

# A Hierarchical Multi-label classification of Multi-resident activities

Hiba Mehri<sup>1</sup>, Tayeb Lemlouma<sup>1</sup>, and Nicolas Montavont<sup>2</sup>

<sup>1</sup> IRISA- University of Rennes 1, Lannion, France

<sup>2</sup> IRISA- IMT Atlantique, Rennes, France

**Abstract.** In this paper, we tackle the problem of daily activities recognition in a multi-resident e-health smart-home using a semi-supervised learning approach based on neural networks. We aim to optimize the recognition task in order to efficiently model the interaction between inhabitants who generally need assistance. Our hierarchical multi-label classification (HMC) approach provides reasoning based on real-world scenarios and a hierarchical representation of the smart space. The performance results prove the efficiency of our proposed model compared with a basic classification task of activities. Mainly, HMC highly improves the classification of interactive activities and increases the overall classification accuracy approximately from 0.627 to 0.831.

**Keywords:** Multi-resident · Recurrent neural networks · LSTM · CNN · Hierarchical multi-label classification · Activity recognition

## 1 Introduction

The rapid introduction of artificial intelligence (AI) and the Internet of Things (IoT) has been offering new opportunities to enhance the capacity of health monitoring systems and help professionals in decision-making. Some studies focus on recognizing the pattern of human activities in smart environments which is of high interest in various domains such as healthcare and elderly care. Human activities are extremely challenging when they involve many residents in the same environment. Comparing to single-resident activity recognition, multi-resident recognition is more complex and open to the interaction between activities.

In our studies, we aim to conceive a complete framework that recognizes and detects abnormal behaviour of individuals, eventually, the evolution of their health status. In this work, we focus as a first step on activity recognition in environmental data interpretation and the detection of abnormalities regarding the residents' behavior and occurrences of their activities of daily life (ADL). We propose a hybrid model based on neural networks (NN) to optimize activities' recognition in multi-resident smart homes while focusing on the interaction between residents. Single resident recognition is interesting to handle before stepping to more complex scenarios. Many approaches [4, 13] study the recognition of ADL based on recurrent neural networks (RNN). The aim is to predict future activities based on a history of previous ones. RNN classifiers trained with time

series data and specifically long short term memory (LSTM) improves the recognition accuracy [13]. In [9], smartphones accelerometer data have been used for various types of activities. The data are tested using Spiking NNs and applied on a real-world time series from the WISDM project [2] with a selection of the basic activities. The work shows the importance of the number of records fed to NNs which should be high enough to assure the stability of learning. Other recent studies in activities recognition used hidden markov models (HMM) to identify the behavior’s evolution [11] and detect abnormalities [3]. They show a lack in performance of HMM if compared to NN as they are not easily adaptable to data unless the number of hidden states is known. However, these approaches recall the coexistence of different constraints in smart homes as relationships between locations, activities and objects. Most of existing approaches still lack in prediction of some types of activities. As stated in [8], for multi-resident smart homes, it is more complex to recognize collaborative activities.

In [6], the potential of the concept of *multi-label classification* (MC) [1] is discussed in multi-resident spaces. Other specific approaches show the interest to use classifier chains along with the multi-label classification for the same purpose [7]. The study demonstrated the adaptability of multi-label classification for multi-resident problems that leads to identifying collaborative activities. However, as only basic features were included in the training and no correlation between labels is exploited, much still had to be done to optimize the activities recognition. In this work, we focus on multi-resident activity recognition to consider the users’ interactions as well as the ambiguity of complex activities. We propose a new version of the multi-label classification by exploiting the conditional dependencies between the target labels. We aim to predict activities by considering the constraints imposed by the environment. The resulting approach involves a prediction task based on different types of NNs in order to identify the most performant hybrid model for activity recognition. The remainder of this paper is organized as follows. In Section 3, we define a hybrid model based on hierarchical multi-label classification and NNs used in the recognition of ADL. In Section 4, we present the results of the implementation of our model applied to real-world datasets. Section 5 concludes the work with some perspectives.

## 2 Methodology

In this section, we present our recognition framework and the theory behind the predictive model. We propose a new approach based on a deep neural networks (DNNs) to perform multi-label classification (MC). We propose to adapt the model to smart spaces by adding a hierarchy between location, activities, and objects.

### 2.1 Proposed Model

Multi-label classification (MC) is a predictive modeling task used in machine learning. In our context, we adopt MC in smart spaces by associating sensor

events with a set of target labels. Each label represents an appropriate activity being achieved. It is worth noting that in AI, the difference in multi-label classification from basic classification problems lies in the fact that a sequence is generally associated with one specific class while with multi-labels, a given sequence can belong to different labels as different activities could be performed simultaneously. Hence, MC are well suited to multi-resident environments. Formally, we define a MC problem with  $\mathcal{MC} = (\mathcal{A}, \mathcal{X})$  where  $\mathcal{A}$  is a finite set of classes :  $\mathcal{A} = \{A_1, A_2, \dots, A_n\}$  and  $\mathcal{X}$  is a set of sensor readings  $(x, y)$ , where  $y$  is the ground truth of  $x$  (the set of true labels associated with  $x$ ). We associate a model  $m_A$  for each  $\mathcal{MC}$  problem by using a function mapping every class  $\mathcal{A}$  and every sensor event  $x$  to  $[0,1]$ . A sensor event  $x$  is predicted to belong to class  $\mathcal{A}$  whenever  $m_A(x)$  is greater than a pre-defined threshold  $\theta$ . We define  $\theta$  as the probability that the sensor event belongs to class  $\mathcal{A}$ . Besides, the MC problem is extended to be a hierarchical multi-label classification (HMC) problem by associating it with a finite set of constraints defined by  $A_1 \rightarrow A$  in the case of two-levels hierarchy architecture. Logically, the model has to predict  $\mathcal{A}$  whenever it predicts  $A_1$ . To design an HMC adapted to smart spaces, we define a selection of ADL performed in the context of smart homes that we aim to recognize. With this aim in mind, we associate each human activity and used object with a logical location to define hierarchical constraints as presented in Table 1. It should be underlined that some activities are strictly associated with specific obvious locations, while other activities may differ in other specific scenarios such as performing tasks remotely using the Internet. This type of activity can be performed anywhere in the house.

Table 1: Association of activities, objects to location

Location	Activities	Objects
Hall	Going out, other	House door, Hall
Kitchen	Preparing a meal, eating, having a snack, washing dishes	Fridge, kitchen drawer, chair
Bathroom	Toileting, having shower	Bathroom door, shower cabinet door, tap, water closet
Bedroom	Sleeping, studying, changing clothes	Bed, wardrobe
Living room	Watching TV, socializing, using internet	Couch, TV receiver, modem
Laundry	Washing clothes	Washing machine

Fig. 1 presents a hierarchical structure for a selected set of ADL activities. Three levels of hierarchy are defined, every level groups the same classes. Level 1 groups the location classes such as kitchen, bathroom, bedroom, living room, and laundry. Level 2 groups the activities associated with each location. Finally, level 3 groups the objects associated with each activity and hierarchically with each location. To predict activities (level 2) from unlabeled data, objects related to each sensor sequence are considered the ground truth. The HMC model associated them with the appropriate location (level 1), and then, it determines the activity by applying a constraint resolution module. For instance, the HMC model handles the following constraint:  $bed \rightarrow sleeping \rightarrow bedroom$  as follows. If the sensor event is associated with the *bed*, the inhabitant is supposed to be

in the bedroom. This constraint increases the probability of predicting *sleeping* as the recognized activity.

Algorithm 1 presents an overview of the implementation of the HMC model. Based on a dictionary of locations, activities, and objects extracted from real-world logical scenarios, the model determines the appropriate location of each sensor event, applies a first step of the hierarchical constraint module that resolves the constraints by connecting levels 3 and 1 of the structure. In the presented example of Algorithm 1, the model guarantees the satisfaction of the constraints with a post-processing step to enforce  $m_A(x) > \theta$  whenever  $m_{A_{11}}(x) > \theta$ . In the final step of activities prediction, the model connects hierarchy levels 2 and 1 based on the input dictionaries and enforces the  $m_{A_1}(x) > \theta$  condition to predict the appropriate activity. Note that the probability of predicting the correct activity is highly improved by the input hierarchical constraints.

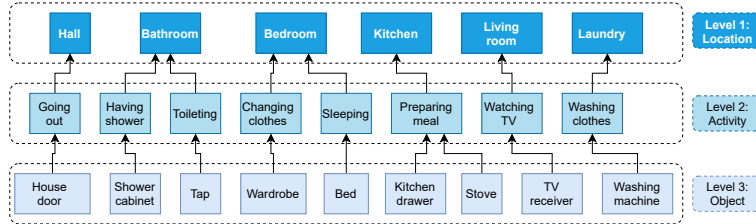


Fig. 1: Hierarchical structure of smart home

Algorithm 1 : HMC for recognition of activities

---

**Data** : Sensor events, dictionary of activities associated with  
(*location, object*)

**Result** : Prediction of activities  $A_1, A_2, \dots$

```

1 begin
2   Location = {A, B, C, D...};
3   Activities = {A1, A2, A3, B1, B2, C1...};
4   Objects = {A11, A12, A21, B11, B22, C11...};
5   for each sensor_event at time t do
6     if Sensor_event.Sensor_value = 1 then
7       Associate(Object)
8       Impose(HierarchicalConstraintModule)
9       Associate(Location)
10    end
11  end
12  Impose(HierarchicalConstraintModule)
13  if Sensor_event.Object = A11 and Sensor_event.Location = A
14    then
15      Predict(Sensor_event.Activity = A1)
16  end
  
```

---

## 2.2 Hybrid HMC Model Using Recurrent and Convolutional NNs

In our prediction approach, we use DNNs that are considered more accurate if compared to other machine learning approaches where feature engineering is essential. To improve their capacity of prediction, DNNs are able to learn the feature encoding from data within each training iteration. To benefit from this advantage, we propose to associate the HMC task to a NN module during the prediction of the associate classes while respecting the hierarchical constraints. The NN module is split into two sub-modules: one input handling the pre-processed features for the multi-label classification and another module for the hierarchical constraint module (HCM) that adjusts the attributed weights in order to satisfy

the constraints. In our reasoning on the type of NN used in the prediction module (HMC-NN), we take into consideration the dependencies that may exist between sensor events and the dependencies related to the occurrence time. In [10], the work confirmed that RNNs are suitable for time series analysis handling single and multi-resident contexts. RNNs consider more efficiently the interaction between activities and residents. Another candidate is the convolutional neural networks (CNNs) which are much more used in handling multi-label classification tasks especially for text classification [5]. Besides, they are also compatible with time-series data. Our hybrid HMC model consists of the following associations *HMC-CNN* and *HMC-LSTM*. **HMC-CNN** is the association of our HMC algorithm with a CNN for the prediction of activities. Using CNNs for time series classification brings many advantages. Mainly, they are highly resistant to noise in data and able to extract very informative and deep features along with the time dimension. They are also, very useful in capturing the correlation between variables which highly affects the accuracy of prediction. **HMC-LSTM** is the association of HMC with an LSTM that is a special kind of RNN with a learned memory state [12]. This enables a conditioned prediction even on events that happened in the past while still handling one event at a time thanks to their inherited recurrence propriety. In sum, thanks to LSTM, the model gains the ability to remember information for long periods as a default behavior. This concept is very adaptable for activity recognition as memorizing the history of past activities is important to predict future ones.

### 3 Implementation and Performance Evaluation

In this section, we experiment our approach using different metrics for the classification problem. The challenge is to test the hybrid model on real world datasets specifically adapted for multi-resident activity recognition. Our implementation of the recognition model is performed using Python with other required libraries specific to data science as Pandas and Scikit-Learn. We use the Keras framework in the implementation of NNs, for the training and testing process. Our experiments are based on the data of ARAS project <sup>3</sup>. Collected data concern two-residents smart homes with several sequences triggered along with the performed activities during 30 days. The time gap between sensor events is one second. Sensed data relate to different types of information such as force, contact, distance, temperature, IR, and photocell. The sensor sequences are pre-processed and analyzed to apply the hierarchical model and then be fed to the NN module to predict on unlabeled data. As a first step, we test the evolution of our model's accuracy using different lengths of samples: 6, 12, 20 days...etc. Proportionally to the number of samples, we observe that the accuracy does not get significantly affected by the hierarchical optimization. However, changing the type of NN notably affects the accuracy especially using the LSTM network which is known to react easily to the variety of the training data. This has a positive impact on the final prediction accuracy compared to other NNs.

<sup>3</sup> <https://www.cmpe.boun.edu.tr/aras/>

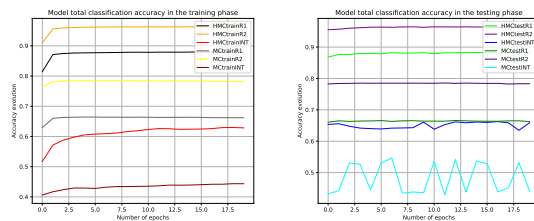
### 3.1 Evaluation Metrics and Results

For each used model, we measure the average classification accuracy for all the labels. Then, an individual evaluation of each label (activity) is performed to estimate the model’s robustness in predicting specific activities. The average classification accuracy is a common metric that estimates the proportion of true predictions among the complete data. Initial data is composed of 25% for testing and 75% for training. We focus on the testing phase to assess the classification accuracy when facing unlabeled samples. Table 2 presents the average classification accuracy of the HMC model combined with a basic neural network (BNN), CNN, and LSTM. Also, we consider the comparison with a basic task of MC to evaluate the benefits of the hierarchical representation.

Table 2: Average classification accuracy for different tested models

Label	MC			HMC		
	MC-BNN	MC-CNN	MC-LSTM	HMC-BNN	HMC-CNN	HMC-LSTM
Resident 1	0.661	0.646	0.666	0.88	0.882	<b>0.885</b>
Resident 2	0.78	0.775	0.785	0.9	<b>0.964</b>	0.95
Interaction	0.49	0.42	0.431	0.57	0.63	<b>0.66</b>

The initial results show that HMC outperforms the MC task independently of the used NN and the type of recognition. For LSTM, the classification accuracy evolves from an average accuracy of 0.666 to 0.885 for resident 1, from 0.785 to 0.95 for resident 2, and from 0.431 to 0.66 for the recognition of interactive activities. On another hand, we observe that the classification accuracy is highly impacted by the variety of activities between the two residents. Indeed, the model provides better accuracy for resident 2. Our experimentations show that HMC-CNN proves a high performance in detecting individual activities of each resident. For all the models, it seems more difficult to detect the interaction, especially for the basic classifier. The results of HMC-LSTM are the more accurate with a mean prediction accuracy of 0.831.



(a) Training phase (b) Testing phase

Fig. 2: Accuracy of HMC-LSTM and MC-LSTM

In Fig. 2, we compare the training and testing accuracy between HMC-LSTM and MC-LSTM. We evaluate the accuracy by considering the activities of residents 1 and 2 and their interactions. For all the situations, we observe that HMC outperforms MC. This is particularly true during the testing phase where it reaches its best accuracy for resident 2 with a value of 0.96 while reaching 0.66 for recognizing the interactive activities. Drawing on this observation of the HMC-LSTM model, we focus on the evaluation of its classification by common metrics precision, recall, and F1-score. Precision-Recall is the measure of

prediction success while having very unbalanced classes. The precision is the measure of result relevancy, while recall measures the number of truly relevant results of the model. Finally, the F1-score is calculated by the following equation:  $F_1 score = \frac{Precision \times Recall}{Precision + Recall}$ . Table 3 provides the results for each activity.

Table 3: Per-label classification of activities

Label	Precision	Recall	F1-score	Label	Precision	Recall	F1-score
Going out	0.99	0.99	0.99	Having shower	0.85	0.8	0.82
Preparing meal	0.79	0.87	0.83	Toileting	0.85	0.78	0.82
Having a meal	0.84	0.92	0.88	Laundry	0.99	0.99	0.99
Washing dishes	0.82	0.39	0.52	Using internet	0.63	0.66	0.65
Having snack	0.36	0.83	0.5	Washing routine	0.47	0.79	0.59
Sleeping	0.99	0.99	0.99	Socializing	0.51	0.66	0.58
Watching TV	0.99	0.99	0.99	Changing clothes	0.39	0.68	0.54

According to the initial results of the F1-score, we observe that the model is highly affected by the variety of activities in the real-world dataset where it is impossible to control the scenarios of daily living in order to improve the accuracy of prediction. The frequency and duration of each activity affect highly the samples injected for the training. For some activities, such as going out, sleeping, watching TV, or having a meal, the model is more accurate because the capacity of learning is more important for such frequent and time-fixed daily activities. For other activities, that are less associated with a specific interval of time, the model lacks accuracy while interpreting activities due to the absence of routines. In order to evaluate the consistency of our model, we consider the evolution of accuracy in terms of iterations and the cross-entropy loss estimation.

**Evolution of accuracy in terms of number of training iterations:** we propose to study the evolution of training and compare it with the testing accuracy to evaluate an over-fitting possibility affecting the model and determine the optimal number of training iterations. Fig. 3a, 3b, 3c present the evolution of accuracy of HMC-LSTM for the activities of residents 1 and 2 and their interaction. We observe that the stabilization of the accuracy is similar during the training and testing for residents 1 and 2. However, for the residents' interaction, the training process takes longer to stabilize in approximately the 10<sup>th</sup> iteration. The training process lacks stability as there are not many interactive activities in the dataset. Moreover, we observe that the model is not committing over-fitting as the training and testing accuracy are synchronized. Such situations are detected if the training accuracy is significantly better than the testing. This means integrally that the model is perfectly memorizing the training dataset without being able to recognize new injected samples in the testing phase. Concerning the optimal number of iterations, the model is reaching its peak of performance with the selected parameters at the 20<sup>th</sup> training iteration for all the cases.

**Cross-entropy loss function estimation :** we use this function to measure the performance of the classification when the output is a probability. The loss increases when the predicted probability deviates from the true label (i.e. the right activity that should be predicted). Cross-entropy loss is interesting for possible optimizations. It is calculated by the following equation:  $Cross\_entropy\_loss(\mathbf{y}, \hat{\mathbf{y}}) = - \sum_{i=1}^N y_i \log(\hat{y}_i)$  where  $\mathbf{y}$  is a one-hot label vector and N is the number of classes (14 activities in our case). As a perfect model has a cross-entropy loss of 0, we aim to minimize this loss. In Fig. 3d 3e 3f, for all the

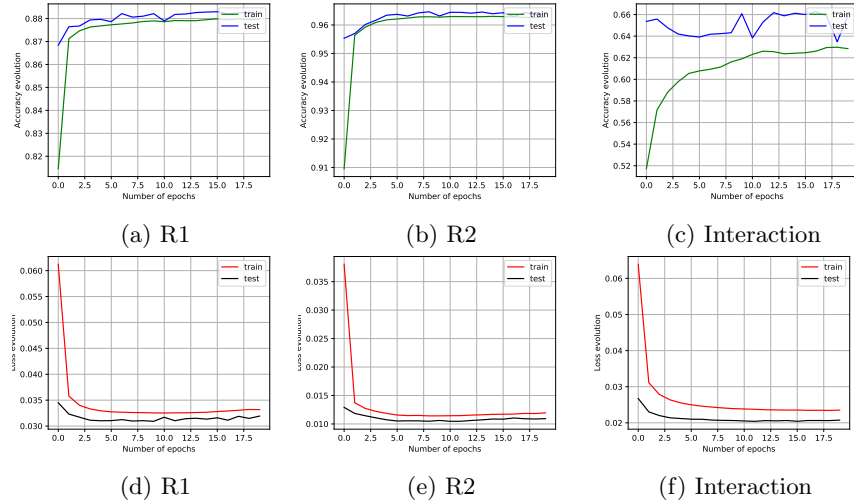


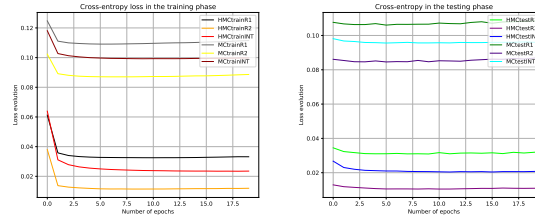
Fig. 3: Accuracy and cross-entropy loss of HMC-LSTM

three classes, resident 1, resident 2, and interactive activities, the loss function decreases when the number of iterations increases until attending the final values that are in an interval of  $[0.01, 0.035]$ . We observe that the model provides a high prediction performance which minimizes the loss function. However, it still an object of optimization as it is possible to adjust the model weights during the training. the aim is to resolve the problem of unbalanced data and represent better the classes that are less present in the tested dataset. Fig. 4 presents the evolution of loss function for the HMC and MC models for the testing and training phase. As we can observe, the HMC loss decreases to reach values under 0.04 for the two phases while the MC loss varies between 0.08 and 0.12. This result confirms the good prediction performance of the hierarchical proposed model.

## 4 Conclusion

In this work, we focused in developing an activities' recognition model for multi-resident in smart environments. We proposed HMC: a hierarchical multi-label classification model that satisfies existing constraints between different elements of the environment such as location, activities, and objects. We evaluated our model with different types of NNs and along with a simple task of multi-label classification and a hierarchical one on a real-world dataset. The results reveal that the proposed HMC-LSTM model provides the best prediction accuracy while minimizing the loss function. For instance, HMC-LSTM model outperforms MC-LSTM by increasing the overall accuracy from 0.627 to 0.831 while optimizing the recognition of interactive activities. However, the model still lacking accuracy in determining interactive activities if compared to individual ones. In the future, we will explore the performance of our model on other real-world data in order to assure its generalizability. Other techniques of optimization may be applied such as a manual weighing methodology of features that could enable the consideration of activities that are not much represented in datasets.





(a) Training phase

(b) Testing phase

Fig. 4: Cross-entropy loss of HMC-LSTM and MC-LSTM

## References

- Giunchiglia, E., Lukasiewicz, T.: Multi-label classification neural networks with hard logical constraints. *Clinical Orthopaedics and Related Research* **479** (2021)
- Kwapisz, J.R., Weiss, G.M., Moore, S.A.: Activity recognition using cell phone accelerometers. *ACM SigKDD Explorations Newsletter* **12**(2), 74–82 (2011)
- Liouane, Z., Lemlouma, T., Roose, P., Weis, F., Hassani, M.: A markovian-based approach for daily living activities recognition. In: *2016 5th International Conference on Sensor Networks* (2016)
- Natani, A., Sharma, A., Peruma, T., Sukhavasi, S.: Deep learning for multi-resident activity recognition in ambient sensing smart homes. In: *2019 IEEE 8th Global conference on consumer electronics*. pp. 340–341. IEEE (2019)
- Qiu, M., Zhang, Y., Ma, T., Wu, Q., Jin, F.: Convolutional-neural-network-based multilabel text classification for automatic discrimination of legal documents. *Sensors and Materials* **32**(8), 2659–2672 (2020)
- Raihani, M., Thinagaran, P., Md Nasir, S., Norwati, M.: Multi resident complex activity recognition in smart home: a literature review. *Int. J. Smart Home* **11**(6), 21–32 (2017)
- Raihani, M., Thinagaran, P., Md Nasir, S., Norwati, M., MN shah, Z.: Multi label classification on multi resident in smart home using classifier chains. *Advanced Science Letters* **24**(2), 1316–1319 (2018)
- Riboni, D., Murru, F.: Unsupervised recognition of multi-resident activities in smart-homes. *IEEE Access* **8**, 201985–201994 (2020)
- Suriani, N.S., Rashid, F.N.: Smartphone sensor accelerometer data for human activity recognition using spiking neural network. *Int. J. Machine Learn. Comput* **11**, 298–303 (2021)
- Thapa, K., Al, A., Md, Z., Lamichhane, B., Yang, S.H.: A deep machine learning method for concurrent and interleaved human activity recognition. *Sensors* **20**(20), 5770 (2020)
- Tran, S.N., Zhang, Q., Smallbon, V., Karunanithi, M.: Multi-resident activity monitoring in smart homes: A case study. In: *2018 IEEE International Conference on Pervasive Computing and Communications Workshops*. pp. 698–703. IEEE (2018)
- Xia, K., Huang, J., Wang, H.: Lstm-cnn architecture for human activity recognition. *IEEE Access* **8**, 56855–56866 (2020)
- Zehtabian, S., Khodadadeh, S., Bölöni, L., Turgut, D.: Privacy-preserving learning of human activity predictors in smart environments. In: *2021 IEEE 40th International Conference on Computer Communications*. IEEE (2021)