

# Comparing HMM, LDA, SVM and Smote-SVM Algorithms in Classifying Human Activities

M'hamed Bilal Abidine and Belkacem Fergani

**Abstract** Accurately recognizing the rare activities from sensor network based smart homes for monitoring the elderly person is a challenging task. Typically a probabilistic models such as the Hidden Markov Model (HMM) and Linear Discriminant Analysis (LDA) are used to classify the activities. In this work, we demonstrate that discriminative model named Support Vector Machines (SVM) based on the Synthetic Minority Over-sampling Technique (Smote) outperforms HMM, LDA and standard SVM and it can lead to a significant increase in recognition performance. Our experiments carried out on multiple real world activity recognition datasets, consisting of several weeks of data.

**Keywords** Activity recognition · HMM · LDA · SVM · Imbalanced classification

## 1 Introduction

As the number of elderly people in our society increases and the households will include someone who needs help performing basic activities of daily living such as cooking, dressing, toileting, bathing and so on [1, 2]. For their comfort and because the healthcare infrastructure will not be able to handle this growth, it is suggested to assist sick or elderly people at home. Sensor based technologies in the home is the key of this problem. Sensor data collected often needs to be analysed using data mining and machine learning techniques [3] to determine which activities took place. State of the Art methods used for recognizing activities can be divided in two main categories: generative models and discriminative models [3].

---

M.B. Abidine (✉) · B. Fergani  
Laboratoire d'Ingénierie des Systèmes Intelligents et Communicants,  
Faculty of Electronics and Computer Sciences, University of Science and Technology  
Houari Boumediene (USTHB), Algiers, Algeria  
e-mail: abidineb@hotmail.com

B. Fergani  
e-mail: bfergani@gmail.com

However, activity recognition datasets are generally imbalanced, meaning certain activities occur more frequently than others. However, the learning system may have difficulties to learn the concept related to the minority class. Many popular machine learning algorithms have been tried to see how well they can cope with the imbalanced situation [4], e.g. Weighted Support Vector Machine (WSVM) [5],  $k$ -Nearest Neighbors  $k$ -NN [5], random forests [6] and CS-SVM [7].

The main contribution of our work is twofold. Firstly, we demonstrate the efficiency of the standard discriminative method named Support Vector Machines (SVM) [3] combined with the Synthetic Minority Over-sampling Technique [8] in order to avoid the overfitting caused by imbalanced activity samples in smart homes. Secondly, this method is compared with the standard SVM, Linear Discriminant Analysis (LDA) [9] and Hidden Markov Model (HMM) [2].

## 2 Discriminative Models for Activity Recognition

### 2.1 Linear Discriminant Analysis (LDA)

Given a set of observations in  $n$ -dimensional space:  $D_i = \{x_1^i, \dots, x_{m_i}^i\}$  ( $x_j^i \in R^n$ ) from class  $C_i$  ( $i = 1, \dots, N$ ,  $N$  is the number of classes), we assume that each of the class probability density functions can be modeled as normal distribution. Define the prior probabilities  $p(C_i)$ , means  $\bar{m}_i$ , and covariance matrices  $\Sigma_i$  of each class:

$$\Sigma_i = \frac{1}{m_i} \sum_{j=1}^{m_i} (x_j^i - \bar{m})(x_j^i - \bar{m})^T \quad (1)$$

where  $m_i$  is the number of patterns in class  $C_i$ . With LDA all classes are assumed to have the same covariance matrices  $\Sigma_1, \dots, \Sigma_N$ , on (1). We assign the new feature vector that is to be classified  $x$  to  $C_i$  with the linear discriminant function  $d_i$ . This function is obtained by simplification the quadratic discriminant rule [9]

$$d_i(x) = \log(p(C_i)) - \frac{1}{2} m_i^T S_W^{-1} \bar{m}_i + x^T S_W^{-1} \bar{m}_i \quad (2)$$

in which  $S_W$  is the common covariance matrix

$$S_W = \sum_{i=1}^N \frac{m_i}{m - N} \Sigma_i \quad (3)$$

The classification rule is given in Eq. 4.

$$f(x) = i^* : \Leftrightarrow i^* = \arg \max_i d_i(x) \tag{4}$$

## 2.2 Proposed Approach for Activity Recognition (Smote-SVM)

Smote-SVM approach is shown in Fig. 1. In the training phase, we perform the necessary pre-processing on the activity data represented in a feature space. We need only to correct the class imbalance using the pre-classification named Smote strategy. The balanced data is then used to learn the SVM classifier. It will then be used to process a new observation during the testing phase where the associated ADL class will be predicted.

### a. The Synthetic Minority Over-sampling Technique (SMOTE)

The SMOTE algorithm generates artificial data based on the feature space similarities between existing minority examples in the training set. Synthetic examples are introduced along the line segment between each minority class example and one of its  $k$  minority class nearest neighbors. The  $k$ -nearest neighbors ( $k$ -NN) are defined as the  $k$  elements of subset  $S_{\min} \in S$  whose Euclidian distance between itself and  $x_i \in S_{\min}$  under consideration exhibits the smallest magnitude along the  $n$ -dimensions of feature space  $X$ . To create a synthetic sample, the  $k$ -nearest neighbors are randomly chosen, then multiply the corresponding feature vector difference with a random number  $\delta \in [0, 1]$ , and finally, add it to  $x_i$

$$x_{new} = x_i + (\hat{x}_i - x_i) \times \delta, \tag{5}$$

where  $x_i \in S_{\min}$  is the minority instance under consideration,  $\hat{x}_i$  is one of the  $k$ -NN for  $x_i$ ;  $\hat{x}_i \in S_{\min}$ .

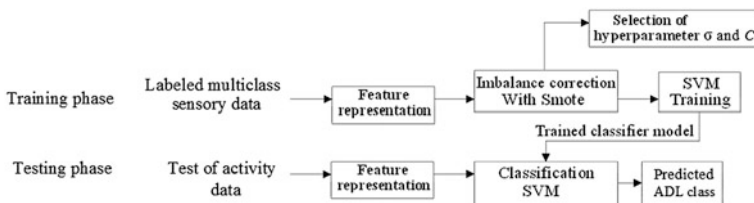


Fig. 1 Diagram of Smote-SVM approach

### b. Support Vector Machines (SVM)

We assume that we have a training set  $\{(x_i, y_i)\}_{i=1}^m$  where  $x_i \in R^n$  are the observations and  $y_i$  are class labels either 1 or  $-1$ . The dual formulation of the SVM can be solved by representing it as a Lagrangian optimization problem as follows [3]:

$$\begin{aligned} \max_{\alpha_i} \quad & \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \alpha_i \alpha_j y_i y_j K(x_i, x_j) \\ \text{Subject to} \quad & \sum_{i=1}^m \alpha_i y_i = 0 \text{ and } 0 \leq \alpha_i \leq C \end{aligned} \quad (6)$$

where  $K(x_i, x_j)$  is the kernel, the radial basis kernel function (RBF) is used in the study:  $K(x, y) = \exp\left(\frac{-1}{2\sigma^2} \|x_i - x_j\|^2\right)$ .  $\alpha_i > 0$  are Lagrange multipliers. The regularization parameter  $C$  is used to control the trade-off between maximization of the margin width and minimizing the number of training error.

Solving (6) for  $\alpha$  gives a decision function in the original space for classifying a test point  $x \in R^n$  [3]

$$f(x) = \text{sgn}\left(\sum_{i=1}^{m_{sv}} \alpha_i y_i K(x, x_i) + b\right) \quad (7)$$

with  $m_{sv}$  is the number of support vectors  $x_i \in R^n$ .

In this study, a software package LIBSVM [10] was used to implement the multiclass classifier algorithm. It uses the one-versus-one method [3].

## 3 Experimental Results

We use an openly datasets [11] gathered from three houses KasterenA, KasterenB, KasterenC, having different layouts and different number of sensors, thus providing a diverse testbed. The activities performed using a wireless sensor network with a single man occupant. Data are collected using binary sensors, such as reed switches and float sensors. The sensor data were labelled using different annotation methods using Bluetooth headset or Handwritten diary. We separate the data into a test and training set using a ‘‘Leave one day out cross validation’’ approach [2].

As the activity instances were imbalanced between classes, we evaluate the performance of our models by two measures, the accuracy and the class accuracy. The accuracy shows the percentage of correctly classified instances, the class

accuracy shows the average percentage of correctly classified instances per classes. They are defined as follows:

$$Accuracy = \frac{\sum_{i=1}^m [\text{inf } erred(i) = \text{true}(i)]}{m} \tag{8}$$

$$Class = \frac{1}{N} \sum_{c=1}^N \left[ \frac{\sum_{i=1}^m [\text{inf } erred_c(i) = \text{true}_c(i)]}{m_c} \right] \tag{9}$$

in which [a = b] is a binary indicator giving 1 when true and 0 when false.  $m$  is the total number of samples,  $N$  is the number of classes and  $m_c$  the total number of samples for class  $c$ . A problem with the accuracy measure is that it does not take differences in the frequency of activities into account. Therefore, the class accuracy should be the primary way to evaluate an activity classifier’s performance.

In our experiments, for the Smote-SVM method, the minority class examples were over-sampled using  $k = 4$  nearest neighbors for Smote. We utilize the leave-one-day-out cross validation technique for the selection of width parameter for the SVM classifier. We found  $\sigma_{opt} = 1$ ,  $\sigma_{opt} = 1$  and  $\sigma_{opt} = 2$  for these datasets respectively. The summary of the accuracy and the class accuracy obtained, for HMM, LDA, SVM and Smote-SVM methods performed using various real world datasets are shown in Table 1. This table shows that Smote-SVM performs better in terms of class accuracy.

Our results give us early experimental evidence that Smote-SVM works better for model classification; it consistently outperforms the other methods in terms of the class accuracy for all datasets. In the rest of section, we explain the difference in terms of performance between HMM and our method. HMM is trained by splitting the training data in which a separate model  $P(x|y)$  is learned for each class, as

**Table 1** Class accuracy and Accuracy for HMM, LDA, SVM and Smote-SVM

Houses	Models	Class (%)	Accuracy (%)
KasterenA	HMM [11]	66	92
	LDA	60.9	89.8
	Standard SVM	50.3	92.1
	Smote-SVM	75	88.6
KasterenB	HMM [11]	55	80
	LDA	48.7	80.6
	Standard SVM	39.3	85.5
	Smote-SVM	67.3	64.6
KasterenC	HMM [11]	40	76
	LDA	40.1	79.7
	Standard SVM	35.6	80.7
	Smote-SVM	56.6	78.3

parameters are learned for each class separately. This is why HMM performs better for the minority activities. Our method shows that SVM becomes more robust for classifying the minority class.

## 4 Conclusion and Perspectives

Our experiments on real world datasets show that the choice of Smote-SVM approach can significantly increase the recognition performance to classify multi-class sensory data, and are less prone to overfitting caused by imbalanced datasets. It significantly outperforms HMM, LDA and SVM. Developing Classifiers which are robust and skew insensitive or hybrid algorithms can be point of interest for the future research in activity recognition. It would be interesting to compare Smote-SVM and Smote-CS-SVM [7] and then deciding which gives the best results.

## References

1. Abidine, M.B., Fergani, L., Fergani, B., Fleury, A.: Improving human activity recognition in smart homes. *Int. J. E-Health Med. Commun. (IJEHMC)* **6**(3), 19–37 (2015)
2. Van Kasteren, T., Noulas, A., Englebienne, G., Kröse, B.: Accurate activity recognition in a home setting. In: *Proceedings of UbiComp'08*, pp. 1–9. ACM, New York, USA (2008)
3. Vapnik, V.N.: *The Nature of Statistical Learning Theory*. Springer, New York (2000)
4. Chawla, N.V.: Data mining for imbalanced datasets: an overview. In: *Data Mining and Knowledge Discovery Handbook*, (pp. 875–886). Springer, US (2010)
5. Abidine, M.B., Fergani, B.: A new multi-class WSVM classification to imbalanced human activity dataset. *J. Comput.* **9**(7), 1560–1565 (2014)
6. Chen C., Liaw, A., Breiman, L.: Using random forest to learn unbalanced data. Technical Report 666, Statistics Department, University of California at Berkeley (2004)
7. Abidine, M.B., Fergani, B., Oussalah, M., Fergani, L.: A new classification strategy for human activity recognition using cost sensitive support vector machines for imbalanced data. *J. Kybernetes* **43**(8), 1150–1164 (2014)
8. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **16**, 321–357 (2002)
9. Croux, C., Filzmoser, P., Joossens, K.: Classification efficiencies for robust linear discriminant analysis. *Statistica Sinica* **18**(2), 581–599 (2008)
10. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. *ACM Trans. Intell. Syst. Technol.* **2**:1–27 (2013). <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>
11. Van Kasteren, T., et al.: Effective performance metrics for evaluating activity recognition methods. In: *Proceedings of ARCS 2011 Workshop on Context-Systems Design, Evaluation and Optimisation*, pp. 22–23. Italy (2011)