Improving Activity Prediction and Activity Scheduling in Smart Home Networks for Enhanced QoS

Koteswara Rao Vemu

Samsung R & D Institute India-Bangalore
koti.vemu@samsung.com

Abstract. This paper proposes an algorithm, to enhance the prediction accuracy of inhabitant activities in smart home networks. This work is an enhancement to SPEED [1], which was earlier drawn upon [2,3]. It works with the nested episodes of activity sequences along with the innermost episodes to generate user activity contexts. For a given sequence, our approach on an average predicts 86 percent accurately, which is much better than SPEED's 59 percent accuracy.

1 Introduction

The daily routines of inhabitants generate activity sequences in Smart Home Networks, which follow a pattern for each inhabitant. Historic activity sequences can be used to predict user behavior and give reminders about upcoming activities. We have adopted an algorithm SPEED [1], which is drawn from [2,3]. SPEED categorizes sequence of user activities into episodes and generate contexts. These contexts are used for predicting inhabitants activities using prediction by partial matching algorithm (PPM). For details about PPM one can refer to [1].

In a sequence, where multiple nested episodes are possible, SPEED only considers the inner most episode for identifying context frequencies, which results in reduced prediction accuracy. Our proposed algorithm overcomes the limitations mentioned above and results in better prediction with a faster convergence rate. Also, resource management and concurrent activity scheduling is an important functionality of any multi-resource Smart Home solution, having more resources of same type (multiple TV sets). The better the resource management, the better is user experience in a Smart Home. For example, consider a smart home solution deployed in a 2 user ($User1$, $User2$) environment . Assume there are 2 TV sets installed in a smart home, one in living room and another in bed room of $User1$. Consider that based on the past history of $User1$ activities, there is a prediction that Sunday $7:30pm$, a scheduled TV show to be watched by $User1$. At the same time $User2$ has a weekly show scheduled at $7:30p$m on Sunday. So in this situation, smart home should be able to allocate living room TV set to $User2$ and suggest $User1$ to watch show in bed room. This idea provides an algorithm to handle concurrent activities in Smart Home environment through resource management.

R. Natarajan et al. (Eds.): ICDCIT 2015, LNCS 8956, pp. 261–264, 2015.

2 Predicting and Scheduling User Activities in Smart Home

In this section, we propose our algorithm to improve prediction accuracy of inhabitant activities and we use inhabitant and resident interchangeably. Using the activity sequence, we can predict inhabitant's activities based on certain parameters, such as day and time. SPEED [1] categorizes sequence of events of inhabitant into episodes, we use the same approach here as well. And we represent start of an activity with a capital letter and finishing of an activity with a small letter, same as SPEED [1]. A sample activity sequence can be represented as $KCRrckAaKRCcrk$, we will use this sequence to compare our results with SPEED. An episode is defined as all activities between starting an activity and ending an activity both inclusive. For Example $KCRrck$ is an episode, $CRrc$ is another episode. Once the episodes are identified, our approach as that of SPEED [1] uses them to generate contexts, which will be used to create context trees. These context trees are used for predicting inhabitant's activities using PPM as below.

$$P_k(\psi) = p_k(\psi) + e_k(\psi).P_{k-1}(\psi) \tag{1}$$

Where, ψ is the symbol to predict, $P_k(\psi)$ is the final probability of symbol ψ, k is the length of the phase, $p_k(\psi)$ is the probability of seeing the ψ symbol after the k length phase, $e_k(\psi)$ is the escape probability (probability of null outcomes) of symbol ψ. Using SPEED algorithm, the contexts are getting generated from smallest episode, which is available in window, refer [1] for algorithm and generated contexts for an episode. In our example sequence $KCRrckAaKRCcrk$, SPEED considers only Aa episode for generating contexts, even though there are other episodes, such as $KCRrck$ and $CRrc$. It treats the remaining episodes as noise data generated in smart home. Our approach considers outer episodes along with inner episodes, results in extracting more contexts from a given sequence. While considering nested episodes, we ensure de-duplication of episodes. Also we propose a model to allocate resources to inhabitant in a multi-inhabitant and multi-resource environments [4,5]. Nowadays, it is quite common to have multiple appliances/resources of same kind in homes; e.g., having a TV set in living room, bed room and drawing room. All these having restricted access for users, e.g., all users can have access to living room TV set, but access is restricted for TV sets that are installed in bedrooms. We form a bipartite graph between inhabitants and resources by constructing edges between inhabitant and accessible resource. While predicting the activity, the smart home controller is able to find out what are all the available resources to allocate and inform to the user.

3 Simulation and Results

For simulations, we considered user sequence as $KCRrckAaKRCcrk$, which is already mentioned in previous section. We have generated list of all possible contexts using SPEED and our approach, Table 1 represents the comparison of all possible contexts along with their frequencies.

Table 1. Comparison of all possible contexts using SPEED and Our Approach

Possible Contexts using SPEED	Possible Contexts using Our Approach
R(1),r(1),A(1),a(1), C(1),c(1),Cc(1), Aa(1),Rr(1)	A(1),a(1),K(2),k(2),C(2),c(2),R(2),r(2),Rr(1),rc(1),KC(1), ck(1),Aa(1),Cc(1),RC(1),cr(1),KR(1),rk(1),Rrc(1),KCR(1), rck(1),CRr(1),RCc(1),Ccr(1),crk(1),KRC(1),CRrc(1),KCRr(1), Rrck(1),RCcr(1),KRCc(1),Ccrk(1),KCRrc(1),CRrck(1), KRCcr(1),RCcrk(1),KCRrck(1),KRCcrk(1)

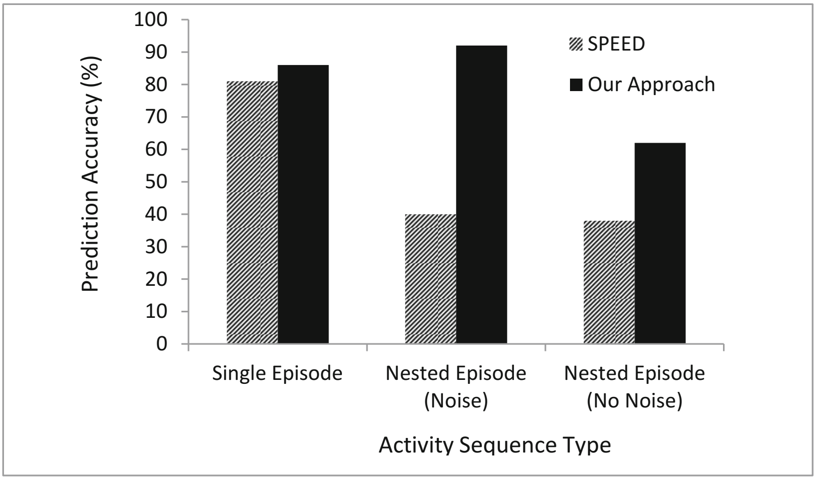

Fig. 1. Comparison of SPEED and our approach

3.1 Example

Now let us see an example of how to calculate the probability of events using both the approaches using the previous sequence $KCRrckAaKRCcrk$. Assume that our current window state is K and we try predicting an event from current window state. We assume all of the suffixes of window $(K, null)$ to calculate the probability of the next event. Suppose that we have to estimate the probability of the next event of switching on Coffee Machine (C). **Using SPEED**, there are two contexts $(K, null)$, from which we can estimate the probability of event C. From context K, probability of event C after context $K = 0$, escape probability is 0. From context $null$, probability of event C after context $null = 1/6$. The total probability of event r occurring after window sequence K is $0+0(1/6) = 0$. **Using Our Approach**, there are two contexts $(K, null)$, from which we can estimate the probability of event C. From context K, probability of event C after context $K = 1/2$, escape probability is 0. From context $null$, probability of event C after context $null = 2/14$. The total probability of event r occurring after window sequence R is $1/2 + 0(2/14) = 0.5$. In this example, the probability of predicting C from Sequence K is not 0 as in SPEED, because after K there is at least one instance of C happened. In this example, our algorithm predicts the probability of events more accurately than that of SPEED. The comparison of both the approaches for various activity sequences are shown in Fig. 1. In Fig. 1 both SPEED and our approach results in almost equal prediction accuracy in case of sequence having single episode (with or without noise data), where as in nested episode sequences with noise data our approach is performing much better than SPEED. Even in Nested episodes with no noise data also, our algorithm is predicting more accurately than SPEED. Overall our algorithm's prediction accuracy is much better than that of SPEED and provides better QoS to inhabitants.

4 Conclusion and Future Work

We proposed an algorithm for predicting inhabitant activities in smart home environments. Our approach considers user activities in all the episodes, without losing any activity in a given sequence. Hence it results in generating more contexts than that of SPEED, this leads to better prediction accuracy and better convergence rate compared to SPEED. As a future work, one can try to extend the research on predicting user activities in a multi-inhabitant environments with bigger activity sequences and more nested episodes.

References

1. Alam, M.R., Reaz, M.B.I., Ali, M.A.M.: SPEED: An inhabitant activity prediction algorithm for smart homes. IEEE Transactions on Systems, Man, and Cybernetics, Part A 42(4), 985–990 (2012)
2. Gopalratnam, K., Cook, D.J.: Active lezi: an incremental parsing algorithm for sequential prediction. International Journal on Artificial Intelligence Tools 13(4), 917–930 (2004)
3. Gopalratnam, K., Cook, D.J.: Online sequential prediction via incremental parsing: The active lezi algorithm. IEEE Intelligent Systems 22(1), 52–58 (2007)
4. Roy, N., Misra, A., Cook, D.: Infrastructure-assisted smartphone-based ADL recognition in multi-inhabitant smart environments. In: PerCom, pp. 38–46. IEEE Computer Society (2013)
5. Crandall, A.S., Cook, D.J.: Using a hidden markov model for resident identification. In: Callaghan, V., Kameas, A., Egerton, S., Satoh, I., Weber 0001, M.(eds.) Sixth International Conference on Intelligent Environments, IE 2010, Kuala Lumpur, Malaysia, July 19-21, pp. 74–79. IEEE Computer Society (2010)