A Windowing Approach for Activity Recognition in Sensor Data Streams

Fadi Al Machot, Heinrich C. Mayr and Suneth Ranasinghe
Institute of Applied Informatics
Alpen-Adria Universität Klagenfurt, Austria
Email: firstname.lastname@aau.at

***Abstract*— Determining the appropriate data window size for online sensor data streams to recognize a specific activity is still a challenging task. In particular, when new sensor events are recorded. This paper proposes a windowing algorithm which presents promising results to recognize complex activities, e.g., in a smart home environment. The underlying basic idea is to analyze the sensor data in order to identify the set of "best fitting sensors": it contains those sensors that most contribute to the recognition task, and therefore should be considered in a window. To validate our approach, we applied it on the CASAS data set which is an international data set for activity recognition. Based on the promising results, we believe that this algorithm can assist to detect human activities. Thus, our approach might be used in Active and Assisted Living Environments (AAL), where activity recognition is required to distinguish the types of help, a person needs to master his/her daily life activities.**

***Index Terms*— Activity recognition, Smart homes, Pattern Recognition, Active and Assisted Living (AAL), Pervasive Computing**

I. Introduction

Active and assisted living aims at supporting older people in their daily activities using intelligent environments. Such kind of support requires to know about a person's current behavior, and therefore a robust recognition of her/his actions.

For this purpose, "smart homes" [1] are equipped with different types of sensors to extract information from the environment, which helps to detect simple events like closing, opening a door or turning on/off devices. The combination of such simple events can facilitate the detection of complex events like preparing a meal or preparing a drink. This leads to the question which chunk of a sensor data stream reflects a complex event. Typically, chunks are determined by windowing the data stream [3]. In this paper, we present a dynamic windowing approach, which estimates, for each activity, the appropriate window size. This can help to avoid situations in which (a) the sensor data of an activity is spread over several different chunks or (b) the window size, i.e. the set of simple events, is not sufficient to predict a specific activity.

The methodology has been developed within the realm of the Human Behavior Monitoring and Support (HBMS[1]) project [2] that aims at deriving support services from integrated models of abilities, and from episodic knowledge an individual had or has temporarily forgotten.

The paper is organized as follows: Section 2 gives an overview of the state-of-the-art approaches to sensor data chunking and its limitations. In section 3 we shortly describe the CASAS data set before presenting our dynamic windowing approach in section 4. Section 5 lists the extracted features for evaluation. In section 6 we outline the obtained results and discuss the overall performance. The paper ends with a conclusion.

II. Related Work

In order to provide robust activity related information for real-life applications based on sensor data, researchers suggested different approaches. Commonly, they apply a windowing technique such that each window identifies a data stream chunk that possibly indicates a specific activity.

There are three major windowing techniques:

***Time periods of equal length*:** A simple method to learn activity models during the training phase. However, some activities might spread over more than one time slice [4,5].

Chunks of equal cardinality (equal number of sensors): This approach comes with the advantage that the resulting windows cover varying time periods. However, there might be chunks, which contain not enough data for the detection of a complex event [6].

Probabilistic dynamic windowing: Maximizes the probability of the most likely window size for a specific activity. The idea is to incorporate the time decay and mutual information, using weightings of sensor events within a window [3]. A limitation of such approaches is its inefficiency in modeling the 'Other' complex events where the exact complex event is not known.

III. Data set

The CASAS project [7] deals with the support of residents in a smart homes environment where sensors sense the environment to receive the status of the inhabitants. For our evaluations, the Aruba CASAS dataset is used to validate our approach. It has been collected in a house which has a single bedroom, a kitchen, a bathroom, a dining room and also an office. The home Aruba is "smart" as 34 sensors are installed to collect information like door closure, motion and temperature.

[1] This work was funded by Klaus Tschira Stiftung GmbH, Heidelberg

All activities are collected by a single inhabitant within the period from 2010-11-04 to 2011-06-11. The following activities are annotated in the dataset: Sleeping, Bed to Toilet, Meal Preparation, Dish Washing, Relaxing, House Keeping, Eating, Resperate, Leave Home, Enter Home, and Work

IV. Dynamic Windowing (DW)

As already mentioned in the introduction, we aim at a dynamic windowing technique in order to avoid the previously mentioned disadvantages.

The basic concept is to analyse the dataset based on different features in order to determine the optimal number of activated sensors: "*best fitting sensor set*" for each activity [8].

The overall approach is structured in two major phases:

a. Learning phase: Find the best fitting sensors for each activity,
b. Online phase: Collect sensor data in a window until the elements of a set of best fitting sensors are activated.

Learning Phase:
In order to determine the best fitting sensors set per activity, we extracted several features[2] of the given (annotated) dataset. The impact of these features is analyzed by using the Information Gain (IG) attribute evaluation [9]. The results showed that the number of activations for each sensor was the most relevant attribute. The overall general steps can be summarized in Figure 1:

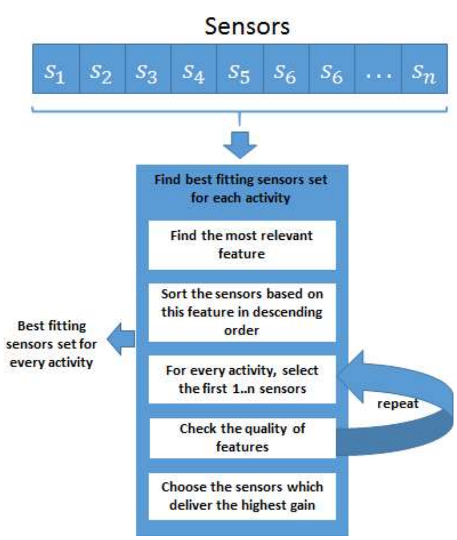

Figure 1: The steps to find the best fitting sensors

The output would be the best fitting sensors for each activity. The previous steps can be applied on any dataset.

Online Phase:
To dynamically determine from a sensor data stream the (next) window, we use the following algorithm:

Input:

BestSensors(J,K_j), the best fitting sensors (K_j) for every activity j, J is the total number of activities;

tmpBestSensors(J,K_j), a reference structure initialized with zeros;

Result: the algorithms stops, if with the current stream element s, the cells of a tmpBestSensors row are set to 1. I.e., by s a window is"closed".

```
WHILE GetNextSensor.HasNext()
GetNextSensor (s);
    For j=1 to J
        For k=1 to K
            If (s == BestSensors (j,k))
                tmpBestSensors(j,k)=1;
            EndIf
        EndFor
    EndFor
    For p=1 to J
        If (tmpBestSensors(p) does not contain zero)
            Exit (with p);
        EndIf
    EndFor
EndWhile.
```

V. Features

To prepare for the evaluation of our approach, we identified a set of features out of the Aruba CASAS dataset; for this purpose we

- considered only the event-based sensors (30 out of 34), and 9 of the 11 activities. The latter restriction was due to the low number of annotated samples of the excluded 2 activities;
- created for each window an occurrences histogram with the dimensions Sensor_Id and Number_of_Activations.

From such occurrences histogram, we extracted the following features,

- the histogram vector,
- the sum of all activations (best sensors histogram),
- the mean value of the occurrences histogram, and
- the sum of the number of activations for other (non-best fitting) sensors.

[2] Number of activations of each sensor, activation duration of each sensor, duration of each activity, time of performing every activity, number of activated sensors for each activity, and location of the sensor

VI. Results

To assess the quality of the approach, we applied the 10-fold cross validation using multinomial logistic regression classifier which generalizes logistic regression to multiclass problems [10]. The evaluation has been performed using the data mining and machine learning software WEKA [11].

Table 1 shows the results represented in 6 different Evaluation metrics where TP is true positives, FP is false positives, Pr is the precision, Re is the recall, F-M is the F-Measure, and ROC is the Receiver-Operating-Characteristic Curve. The table demonstrates that the class C7 does not appear in the evaluation due to the fact that after applying our windowing approach, the number of remaining samples for training is very small. Additionally, the overall F-Measure value is 0.87, although the worst result belongs to C5, C8, and C9. This can be explained due to the imbalance problem of the given dataset.

Table 1: The overall performance of the proposed windowing technique, *where C1: Sleeping, C2: Bed to Toilet, C3: Meal Preparation C4: Relaxing, C5: House Keeping, C6: Eating, C8: Leave Home: C9: Enter Home and C10: Work*

Class	TP	FP	Precision	Recall	F-measure	ROC Curve
1	0.88	0.002	0.94	0.88	0.91	0.98
2	0.63	0.001	0.64	0.63	0.63	0.99
3	0.98	0.29	0.88	0.98	0.93	0.90
4	0.78	0.015	0.93	0.78	0.85	0.95
5	0.11	0.001	0.54	0.11	0.19	0.76
6	0.46	0.001	0.80	0.46	0.58	0.85
8	0.33	0	0.15	0.33	0.21	0.99
9	0.26	0	0.36	0.26	0.30	0.83
10	0.77	0	0.85	0.77	0.81	0.97
Weighted AVG	0.89	0.20	0.86	0.89	0.87	0.91

The imbalance problem is reflected in Figure 2 which shows the percentage of training data quantity for C5, C6, C8 and C9 is 1%, 4%, 6.7% and 2.7%, respectively. In contrast to that, the classification model for other classes C1, C2, C3, C4 and C10 show a high performance. We believe that if the given dataset did provide more samples for training, the performance for each activity will improve. This can be explained by the fact that during our feature evaluation problem, the quality of feature varied between 0.52 and 0.12, using Info Gain (IG) Attribute Evaluation approach [9].

VII. Conclusion

Activity recognition demands a detailed analysis and full understanding of the environment in which the activities occur. Our work proposed a dynamic windowing technique supported by spatial robust features. We demonstrated that the proposed windowing algorithm is a solid basis for having a strong tool to detect complex activities in smart houses. Our future work will focus on considering different residents living in the same house. We will try to implement the proposed approach in our HBMS lab for more complex real life scenarios.

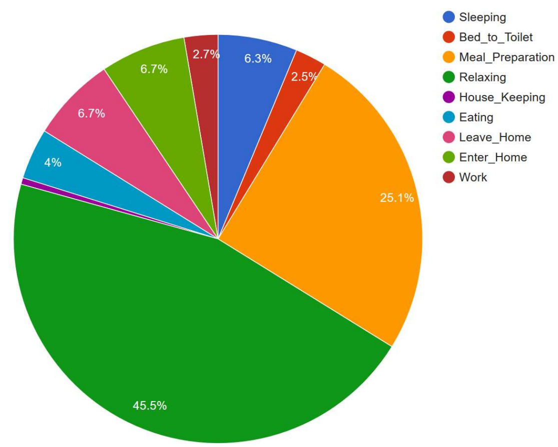

Figure 2: The overall distribution of all activities in the dataset

References

[1] A. M. Raisul, M. Bin Ibne Reaz, and M. Alauddin Mohd Ali. "A review of smart homes—Past, present, and future." Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on 2012: pp 1190-1203.

[2] J. Michael, A. Grießer, T. Strobl, and H. C. Mayr, “Cognitive modeling and support for ambient assistance”, in: Information Systems: Methods, Models, and Applications. Springer, 2013, pp 96–107.

[3] N.C. Krishnan, and D. J. Cook, "Activity recognition on streaming sensor data", Pervasive and mobile computing 10 (2014): pp 138-154.

[4] T. van Kasteren, G. Englebienne, and B. Krse, “An activity monitoring system for elderly care using generative and discriminative models”, Journal Personal and Ubiquitous Computing, Special Issue on Pervasive Technologies for Assistive Environments. 2010: pp489–498.

[5] N.C. Krishnan and S. Panchanathan, “Analysis of low resolution accelerometer data for human activity recognition”, Proc. Int. Conf. on Acoustic Speech and Signal Processing, ICASSP; 2008.

[6] D.J. Cook and N.C. Krishnan, “Activity Learning: Discovering, Recognizing, and Predicting Human Behavior from Sensor Data”, John Wiley & Sons, 2015.

[7] D. J. Cook, A. S. Crandall, B. L. Thomas and N. C. Krishnan, “CASAS: A smart home in a box”. Computer, 46(7), 2013.

[8] F. Al Machot and H. C. Mayr, “Improving Human Activity Recognition by Smart Windowing and Spatio-Temporal Feature Analysis” Proc of the 9th ACM International Conf on PErvasive Technologies Related to Assistive Environments. ACM, 2016.

[9] J. Novakovic, “Using information gain attribute evaluation to classify sonar targets”, in 17th Telecommunications forum TELFOR, 2009, pp 1351–1354.

[10] S. Le Cessie, and J.C. van Houwelingen, ”Ridge estimators in logistic regression”. Applied statistics 1992, pp 191-201.

[11] I.H. Witten, E. Frank, L.E. Trigg, M.A. Hall, G. Holmes, and S.J. Cunningham, “WEKA: Practical machine learning tools and techniques with Java implementations”, 1999.