (OM) that describes the probability of observing a state

```
domain = (define (domain name){domain-entry})

domain-entry = (:types {name{name} - name}{name})
       | (:predicates  {(name[declarations])})
       | (:constants {constant{constant} - type-name}{constant})
       | (:observation {observation-formula})
       | (:action {action-formula})
```

Fig. 1: CCBM domain definition components

```
problem = (define (problem name){problem-entry})

problem-entry = (:domain name)
       | (:objects {constant{constant} - type-name}{constant})
       | (:init [:duration atomic-formula]{init-elem})
       | (:goal formula)
```

Fig. 2: CCBM problem definition components

x given an observation y $P(y|x)$. The OM could be trained on sensor data,but it also could rely only on context information.

In difference with other human behaviour models [7], in CCBM one does not need to explicitly describe the execution sequence of the user actions. They are rather compiled lated based on the preconditions and effects of every action. Thus, the model is able to describe multiple hypotheses without the need for the system designer to waste time on describing these sequences. Additionally, the causal actions description allows for producing much more valid execution sequences than when doing that by hand. In short, by describing only a small set of actions, one could generate huge state space with all logically valid executions.

Compiler: The compiler takes the domain and the problem files and translates them into a C-code module which is compiled and linked against the modules containing the filter routines. Additionally, the OM containing the routines for computing the observation probabilities is compiled and linked into the final executable.

Analyser: The analyser computes the state space of the causal model (the domain-problem combination) and calculates the distance to the goal. The state-space is computed by a depth-first search of the state graph with edges the available actions. Later the goal distances are calculated by running the Dijkstra's algorithm on the transposed graph and states that are not reachable are assigned a goal distance of infinity.

Filter: The filter uses the compiled model and filters the sensor observations by using an action selection formula. For an action a and states x, x' such that $x' = a(x)$, let $\delta(x')$ be the goal distance of state x', and let $s(a)$ be the saliency value of a. The probability of selecting a in state x is then defined by:

$$P(a|x) \propto \begin{cases} \gamma(x')s(a)(\beta + e^{\lambda\delta(x')}) & \text{if } x \models pre(a) \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

where β is the *bias* and λ is the *weight factor*. Note that in the presence of states with infinite goal distance, it is required that $\lambda \leq 0$. The factor $\gamma(x')$ is determined by the history: if x' has not been visited before, $\gamma(x')$ equals 1. Otherwise, it will be 0. This forces the system not to re-enter states it has already visited. The term $e^{\lambda\delta(x')}$ assumes that an agent will pursue its goal based on a Boltzmann policy [5].

Validator: The plan validator is developed as a help tool for the model developer to check if a given plan is valid according to the model description. The validator can output three possible plan outcomes: the plan is successful and the goal is reached; the plan is successful but the goal is not reached; and the plan has failed. The third output provides the additional extremely useful information of the time slot and agent where the preconditions for successful plan execution were not met.

4 Application Domains

The CCBM is developed with the idea to perform activity recognition in context-aware domains where prior knowledge could be used for substituting training data and thus reducing the need for training the model. To use the tool for inferring the user activity and intention, a CCBM model of the domain in question is build beforehand, containing the set of actions, their durations and any additional problem specific information. The tool then compiles the model into probabilistic inference machine and uses an HMM (for exact inference) or a particle filter (for approximate inference) in order to estimate the user state (at present forward filtering is used). For estimating the user goal, the model could be run in parallel with different initial and goal states and the one with the highest likelihood is assumed to be the user goal. However, at present the tool evaluation was centered on the activity recognition process and the intention (goal) recognition is a matter of current and future research.

So far the tool was successfully used in a smart meeting room scenario where several 3-Person meetings with different agenda and durations took place. The CCBM tool was able to recognise the performed team and agent activities relying only on context information with accuracy of about 90% [8].

Another application domain of the CCBM tool is an office scenario where two colleagues act autonomously while preparing coffee, fixing the printer and printing documents. Although the tool was receiving only scarce sensor information (location sensors detected whether a person is standing next to one of the objects in question) and was heavily relying on causal reasoning, it was able to recognise the executed activity and to provide a reasonable plan execution sequence of the two non-interacting agents [2].

Furthermore, the tool is currently being applied to a kitchen task assessment problem where the preliminary results show that the tool is successful in reasoning not only about the performed action but also about the objects being used, regardless of the fact that the observed data represents only the actions.

5 Conclusion and Future Work

In this paper we presented a tool support for activity recognition using computational causal models. The tool is able to combine symbolic causal model representations with probabilistic sensor information in order to perform probabilistic activity recognition. Among other features, the tool provides real time inference and tracing of multiple users, thus the ability to recognise the team behaviour and the separate users goals. The CCBM tool was successfully applied to two different domains and is currently being used for building models for a third domain. In the future, the tool functionality will be extended (e.g with sub-models for movement trajectories) and it will be tested more extensively on activity recognition problems from the daily living domain. Additionally, it will be applied to problems where not only the current activity has to be recognised, but also the future user goal, or intention, thus providing a vital information for smart assistance systems that need to configure themselves in order to assist the user proactively.

References

1. S. Miksch K. Kaiser. Treating Temporal Information in Plan and Process Modeling. Technical Report Asgaard-TR-2004-1, Institute of Software Technology and Interactive Systems, Vienna University of Technology, Vienna, 2004.
2. Frank Krüger and Thomas Kirste. Synthesizing sequential bayesian filters for plan and activity recognition from extended precondition-effect rules. In *Adj. Proc. 2nd International Joint Conference on Ambient Intelligence*. Springer, 2011.
3. M. Wurdel and C. Burghardt and P. Forbrig. Supporting ambient environments by extended task models. In *Proc. AMI07 Wor. Model Driven Software Engineering for Ambient Intelligence Application*, 2007.
4. G. Okeyo, L. Chen, H. Wang, and R. Sterritt. Ontology-Based Learning Framework for Activity Assistance in an Adaptive Smart Home. In *Activity Recognition in Pervasive Intelligent Environments*, volume 4. Atlantis Press, 2011.
5. M. Ramirez and H. Geffner. Goal recognition over pomdps: Inferring the intention of a pomdp agent. In *Proc. 21st International Conference on Automated Planning and Scheduling*, 2011.
6. T. van Kasteren, A. Noulas, and G. Englebienne B. Krse̎. Accurate activity recognition in a home setting. In *Proc. 10th International Conference on Ubiquitous Computing*, 2008.
7. Kristina Yordanova. Toward a unified human behaviour modelling approach. Technical Report CS-02-11, Institut für Informatik, Universität Rostock, Rostock, Germany, May 2011. ISSN 0944-5900.
8. Kristina Yordanova, Frank Krüger, and Thomas Kirste. Context aware approach for activity recognition based on precondition-effect rules. In *Proc. Wor. COMOREA, PerCom'12*, 2012.