Expert Systems With Applications 74 (2017) 19–28

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Learning general model for activity recognition with limited labelled data

Jiahui Wen[a,b,*], Zhiying Wang[a,b]

[a] State Key Laboratory of High Performance Computing National University of Defense Technology, Changsha 410073, China
[b] School of Computer, National University of Defense Technology, Changsha, China

ARTICLE INFO

Article history:
Received 21 July 2015
Revised 2 January 2017
Accepted 3 January 2017
Available online 6 January 2017

Keywords:
Activity recognition
General model
Co-training

ABSTRACT

Activity recognition has been a hot topic for decades, from the scientific research to the development of off-the-shelf commercial products. Since people perform the activities differently, to avoid overfitting, building a general model with activity data of various users is required before the deployment for personal use. However, annotating a large amount of activity data is expensive and time-consuming. In this paper, we build a general model for activity recognition with a limited amount of labelled data. We combine Latent Dirichlet Allocation (LDA) and AdaBoost to jointly train a general activity model with partially labelled data. After that, when AdaBoost is used for online prediction, we combine it with graphical models (such as HMM and CRF) to exploit the temporal information in human activities to smooth out the accidental misclassifications. Experiments with publicly available datasets show that we are able to obtain the accuracy of more than 90% with 1% labelled data.

1. Introduction

Activity recognition (Banos, Damas, Pomares, Prieto, & Rojas, 2012; Chernbumroong, Cang, Atkins, & Yu, 2013; Huang, Lee, Kuo, & Lee, 2010; Ordóñez, Iglesias, De Toledo, Ledezma, & Sanchis, 2013; Wen, Loke, Indulska, & Zhong, 2015a) has gained much attention during the past ten years, both scientifically and commercially, because of the widely used applications in everyday life. For example, recognising human lifestyle can help to evaluate energy expenditure (Albinali, Intille, Haskell, & Rosenberger, 2010), and walking detecting and step counting can help to monitor elderly health (Brajdic & Harle, 2013). Recently, numbers of commercial products have been released for personal purpose, such as NIKE SPORTWATCH[1], MI miband[2] and APPLE iWatch[3]. In 2013, Google announced its Android activity recognition API[4], with which people can easily develop applications detecting activities such as *Stationary, On Foot, Cycling* and *In Vehicle*.

In order to provide a robust recognition system, it is necessary to train a general model with activity data from various users. The underlying reason is that people may perform the activities differently due to their age, gender and other physical characteristics. Therefore, activity models trained for a specific user may overfit the activity data of a group of users that present the similar patterns, and can not be scaled to others. A general model is extremely important for commercial products. Since they are users-oriented, low recognition accuracy can negatively affect the user experience and decrease the profit margin. However, annotating huge amount of activity data to build a general model is expensive, time-consuming and error-prone. Therefore, this paper aims to build a general activity model with limited labelled data and unlimited unlabelled data while maintaining a satisfactory accuracy.

Human activity recognition is a hot topic in pervasive computing community, and it has been addressed by numerous previous work with different sensing modalities and learning methods. For example, Zhan, Faux, and Ramos (2014) propose hierarchical classifiers to recognition daily activities with camera and accelerometer. In the low level, LogitBoost and Support Vector Machine (SVM) are used to recognise local vision and motion features. While in the high level, Conditional Random Fields (CRFs) are leveraged to exploit the temporal information in the human behaviours and smooth out the outliers. They find that video features are more accurate for stationary activities (e.g. Watching TV) and acceleration generally has good accuracy on locomotive activities (e.g. running). In Khalifa, Hassan, and Seneviratne (2015), the authors show that the motion (kinetic) energy harvested when the people is performing the activities can be used to classify human activities. The

* Corresponding author.
E-mail addresses: j.wen@uq.edu.au (J. Wen), zywang@nudt.edu.cn (Z. Wang).

1 https://secure-nikeplus.nike.com/plus/products/sport_watch/.
2 http://www.mi.com/tw/miband/.
3 http://www.apple.com/au/watch/.
4 http://developer.android.com/google/play-services/location.html.

http://dx.doi.org/10.1016/j.eswa.2017.01.002

basic idea is that different activities produce kinetic energy in a different way leaving their signatures in the harvested power signal. Therefore, they are able to recognise activities without external sensing devices such as accelerometer. In Cheng, Griss, Davis, Li, and You (2013a); Cheng et al. (2013b), the authors propose zero-shot learning to recognise unseen activities. They introduce middle-level semantic attributes to relate low-level sensor readings and high-level activities. In this way, low-level observations can be classified into middle-level semantic attributes, and unseen activities can be recognised by exploiting the relationships between the semantic attributes and the unseen activities with commonsense knowledge and domain knowledge. Sundholm, Cheng, Zhou, Sethi, and Lukowicz (2014) propose to integrate a cheap, simple textile pressure sensor matrix into exercise mats to recognise and count exercise activities which are difficult to recognise with a single body worn motions sensors. Cao, Nguyen, Phua, Krishnaswamy, and Li (2012) develop a novel framework that contains simple pre- and post- classification strategies to improve the overall performance. They address the problem of class imbalance with structure preserving oversampling and exploit the sequential nature of sensor data with smoothing and classifier fusion. In Keally, Zhou, Xing, Wu, and Pyles (2011), the authors recognise activity with smartphone-based body sensor networks. They perform retraining detection by analysing the K-L divergence of the sensor data and sensor selection by identifying sensing redundancies with decision correlations among sensors. In this way, they are able to improve classification efficiency and reduce reliance on user annotated ground truth. There are also other works that address the energy efficiency by selecting a subset of the sensors dynamically (Gordon, Czerny, Miyaki, & Beigl, 2012; Keally et al., 2011; Zappi et al., 2008) or changing the sampling rate of the sensors adaptively (Yan, Subbaraju, Chakraborty, Misra, & Aberer, 2012). Other works (Huynh, Fritz, & Schiele, 2008; Seitr, Chiu, Fritz, Amft, & Troster, 2015; Sun, Yeh, Cheng, Kuo, & Griss, 2014) even use unsupervised methods to discover frequent patterns from human daily lives with accelerometers.

However, the aforementioned methods for activity recognition are based on personal activity data, disregarding the fact that different people may have different ways to perform the activities due to their age, gender and physical characteristic. Therefore, the activity model trained on one person cannot be scaled to others who present different activity patterns. In this paper, we will show the variations that people perform the activities with publicly available datasets, and build a general activity model with activity data of various people. As data labelling is expensive and time-consuming, we learn the general model with limited labelled data. Specifically, We build initial model with AdaBoost by using the limited labelled data, and combine LDA with AdaBoost to iteratively re-estimate the posterior distribution for each example. LDA is known to be effective in collaborative learning. In our case, each user has different mixtures of activity classes, while the parameters of each activity class is globally shared among the users. In this way, sensor data from all the users are collaboratively combined to overcome the problem of label sparsity. However, as LDA cannot be applied directly to the activity data, we combine it with AdaBoost to perform collaborative learning. AdaBoost resulted from the learning process is the general activity model. Finally, when AdaBoost is deployed for prediction, we combine it with graphical models (such as HMM and CRF) to smooth out the outliers.

To conclude, we propose an activity recognition method that is able to achieve high recognition accuracy with less annotated data. It is able to deal with variants of activity patterns because it trains the activity model with data of different users collaboratively, and it can exploit the temporal information to improve recognition accuracy. The propose method is related to expert and intelligent computing area since it requires less human effort for labelling the data while maintain high activity recognition accuracy. The contributions of this paper include:

1. We demonstrate that people perform the activities differently, and activity model built on one person cannot be scaled to others that have different activity patterns.
2. By combining AdaBoost and LDA, we propose a method to build a general activity model with limited labelled data and unlimited unlabelled data;
3. We propose a novel way of combining AdaBoost with HMM and CRF to exploit the temporal characteristics of human behaviour, so as to smooth out the outliers during the online prediction;
4. We demonstrate the effectiveness of the proposed methods with publicly available datasets, and analysis its effectiveness through comprehensive experimental and comparison studies.

2. Related work

Generally, the models for recognizing human activities can be classified into two categories: knowledge-driven models (Azkune, Almeida, López-de Ipiña, & Chen, 2015; Chen, Nugent, & Wang, 2012b; Fernández-Caballero, Castillo, & Rodríguez-Sánchez, 2012; Riboni, Pareschi, Radaelli, & Bettini, 2011; Wen, Indulska, & Zhong, 2016) and data-driven models (de la Concepción, Morillo, Gonzalez-Abril, & Ramírez, 2014; Kwon, Kang, & Bae, 2014; Wen & Indulska, 2015; Wen & Wang, 2016; Wen & Zhong, 2015; Wen, Zhong, & Wang, 2015c). In knowledge-driven models, the activities are usually represented in the form of rules specified with common sense, and the models have an advantage in being reused among different environments. However, the limitation of the statically and strictly defined rules makes the models being unable to deal with noises and uncertain information in sensor readings (Gu, Chen, Tao, & Lu, 2010). By contrast, data-driven models, which are trained with realistic data, are more powerful when facing the characteristics of randomness and erratic nature of human behaviours. To name a few, they include Naive Bayesian used in Bao and Intille (2004); Tapia, Intille, and Larson (2004), HMM used in Patterson, Fox, Kautz, and Philipose (2005); Van Kasteren, Noulas, Englebienne, and Kröse (2008), Support Vector Machine(SVM) in Brdiczka, Crowley, and Reignier (2009); Cook, Krishnan, and Rashidi (2013); Zhan et al. (2014), Decision Trees in Bao and Intille (2004); Hevesi, Wille, Pirkl, Wehn, and Lukowicz (2014), KNN in Hevesi et al. (2014); Sundholm et al. (2014) and Conditional Random Fields(CRF) in Vail, Veloso, and Lafferty (2007); Zhan et al. (2014). The reader is referred to survey (Chen, Hoey, Nugent, Cook, & Yu, 2012a; Ye, Dobson, & McKeever, 2012) for more details.

Since we need to train a general activity model with labelled and unlabelled data, our work is related to traditional semi-supervised activity recognition, in which the examples classified with high confidence are used to retrain and refine the model. For example, Stikic, Larlus, and Schiele (2009); Stikic and Schiele (2009) propose multi-graph based label propagation and multi-instance learning to iteratively model activities from both labelled and unlabelled data. Lee and Cho (2014) propose a mixture-of-experts co-trained model for activity recognition with label and unlabelled data. In their model, the global model and mixture-of-expert model iteratively select the instances that they are confident with and add them to each other's training data. However, these methods are proposed based on personalized sensor data, and are not applicable for multiple users who have significantly different activity patterns. Another potential problem with semi-supervised methods is that, for example, even though many labels have comparable likelihood in a step, it only considers the most confident one and ignore the others

Table 1
Related works and their shortcomings.

Methods	Representative works	Shortcomings
Supervised activity learning	Bao and Intille (2004); Tapia et al. (2004); Patterson et al. (2005); Van Kasteren et al. (2008); Cook et al. (2013); Brdiczka et al. (2009); Zhan et al. (2014); Hevesi et al. (2014); Sundholm et al. (2014); Vail et al. (2007)	Require large amount of labelled data
Semi-supervised activity learning	Stikic et al. (2009); Stikic and Schiele (2009); Lee and Cho (2014)	Limited to personalized dataset
Activity personalization	Zhao et al. (2011); Reiss and Stricker (2013)	Limited to personalized dataset
Unsupervised activity learning	Zheng and Yang (2011); Maekawa and Watanabe (2011); Riboni et al. (2011); Chen et al. (2012b); Azkune et al. (2015); Fernández-Caballero et al. (2012); Gu et al. (2010)	Low accuracy, need to define rules

(Wang, Pentney, Popescu, Choudhury, & Philipose, 2007). We use the traditional semi-supervised method as baseline for building general model in our evaluation, and analysis the potential problems when using it to recognise activities of multiple users.

Personalization of activity model also leverages the labelled and unlabelled data to adapt the model. In Zhao, Chen, Liu, Shen, and Liu (2011), the authors classify the examples with pre-trained decision tree, and then perform K-means to cluster the examples. The model is adapted by re-estimating its parameters (thresholds in branch nodes) via the clustered examples. In Reiss and Stricker (2013), the general model is a set of classifiers from different users, and the personalization is performed by estimating the weight of the classifiers using labelled data from a previously unknown user. In the testing phase, the probability that a classifier is chosen for prediction is proportional to its weight. As we can see, even though activity personalization makes use of both labelled and unlabelled data, the prerequisite is to build a general model that will be adapted to a specific user. While our method is to train a general model with limited labelled data and unlimited unlabelled data, thus activity personalization and our method fall into different levels and can complement each other.

There are also solutions using unsupervised methods (Maekawa & Watanabe, 2011; Zheng & Yang, 2011) to build a model for a specific user with data collected from other users that present similar physical characteristics. However, the activity data of others needs to be annotated. Moreover, there is no significant difference between selecting users based on their physical similarity and random selection, since the physical characteristics and the activity patterns may not have correlation (Reiss & Stricker, 2013). The summary of the related works and their drawbacks are presented in Table 1.

3. Latent Dirichlet allocation

In this section, we briefly introduce LDA and analyse why it cannot be applied directly to activity data. LDA is a hierarchical bayesian model, being primarily used for text mining. In LDA, the document is modelled as a multinomial distribution over the latent topics, while the latent topic is modelled as a multinomial distribution over the words. LDA explores the documents and clusters the frequent co-occurrence words into the same latent topic. The graphical representation of LDA is depicted in Figure.1, where T is the pre-specified number of topics, and N_d is the number of words in document d. To generate a word, the topic distribution of the corresponding document is sampled from a prior Dirichlet distribution parametrized by α, $\theta_d \sim Dir(\alpha)$. And then the topic assignment z_i of the word is drawn from a multinomial distribution $z_i \sim Multi(\theta_d)$, and the word is generated by sampling $w_i \sim multi(\phi_{z_i})$. Notice that ϕ_{z_i} specifies the word distribution of topic z_i, which is draw from a prior Dirichlet distribution parametrized by β. Therefore, the likelihood of the words in the corpus is:

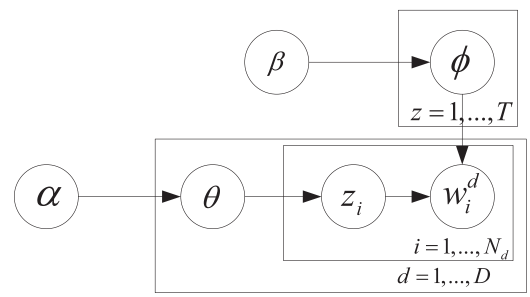

Fig. 1. Graphical model of LDA.

$$\mathcal{L}(\alpha, \beta) = \prod_{d=1}^{D} \int \int p(\phi|\beta) p(\theta|\alpha) \prod_{i=1}^{N_d} \sum_{z=1}^{T} p(w_i^d|z, \phi) p(z|\theta) d\theta d\phi \tag{1}$$

Building the topic model is equivalent to finding the topic-word assignments that can maximise the likelihood. The assignment can be estimated via collapsed Gibbs samples, which iteratively samples the topic for each word while fix the topic assignment of all the others, and then uses the topic assignments to estimate parameters (such as document-topic distribution and topic-word distribution):

$$P(z_i^d = k|\boldsymbol{z}_{-i}, \boldsymbol{w}) \propto (\alpha + n_k^{d,-i}) p(w_i^d|\boldsymbol{w}_{k,-i}) \tag{2}$$

where $\boldsymbol{z}_{-i}$ is the topic assignments in the previous iteration excluding the current word w_i^d, and $n_k^{d,-i}$ is the number of all the other words that are assigned to topic k in document d. $\boldsymbol{w}_{k,-i}$ is the set of words across all the documents which are currently assigned to topic k, excluding w_i^d. The likelihood term $p(w_i^d|\boldsymbol{w}_{k,-i})$ can be computed by finding the proper conjugate prior and marginalizing out the parameters ϕ. Since the topic-word distribution is assumed to have a Dirichlet prior(parametrized by β) and the word is drawn from a multinomial distribution, the predictive likelihood of w_i^d given dataset $\boldsymbol{w}_{k,-i}$ can be obtained by fraction counting (Zheng, Liu, & Ni, 2014):

$$p(w_i^d|\boldsymbol{w}_{k,-i}) = \frac{\beta + n_{kw}^{-i}}{\sum_v (\beta + n_{kv}^{-i})} \tag{3}$$

where n_{kw}^{-i} denotes the number of the other words in topic k that have the same symbol as w_i^d, while $\sum_v n_{kv}^{-i}$ is the total number of words in topic k excluding w_i^d. Eq. (3) is the likelihood of word w_i^d being generated by topic k given the current word distribution in topic k. Therefore, the clustering is done by assigning the word to the topic in which it has the maximum likelihood.

Bayesian framework has demonstrated its effectiveness in unsupervised and collaborative learning, such as activity discovery (Chikhaoui, Wang, & Pigot, 2012), mobile context discovery (Zheng et al., 2014), frequent human routines discovery (Nguyen, Phung, Gupta, & Venkatesh, 2013; Sun et al., 2014) and collaborative learning (Zheng & Yang, 2011). The primary motivations of using LDA in this paper is to collaboratively leverage limited labelled data from multiple users to learn a general activity model. As we can see from Eq. (2) that, document-topic distribution may be different across the documents, while the estimation of topic-word distribution makes use of the words across the documents and the resulted parameters are globally shared. This characteristic makes it reasonable to leverage LDA to build a general activity model, as each user may have different proportions of activity data, while the estimation of the parameters for each activity class utilizes the corresponding data from all the users, and the resulted parameters are globally shared among the users. Therefore, the latent topics are mapped to human activities, while the words are viewed as feature vectors extracted from sensor data. The activity data of each user forms an individual document and the data of all the users comprises the corpus.

However, LDA cannot be directly applied to activity data. Since the data points are multi-dimension feature vectors extracted from continuous sensor data, it is scarce that two data points are exactly the same even they belong to the same activity class. Therefore, it is infeasible to assume each activity class is multinomial distributed over the feature vectors. A potential solution is to assume Gaussian distribution over the data points that belongs to the same activity class (Nguyen et al., 2013). However, as data points from on-body sensors usually consist of high-dimensional features, the estimation of a large number of parameters would cause the problem of overfitting (Sun et al., 2014). For example, given 561-dimensional data points, we need to estimate 561-dimensional mean vector and a 561 × 561-dimensional covariance matrix for each Gaussian component. In next section, we introduce our method using AdaBoost to estimate the predictive likelihood in Eq. (3).

4. AdaBoost

In this section, we briefly introduce the basic idea of AdaBoost, and then show the method that leverages LDA and AdaBoost to build a general activity model.

4.1. Brief introduction

The essence of AdaBoost is to train an ensemble of weak classifiers and combine them to form a more robust and accurate classifier. Each weak classifier makes decisions based on a single feature and needs only be slightly better than random guessing. The final classifier is a linear combination of the weak classifiers, with each classifier weighted by the error it makes during the training process, more weight is given to the classifier that makes less false prediction. AdaBoost is also able to choose the most discriminative features to perform classification, instead of relying on the whole feature space to make decision, thus avoid the problem of feature redundancy.

As depicted in Algorithm 1, AdaBoost takes the instances, initial instance weights and maximum iterations as input. The training of AdaBoost follows an iterative process. In each iteration, each weak learners is fitted to the training data, and the one with the minimum weighted error is selected (line 2). After that, the instance weights are updated, so that more weight is given to the misclassified instances (line 4). During the next iteration, the weak classifiers focus more on those problematic instances. The output of the training process is an ensemble of weak learners (line 6). Each weak learner is trained on a specific dimension of the feature space. In this paper, we adopt decision stump as the weak learner, then training each weak learner is equivalent to finding the threshold in each dimension that can separate the instances with minimum weight error on a specific dimension. While in the upper level, we loop over all the dimensions, and chose the weak leaner with the minimum weight error.

Algorithm 1 AdaBoost.

Input:

instances $(x_1, y_1), \cdots, (x_n, y_n)$ where $x_i \in \Re^k$ is k-dimension feature vector, $y_i \in \{+1, -1\}$;
Initial weight of n instances $D_0(i) = 1/n$ for $i = 1, \cdots, n$;
Weak learners $h(x) \in \{+1, -1\}$;
Max iterations T;

Output:

Ensemble of weak learners;

1: **for** $t = 1$ to T **do**
2: Find weak learner $h_t(x)$ that minimises the weight error: $\epsilon_t = \sum_{i=1}^{n} D_t(i) I[h_t(x_i) \neq y_i] P_{i,y_i}$;
3: Compute the weight for the weak learner $h_t(x)$: $\alpha_t = \frac{1}{2} ln(\frac{1-\epsilon_t}{\epsilon_t})$;
4: Update the weight of instances: $D_{t+1}(i) = \frac{D_t(i) exp(-\alpha y_i h_t(x_i)) P_{i,y_i}}{\sum_i D_t(i) exp(-\alpha y_i h_t(x_i)) P_{i,y_i}}$ for $i = 1, \cdots, n$;
5: **end for**
6: **return** $H(x) = sign(\sum_{t=1}^{T} \alpha_t h_t(x))$;

Furthermore, we improve the scalability of AdaBoost by introducing the concept of "virtual evidence". Specifically, the data points are soft assigned to different activities with different probabilities (P_{i,y_i} in step 2 and step 4). The soft assignment demonstrates its advantage over the hard assignment in previous work (Huynh et al., 2008; Reiss, Hendeby, & Stricker, 2013; Wang et al., 2007). We introduce the rationale for adopting and estimating soft assignment later in this section.

The algorithm presents the training of binary classification. However, it can be easily extended to multiclass training. Like Lester, Choudhury, Kern, Borriello, and Hannaford (2005), we train an ensemble for each activity class, which follows the "one-to-all" pattern. Specifically, when training the ensemble for a specific activity, all the instances that belong to the activity are labelled as "+1", while the others are labelled as "−1". If we denote the ensemble of weak learners used to discriminative class k against all the others as

$$H^k(x) = \sum_{t=1}^{T} \alpha_t^k h_t^k(x) \tag{4}$$

then given a instance for prediction, we aggregate its evidence against each class using (4), and choose the activity that has the maximum value as prediction:

$$prediction = argmax_k(H^k(x)) \tag{5}$$

AdaBoost is a discriminative classifier, which perform classification by giving the definitive decision. This approach has the potential problem that even if the classifier is uncertain with the class of the instance(i.e. the classification margin using (4) is minor for a data point), it certainly choose the class against which the instance has the maximum as the prediction, using (5). While we argue that the posterior probability of a instance is much more helpful, since it reflects the confidence in that prediction and it is related to the estimation of soft assignment P_{i,y_i} in Algorithm 1.

To this end, we calculate the posterior probability for instances using the method from (Lester et al., 2005).

$$P(C_k|x) = \frac{e^{\psi m^k(x)}}{e^{\psi m^k(x)} + 1} \tag{6}$$

where ψ is a constant and $m^k(x) = \frac{\sum_{t=1}^{T} \alpha_t^k h_t^k(x)}{\sum_{t=1}^{T} \alpha_t^k}$. $P(C_k|x)$ is thus regarded as the posterior probability that instance x belong to activity class C_k.

4.2. Building general model

Due to the underlying assumption of multinomial distribution and the conjugate prior distribution, $p(w_i^d|\boldsymbol{w}_{k,-i})$ is proportional to fraction of the word with the same symbol in the topic, and can be estimated from the words currently assigned to topic k. Similarly, the probability that a multi-dimensional data point belongs to a activity class can also be estimated through the data points that are currently assigned to that class:

$$p(x_i|\boldsymbol{x}_k) = p(x_i|y_i = k) = \frac{p(y_i = k|x_i)p(x_i)}{p(y_i = k)} \propto p(y_i = k|x_i) \tag{7}$$

where $\boldsymbol{x}_k$ are the data points that are currently assigned to activity class k, $p(x_i)$ is a constant when calculating its posterior probabilities and $p(y_i = k)$ is assumed to be equal for different k by class balancing when training AdaBoost. In this way, the predictive likelihood in Eq. (3) can be approximated with the posterior probability in Eq. (6). This is reasonable, as in Eq. (3) the data point is assigned to a topic in which it has the maximum fraction, while an activity data point should be assigned to the latent activity against which it has the maximum posterior probability.

We also exploit the temporal information when sampling the latent activity for the data points, as temporarily adjacent data points tend to have the same activity label. Therefore, we need to consider the topic assignments of neighbouring data points when sampling the topic for current data point, formulated as follows:

$$P(x_i|\boldsymbol{x}_k) \propto \frac{P(y_i = k|x_i)\prod_{j\in N(i)\backslash i} P(y_j = k|x_j)}{Z} \tag{8}$$

where $N(i)$ indicates the neighbouring data points of x_i and Z is the normalization function.

By combining Eqs. (2) and(8), the topic sampling for a data point can be formulated as:

$$P(x_i^d = k|\boldsymbol{z}_{-i}, \boldsymbol{x}) \propto (\alpha + n_k^{d,-i})P(x_i|\boldsymbol{x}_k) \tag{9}$$

The algorithm of building a general activity model from labelled and unlabelled data is presented in Algorithm 2, which follows an iterative Expectation-Maximization process. At E step, we sample the topic for each data point (line 5), and obtain the predictive likelihood that it belongs to each topic. At M step, these predictive likelihoods are viewed as "virtual evidences" and used to train the AdaBoost (line 7). The reason of using "virtual evidence" is that it is robust to noise and misassignments of the topics (Huynh et al., 2008). Initially, when we are uncertain about the latent topic of a data point, the contribution it makes to the weighted error is further weighted by the "virtual evidence" (line 2 in Algorithm 1). As the iterative process proceeds, the model is able to confidently estimate the labels corresponding to the data points, then the "virtual evidence" approximates the real assignment and the EM process (line 7) results a more accurate AdaBoost (Oberlies, 2007).

5. Sequential prediction

When the resulted AdaBoost is deployed for online prediction, we combine it with graphical models to further smooth out the outliers. In this section, we introduce the methods of combining AdaBoost with HMM and CRF, referred to as BoostHMM and BoostCRF. Note that combining classifiers for activity recognition is not a new topic, for example, in Lester et al. (2005); Maekawa et al. (2010) the authors use the posterior probabilities from discriminative classifiers as new input features to train

Algorithm 2 LDA and AdaBoost.

Input:
Labelled dataset from the different users L
Unlabelled dataset U
Convergence criteria σ
Output:
general model
1: Building initial model: training AdaBoost with labelled data
2: Classifying the unlabelled data with AdaBoost and obtaining for each data point a posterior probability for each activity class, $P(y_i = k|x_i)$, $k = 1\cdots K$. (K is the number of activity classes)
3: **while** not converged **do**
4: //e step:
5: Sampling the topic(latent activity) for each data point by combining Eq. (9)
6: //m step:
7: Retraining AdaBoost with labelled data and currently soft topic assignments of the data points given by previous step.
8: Classifying the unlabelled data points using retrained AdaBoost, and obtaining the posterior probabilities for each data point
9: **end while**
10: **return** AdaBoost;

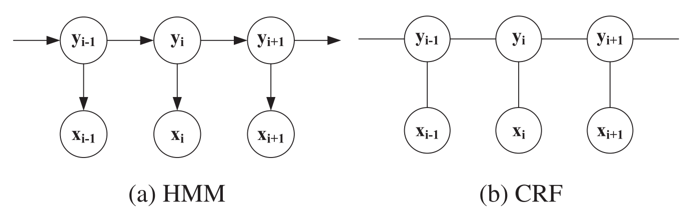

Fig. 2. Graphical model of HMM and CRF.

HMMs. However, the modelling of the discriminative classifier is dissociated from the modelling of the generative classifier. Therefore, the two classifiers are trained independently, using the output of one classifier as input for another. Moreover, they train HMM for each activity class separately, and during the inference phase, they produce the same label that has maximum likelihood for all the instances in a sequence. Therefore, they do not model the transitions among different classes.

5.1. BoostHMM

In HMM, the variables include hidden states and observations and It models the joint distribution of those variables and naively assumes that hidden state y_k at each time step k only depends on hidden state at previous time step, y_{k-1}, while observation x_k at time k only depend the hidden state at the same time slice, as shown in Fig. 2a. Therefore, HMM can be mathematically described by three parameters: the initial state y_1, transition distribution $p(y_k|y_{k-1})$, and emission probability $p(x_k|y_k)$, then the joint distribution of the variables can be formulated as follows:

$$p(\boldsymbol{x}, \boldsymbol{y}) = p(y_1)p(x_1|y_1)\prod_{i=2}^{I} p(y_i|y_{i-1})p(x_i|y_i) \tag{10}$$

We approximate the emission probability $p(x_i|y_i)$ with the posterior distribution $p(y_i|x_i)$ given by AdaBoost using Bayes' rule:

$$p(x_i|y_i) = \frac{p(y_i|x_i)p(x_i)}{p(y_i)} \propto p(y_i|x_i) \tag{11}$$

where prior knowledge $p(y_i)$ is identical for different activities because we balance the training data over all the activity classes. As for the variable x_i that is observed at time i, $p(x_i)$ is a constant when calculating its evidence against different classes. Therefore, the emission probability is proportional to the posterior probability given by AdaBoost, and the joint distribution can be re-formulated as follows:

$$p(\boldsymbol{x}, \boldsymbol{y}) \propto p(y_1)p(y_1|x_1)\prod_{i=2}^{I} p(y_i|y_{i-1})p(y_i|x_i) \tag{12}$$

As for transition probability, we manually set the self-transition probabilities to be large to temporally smooth the activities, and encourage them to continue unless the observation strongly suggests a different activity (Wang et al., 2007), denoted as follows:

$$p(y_i|y_{i-1}) = \begin{cases} 1-\epsilon & y_i = y_{i-1} \\ \epsilon & \text{otherwise} \end{cases} \tag{13}$$

we experimentally set ϵ to be 0.1 in our system, as it is demonstrated to be effective enough to achieve high accuracy. Inferring the hidden states is equivalent to finding the sequences that maximise the joint probability depicted in Eq. (12), which can be performed by the Viterbi algorithm. We use a sliding window with constant number of observations, and perform Viterbi algorithm on this sequence within the window. The window is shifted along the time axis as new instance comes in. In this way, we can provide real-time predictions.

5.2. BoostCRF

Rather than modelling the joint distribution of the variables, Conditional Random Field (CRF) models the conditional distribution of the hidden variables over the observations. The relationships among the connected nodes are now described with potential functions that map them to positive numbers. One advantage of the CRF over HMM is that, it does not assume the dependencies among variables, and it is much more flexible in term of defining the potential functions.

Since it is flexible to define the potential functions, CRF has various structures. In our system, we only consider linear-chain CRF (Fig. 2b). Therefore, we only need to define local potential functions between observation and hidden node at each time step, and pairwise potential functions between consecutive hidden nodes. The conditional distribution can be formulated as:

$$p(\boldsymbol{y}|\boldsymbol{x}) = \frac{exp\left(\sum_{i=1}^{I}\left(\lambda_s^T f_s(y_i, y_{i-1}) + \lambda_j^T f_j(y_i, x_i)\right)\right)}{Z(x)} \tag{14}$$

where $f_j(y_i,\ x_i)$ and $f_s(y_i, y_{i-1})$ are the local and pairwise potential functions at time i. λ_s and λ_j are the corresponding weight vectors, and $Z(x)$ is the normalization factor, formulated as $\sum_{\boldsymbol{y}} exp\left(\sum_{i=1}^{I}\lambda_i^T f_i(y_i, y_{i-1}, x_i)\right)$.

Inspired by Liao, Choudhury, Fox, and Kautz (2007), we map the weak learners trained in AdaBoost to the local potential functions in CRF, while the weights of the potential functions are mapped to the weights of the weak learners. This is reasonable, since more weight are given to the potential functions that better interpret the data, whereas weak learner with less error rate has a larger weight. Using Eq. (4), the weighted sum of local potential functions against activity class k is:

$$\lambda_j^T f_j(y_i, x_i) = \sum_{t=1}^{T}\alpha_t^k h_t^k(x_i) \tag{15}$$

However, mapping the weight of pairwise potential function is non-trivial. To deal with this, we define pairwise potential functions that characterise the temporal transition between any two activities:

$$f_{kl}(y_i, y_{i-1}) = \begin{cases} 1 & y_i = k, y_{i-1} = l \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

where potential function f_{kl} characterises the transition from activity l to activity k. Now assume that there is weak learner $h^k(y_i = k, y_{i-1} = l)$ in AdaBoost that can be mapped to the potential function f_{kl}. Thus, we can estimate the error rate of the weak learner from the training dataset by frequency counting:

$$\epsilon_{kl} = 1 - \frac{\text{expected number of transitions from l to k}}{\text{expected number of transitions out of l}} \tag{17}$$

then according to Algorithm 1, the weight of the weak learner can be approximated as:

$$\alpha_{kl} = \frac{1}{2}ln\left(\frac{1-\epsilon_{kl}}{\epsilon_{kl}}\right) \tag{18}$$

The weight α_{kl} can be mapped to the weight of the pairwise potential function f_{kl} in CRF. Once we have the parameters, the inference process can be carried by dynamic programming to find the most likely latent activities based on the observations. Note that when aggregating the evidence against a particular activity class, only the potential functions corresponding to the weak learners of that class are considered, as each class is trained separately and has it own weak learners.

6. Evaluation

6.1. Datasets

In order to evaluate the proposed methods, we should experiment with datasets that contain activity data from multiple users. We find that the Smartphone dataset (Shoaib, Scholten, & Havinga, 2013) and UCI HAR dataset (Anguita, Ghio, Oneto, Parra, & Reyes-Ortiz, 2012) meet our requirements.

Smartphone dataset (SD) (Shoaib et al., 2013): Activity data was collected from accelerometer, gyroscope and magnetometer on an Android device worn in different body position (arm, belt, waist and pocket), when the 10 subjects performing 7 activities. We compute time domain features such as mean, standard deviation, median, zero crossing rate, variance, root mean square for each axis of the sensors with a 2 s sliding window and 50% overlapped. Since magnetometer is demonstrated to be less discriminative in their work, we only focus on features from accelerometer and gyroscope.

UCI HAR dataset (Anguita et al., 2012): The dataset was collected with accelerometer and gyroscope from a Samsung Galaxy SII smartphone worn by 30 volunteers within an age bracket of 19–48 years. The smartphone was fixed at the waist when the subjects performing six activities. They compute 561 features based on the sliding window of 2.56 s and 50% overlapped. The sensor data was collected at the 50 Hz and manually labelled.

Heterogeneity Dataset for Human Activity Recognition (HHAR) Stisen et al. (2015): The activity data is collected from on-board accelerometer and gyroscopes on 8 smartphones and 2 smartwatches worn by 9 subjects performing six activities. We choose a smartphone data for each type of smartphone and segment the data with a 2 s sliding window and 50% overlapping.

The summaries of the datasets are presented in Table 2. Notice that in SAD, data is collected from 5 body positions, resulting in 5 separate datasets (i.e. SAD-ARM, SAD-BELT, SAD-POCKET, SAD-WRIST, SAD-) with each of them having the same activity classes and instances. In HHAR, the data is split into different sensors and different devices (i.e. acc-nexus4, acc-s3, acc-s3mini, acc-samsungold, gyro-nexus4, gyro-s3, gyro-s3mini), for example, acc-nexus4 stands for the dataset collected from the accelerometer on device nexus4.

Table 2
Dataset description. Notice that in SAD, data is collected from 5 body positions, resulting in 5 separate datasets(e.g. SAD-ARM) with each of them having the same activity classes and instances.

Datasets	Users	Activities (Instances)
SAD	10	walking (8950), standing (8950), jogging (8950), sitting (8950), biking (8950), upstairs (8950), downstairs (8900)
UCI	30	walking (1722), upstairs (1544), downstairs (1406), sitting (1777), standing (1906), lying (1944)
HHAR	9	Biking (17650), Sitting (19169), Standing (17751), Walking (20385), Stair up (16905), Stair down (15199)

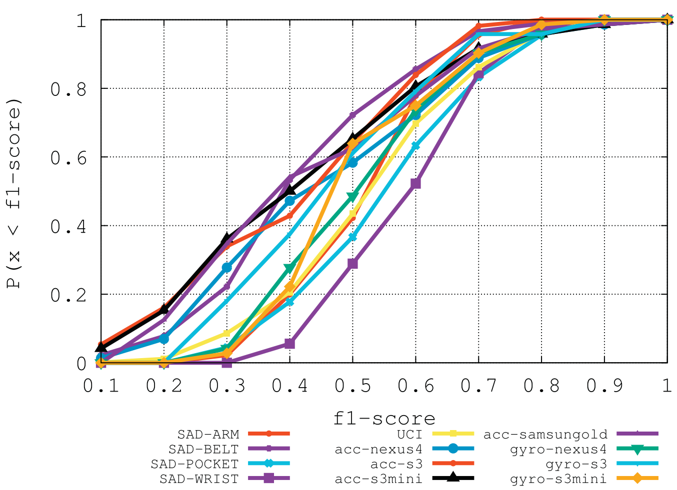

Fig. 3. Cumulative distribution function (CDF) of the F1-score across the datasets when performing pairwise training and testing.

6.2. Pre-analysis

To validate the motivation of building a general activity model, we examine the differences in performing activities among the users by training the activity model with data from one person and testing it on another. We record the F1-score ($F1-score=\frac{2*precision*recall}{precision+recall}$) and present the cumulative distribution function and histogram of the F1-score for each dataset in Figs. 3 and4 respectively. Fig. 4 shows that in most cases, the activity model achieves a low F1-score (0.2–0.8), if it is trained on one person and used to test the data from another. It can be seen from Fig. 3 that 97% of the tests obtains the F-sore under 80%. This experiment demonstrates that, people perform the activities quite differently, and the model trained on individual person causes the problem of overfitting. To deal with the problem, we need to build a general model.

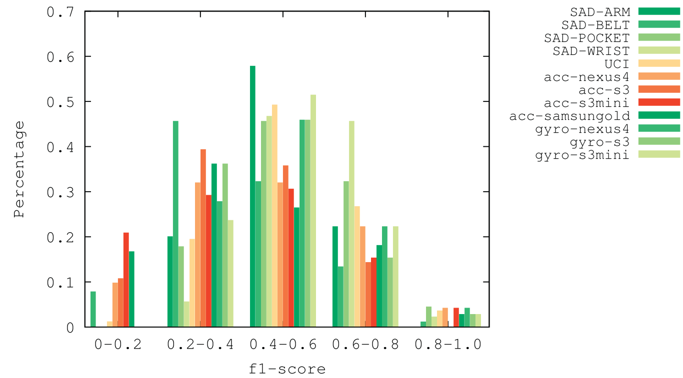

Fig. 4. Histogram of the F1-score across the datasets when performing pairwise train and test.

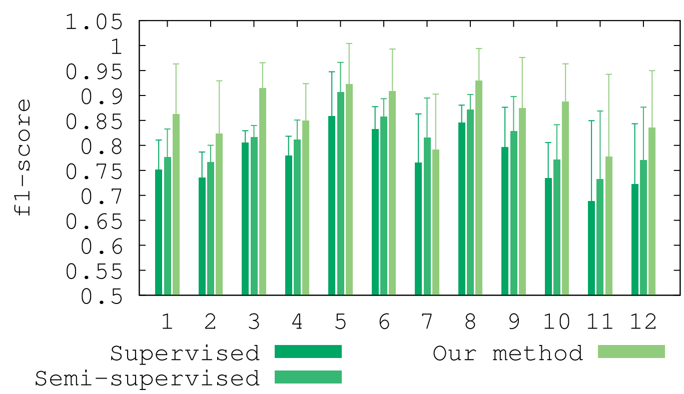

Fig. 5. Comparison results between our method and baselines across the dataset, where the labelled percentage is set to be 1%.

6.3. Comparison

In this section, we compare the proposed method that creates the generic activity model with the following methods:

- *Semi-supervised method*: traditional classifiers that are trained with the initial labelled data, and used to classify the unlabelled instances, then the instances classified with high confidences are selected to retrain the classifiers. For example, the hybrid of LDA and AdaBoost is compared with semi-supervised AdaBoost.
- *Supervised method*: traditional classifiers that are only trained with the initial labelled data. For example, the hybrid of LDA and AdaBoost is compared with supervised AdaBoost.

Since the amount of unlabelled data selected for retraining in semi-supervised method is a free parameter, we vary the parameter and choose the best result of the semi-supervised method for comparison, which means we always compare our method with the best fine-tuned semi-supervised methods. We perform leave-one(user)-out validation and present the average f1-score along

Table 3
Performance(F1-score) improvement when combining general model with HMM and CRF.

Models	SAD-ARM	SAD-BELT	SAD-POCKET	SAD-WRIST	UCI	acc-nexus4
LDA+AdaBoost	0.8662 ± 0.0644	0.8399 ± 0.0591	0.9268 ± 0.0390	0.8558 ± 0.0632	0.8941	0.9198 ± 0.0734
BoostHMM	0.8899 ± 0.0654	0.8583 ± 0.0607	0.9570 ± 0.0325	0.8793 ± 0.0689	0.9118	0.9311 ± 0.0884
BoostCRF	0.9063 ± 0.0657	0.8673 ± 0.0590	0.9724 ± 0.0288	0.8958 ± 0.0673	0.9058	0.9555 ± 0.0496
Models	acc-s3	acc-s3mini	acc-samsungold	gyro-nexus4	gyro-s3	gyro-s3mini
LDA+AdaBoost	0.7825 ± 0.1120	0.9326 ± 0.0575	0.8938 ± 0.0969	0.8894 ± 0.0756	0.7845 ± 0.1683	0.8426 ± 0.1135
BoostHMM	0.8104 ± 0.0606	0.9543 ± 0.0672	0.9024 ± 0.1129	0.9295 ± 0.0876	0.7928 ± 0.1707	0.8602 ± 0.1147
BoostCRF	0.8463 ± 0.0575	0.9752 ± 0.0616	0.9502 ± 0.0618	0.9564 ± 0.0600	0.8228 ± 0.1579	0.9234 ± 0.0648

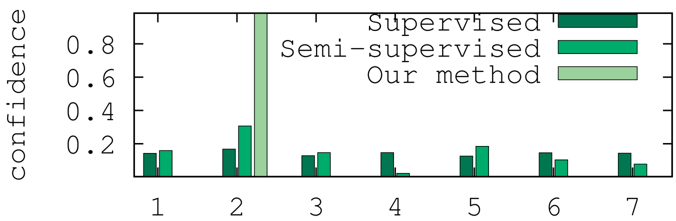

Fig. 6. The posterior distribution of instance when the iterative process converges.

with the standard deviation across all the subjects for each dataset. To study the effect of the amount of initial labelled data, we vary the percentage of labelled data from 1% to 9%. Notice that those percentages of labelled data are randomly sampled to avoid bias.

The comparison results are presented in Fig. 5, where the x-axis stands for the dataset (1. SAD-ARM, 2. SAD-BELT, 3. SAD-POCKET, 4. SAD-WRIST, 5. UCI, 6. acc-nexus4, 7. acc-s3, 8. acc-s3mini, 9. acc-samsungold, 10. gyro-nexus, 11. gyro-s3, 12. gyro-s3mini). We first discuss the results for the labelled percentage set to 1%. The figures

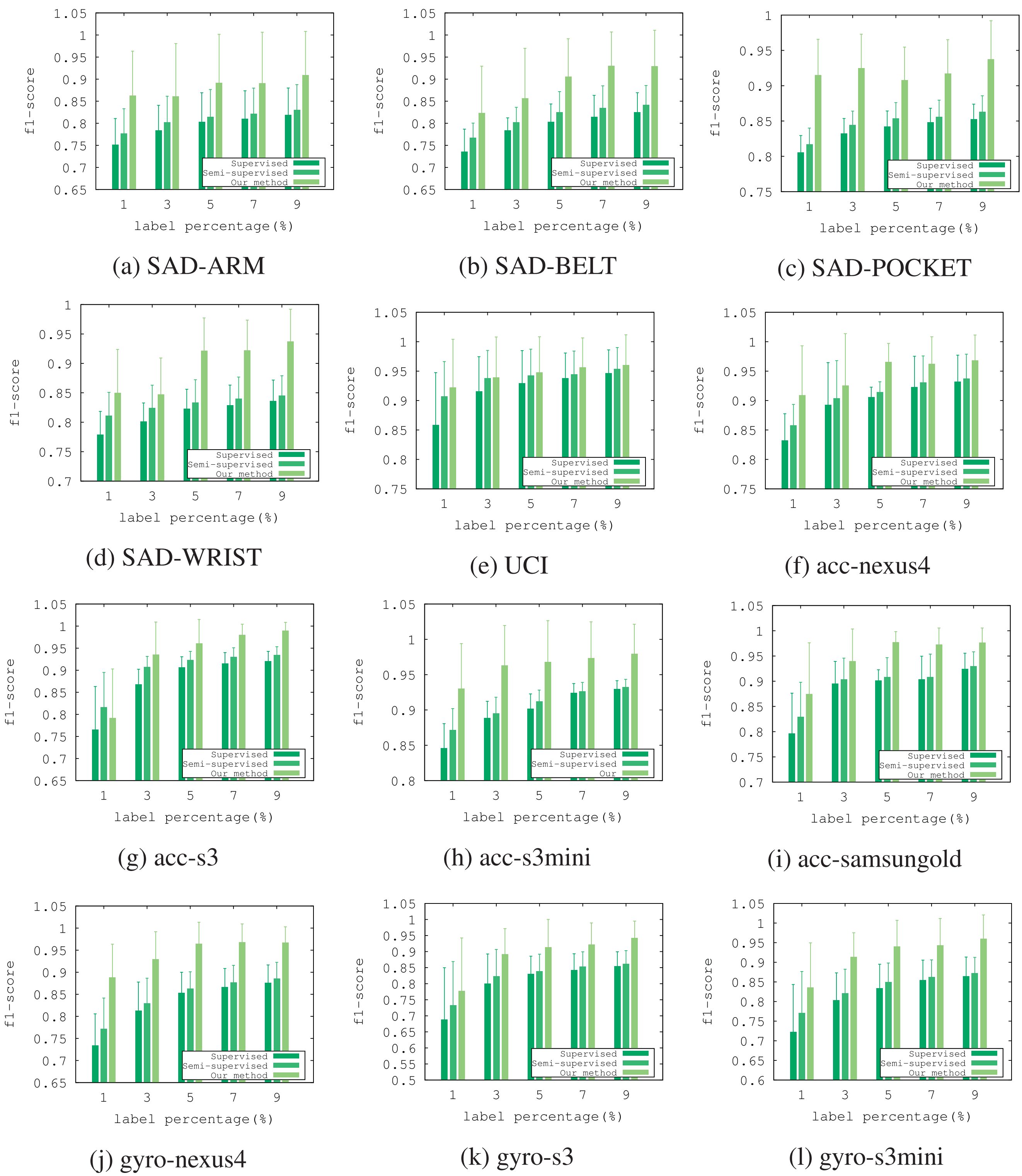

Fig. 7. F1-score with 1%-9% labelled data across datasets.

show that the proposed method is able to create a robust generic model, even if the labelled data is limited. Specifically, the hybrid of LDA and Adaboost is able to achieve an average f1-score 11.4% (max: 21.1%, min: 6.0%) and 8.7% (max: 17.4%, min: 2.8%) higher than the corresponding *Supervised* and *Semi-supervised* baselines, respectively.

The underlying reason for the advantage of our method can be found in Fig. 6, which shows the posterior distribution of a typical unlabelled *walking* instance before and after the iterative process both in our method and *Semi-supervised*. When the instance is originally classified by *Supervised*, the posterior distribution is rather “flat”, which means it is quite uncertain about the true label of the instance. This is because the classifier is trained with limited labelled data from various users, and hence contains much uncertainty when it makes predictions. However, after the iterative EM steps of sampling and retraining, the confidence (0.98) of the instance being activity *walking* approximates to 1 and retraining AdaBoost with this virtual evidence is equivalent to retraining with the true label. As for *Semi-supervised*, the maximum posterior probability(0.306) is still not significant when compared with others, and then retraining with these low confident instances results in less accurate AdaBoost.

From the figures we can also find that the standard deviation of the F1-score across different datasets can be as high as 11%. Inspecting the classification report in detail, we find that certain activities of some users are totally misclassified. The underlying reason is that some people perform certain activities significantly differently from the others. As we perform leave-one-out validation, those activity patterns of the testing subjects are never presented in the training dataset and are frequently mis-recognised during the testing phase, and this problem is exacerbated when we assume that neighbouring instances are related. The reason can also be interpreted as the training data does not experience enough variants of the activity patterns to create a generic model.

To validate our assumption, we increase the percentage of labelled data from 1% to 9%. The results are presented in Fig. 7a–l. One can observe from the experiment on some datasets (e.g. Fig. 7g) that the standard deviation indeed decreases when we increase the amount of labelled data, demonstrating that including more labelled data enable the activity model to be able to deal with more variants of the activity patterns. As for some other datasets, the f1-score difference among the subjects is still large even though we increase the labelled data to 9%. The reason is that the activity data from the other subjects (except the one used for validation) is not diverse enough to create a generic activity model, and adding the activity data of the user used for validation is able to boost the recognition performance Wen, Zhong, and Indulska (2015b).

6.4. Combination with HMM and CRF

In this subsection, we present the improvement in recognition performance (F1-score) when combining our general model with HMM and CRF. We set the percentage of labelled data to be 1% and the result is presented in Table 3.

According to the table, by temporarily smoothing out the outliers, we can improve the F1-score by 2%∼3% and 3%∼5% with BoostHMM and BoostCRF respectively. Notice that BoostCRF slightly outperforms BoostHMM, which has been confirmed by previous work (Van Kasteren et al., 2008). The reason is that, BoostHMM makes strong assumptions among the variables while BoostCRF can have more flexible structures and relationships among connected nodes. Actually, when examining the results provided by BoostHMM, instances of some continuous activity are still sporadically misclassified.

7. Conclusion

In this paper, we collaboratively train a general activity model with partially labelled data by combining LDA and AdaBoost, and combine the general model with HMM and CRF to exploit the temporal information.

Compared with traditional supervised and semi-supervised activity recognition methods, the proposed method is able to achieve higher activity recognition accuracy with a small amount of labelled activity data, and hence alleviate the activity data annotation effort. Furthermore, we propose to combine the AdaBoost with HMM and CRF to exploit the temporal information in human activity, by modelling the activity transition and smoothing out the accidentally misclassified instances we are able to further improve the recognition accuracy.

Building a general activity model can be regarded as the first step in activity model adaptation and refinement. Most of the previous work only consider pre-defined data sources for activity recognition, while ignoring the others that are dynamically available. In the future, we will consider activity adaptation and refinement by incorporating dynamically available data sources.

References

Albinali, F., Intille, S., Haskell, W., & Rosenberger, M. (2010). Using wearable activity type detection to improve physical activity energy expenditure estimation. In *Proceedings of the 12th acm international conference on ubiquitous computing* (pp. 311–320). ACM.

Anguita, D., Ghio, A., Oneto, L., Parra, X., & Reyes-Ortiz, J. L. (2012). Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine. In *Ambient assisted living and home care* (pp. 216–223). Springer.

Azkune, G., Almeida, A., López-de Ipiña, D., & Chen, L. (2015). Extending knowledge-driven activity models through data-driven learning techniques. *Expert Systems with Applications, 42*(6), 3115–3128.

Banos, O., Damas, M., Pomares, H., Prieto, A., & Rojas, I. (2012). Daily living activity recognition based on statistical feature quality group selection. *Expert Systems with Applications, 39*(9), 8013–8021.

Bao, L., & Intille, S. S. (2004). Activity recognition from user-annotated acceleration data. In *Pervasive* (pp. 1–17). Springer.

Brajdic, A., & Harle, R. (2013). Walk detection and step counting on unconstrained smartphones. In *Proceedings of the 2013 acm international joint conference on pervasive and ubiquitous computing* (pp. 225–234). ACM.

Brdiczka, O., Crowley, J. L., & Reignier, P. (2009). Learning situation models in a smart home. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 39*(1), 56–63.

Cao, H., Nguyen, M. N., Phua, C., Krishnaswamy, S., & Li, X.-L. (2012). An integrated framework for human activity classification. In *Proceedings of the 2012 acm conference on ubiquitous computing, ubicomp'2012* (pp. 331–340). ACM.

Chen, L., Hoey, J., Nugent, C. D., Cook, D. J., & Yu, Z. (2012a). Sensor-based activity recognition. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, 42*(6), 790–808.

Chen, L., Nugent, C. D., & Wang, H. (2012b). A knowledge-driven approach to activity recognition in smart homes. *IEEE Transactions on Knowledge and Data Engineering, 24*(6), 961–974.

Cheng, H.-T., Griss, M., Davis, P., Li, J., & You, D. (2013a). Towards zero-shot learning for human activity recognition using semantic attribute sequence model. In *Proceedings of the 2013 acm international joint conference on pervasive and ubiquitous computing, ubicomp'2013* (pp. 355–358). ACM.

Cheng, H.-T., Sun, F.-T., Griss, M., Davis, P., Li, J., & You, D. (2013b). Nuactiv: Recognizing unseen new activities using semantic attribute-based learning. In *Proceeding of the 11th annual international conference on mobile systems, applications, and services, sensys'2013* (pp. 361–374). ACM.

Chernbumroong, S., Cang, S., Atkins, A., & Yu, H. (2013). Elderly activities recognition and classification for applications in assisted living. *Expert Systems with Applications, 40*(5), 1662–1674.

Chikhaoui, B., Wang, S., & Pigot, H. (2012). Adr-splda: Activity discovery and recognition by combining sequential patterns and latent dirichlet allocation. *Pervasive and Mobile Computing, 8*(6), 845–862.

de la Concepción, M. Á., Morillo, L. S., Gonzalez-Abril, L., & Ramírez, J. O. (2014). Discrete techniques applied to low-energy mobile human activity recognition. a new approach. *Expert Systems with Applications, 41*(14), 6138–6146.

Cook, D. J., Krishnan, N. C., & Rashidi, P. (2013). Activity discovery and activity recognition: A new partnership. *Cybernetics, IEEE Transactions on, 43*(3), 820–828.

Fernández-Caballero, A., Castillo, J. C., & Rodríguez-Sánchez, J. M. (2012). Human activity monitoring by local and global finite state machines. *Expert Systems with Applications, 39*(8), 6982–6993.

Gordon, D., Czerny, J., Miyaki, T., & Beigl, M. (2012). Energy-efficient activity recognition using prediction. In *Wearable computers (iswc), 2012 16th international symposium on* (pp. 29–36). IEEE.

Gu, T., Chen, S., Tao, X., & Lu, J. (2010). An unsupervised approach to activity recognition and segmentation based on object-use fingerprints. *Data & Knowledge Engineering, 69*(6), 533–544.

Hevesi, P., Wille, S., Pirkl, G., Wehn, N., & Lukowicz, P. (2014). Monitoring household activities and user location with a cheap, unobtrusive thermal sensor array. In *Ubicomp* (pp. 141–145). ACM.

Huang, P.-C., Lee, S.-S., Kuo, Y.-H., & Lee, K.-R. (2010). A flexible sequence alignment approach on pattern mining and matching for human activity recognition. *Expert Systems with Applications, 37*(1), 298–306.

Huynh, T., Fritz, M., & Schiele, B. (2008). Discovery of activity patterns using topic models. In *Proceedings of the 10th international conference on ubiquitous computing* (pp. 10–19). ACM.

Keally, M., Zhou, G., Xing, G., Wu, J., & Pyles, A. (2011). Pbn: Towards practical activity recognition using smartphone-based body sensor networks. In *Proceedings of the 9th acm conference on embedded networked sensor systems, sensys'2011* (pp. 246–259). ACM.

Khalifa, S., Hassan, M., & Seneviratne, A. (2015). Pervasive self-powered human activity recognition without the accelerometer. *Pervasive computing and communications (percom), 2015 ieee international conference on*. IEEE.

Kwon, Y., Kang, K., & Bae, C. (2014). Unsupervised learning for human activity recognition using smartphone sensors. *Expert Systems with Applications, 41*(14), 6067–6074.

Lee, Y.-S., & Cho, S.-B. (2014). Activity recognition with android phone using mixture-of-experts co-trained with labeled and unlabeled data. *Neurocomputing, 126*, 106–115.

Lester, J., Choudhury, T., Kern, N., Borriello, G., & Hannaford, B. (2005). A hybrid discriminative/generative approach for modeling human activities. *In proc. of the international joint conference on artificial intelligence (ijcai)*.

Liao, L., Choudhury, T., Fox, D., & Kautz, H. A. (2007). Training conditional random fields using virtual evidence boosting.. In *Ijcai: vol. 7* (pp. 2530–2535).

Maekawa, T., & Watanabe, S. (2011). Unsupervised activity recognition with user's physical characteristics data. In *Wearable computers (iswc), 2011 15th annual international symposium on* (pp. 89–96). IEEE.

Maekawa, T., Yanagisawa, Y., Kishino, Y., Ishiguro, K., Kamei, K., Sakurai, Y., & Okadome, T. (2010). Object-based activity recognition with heterogeneous sensors on wrist. In *Pervasive computing* (pp. 246–264). Springer.

Nguyen, T., Phung, D., Gupta, S., & Venkatesh, S. (2013). Extraction of latent patterns and contexts from social honest signals using hierarchical dirichlet processes. In *Pervasive computing and communications (percom), 2013 ieee international conference on* (pp. 47–55). IEEE.

Oberlies, T. (2007). Understanding virtual evidence boosting: A new derivation and application to recognizing spatial context with wearable cameras. *Graphical Models, 2*(5).

Ordóñez, F. J., Iglesias, J. A., De Toledo, P., Ledezma, A., & Sanchis, A. (2013). Online activity recognition using evolving classifiers. *Expert Systems with Applications, 40*(4), 1248–1255.

Patterson, D. J., Fox, D., Kautz, H., & Philipose, M. (2005). Fine-grained activity recognition by aggregating abstract object usage. In *Iswc* (pp. 44–51). IEEE.

Reiss, A., Hendeby, G., & Stricker, D. (2013). Confidence-based multiclass adaboost for physical activity monitoring. In *Proceedings of the 2013 international symposium on wearable computers* (pp. 13–20). ACM.

Reiss, A., & Stricker, D. (2013). Personalized mobile physical activity recognition. In *Proceedings of the 2013 international symposium on wearable computers* (pp. 25–28). ACM.

Riboni, D., Pareschi, L., Radaelli, L., & Bettini, C. (2011). Is ontology-based activity recognition really effective? In *Pervasive computing and communications workshops (percom workshops), 2011 ieee international conference on* (pp. 427–431). IEEE.

Seitr, J., Chiu, W.-C., Fritz, M., Amft, O., & Troster, G. (2015). Joint segmentation and activity discovery using semantic and temporal priors. *Pervasive computing and communications (percom), 2015 ieee international conference on*. IEEE.

Shoaib, M., Scholten, H., & Havinga, P. J. (2013). Towards physical activity recognition using smartphone sensors. In *Ubiquitous intelligence and computing, 2013 ieee 10th international conference on and 10th international conference on autonomic and trusted computing (uic/atc)* (pp. 80–87). IEEE.

Stikic, M., Larlus, D., & Schiele, B. (2009). Multi-graph based semi-supervised learning for activity recognition. In *Wearable computers, 2009. iswc'09. international symposium on* (pp. 85–92). IEEE.

Stikic, M., & Schiele, B. (2009). Activity recognition from sparsely labeled data using multi-instance learning. In *Location and context awareness* (pp. 156–173). Springer.

Stisen, A., Blunck, H., Bhattacharya, S., Prentow, T. S., Kjærgaard, M. B., Dey, A., … Jensen, M. M. (2015). Smart devices are different: Assessing and mitigatingmobile sensing heterogeneities for activity recognition. In *Proceedings of the 13th acm conference on embedded networked sensor systems* (pp. 127–140). ACM.

Sun, F.-T., Yeh, Y.-T., Cheng, H.-T., Kuo, C., & Griss, M. (2014). Nonparametric discovery of human routines from sensor data. In *Pervasive computing and communications (percom), 2014 ieee international conference on* (pp. 11–19). IEEE.

Sundholm, M., Cheng, J., Zhou, B., Sethi, A., & Lukowicz, P. (2014). Smart-mat: Recognizing and counting gym exercises with low-cost resistive pressure sensing matrix. In *Proceedings of the 2014 acm international joint conference on pervasive and ubiquitous computing, ubicomp'2014* (pp. 373–382). ACM.

Tapia, E. M., Intille, S. S., & Larson, K. (2004). *Activity recognition in the home using simple and ubiquitous sensors*. Springer.

Vail, D. L., Veloso, M. M., & Lafferty, J. D. (2007). Conditional random fields for activity recognition. In *Proceedings of the 6th international joint conference on autonomous agents and multiagent systems* (p. 235). ACM.

Van Kasteren, T., Noulas, A., Englebienne, G., & Kröse, B. (2008). Accurate activity recognition in a home setting. In *Proceedings of the 10th international conference on ubiquitous computing* (pp. 1–9). ACM.

Wang, S., Pentney, W., Popescu, A.-M., Choudhury, T., & Philipose, M. (2007). Common sense based joint training of human activity recognizers. In *In: Proceedings of the 20th international joint conference on artificial intelligence*.

Wen, J., & Indulska, J. (2015). Discovering latent structures for activity recognition in smart environments. *2014 ieee uic/atc/scalcom multi-conference*.

Wen, J., Indulska, J., & Zhong, M. (2016). Adaptive activity learning with dynamically available context. In *2016 ieee international conference on pervasive computing and communications (percom)* (pp. 1–11). IEEE.

Wen, J., Loke, S., Indulska, J., & Zhong, M. (2015a). Sensor-based activity recognition with dynamically added context. In *proceedings of the 12th eai international conference on mobile and ubiquitous systems: Computing, networking and services on 12th eai international conference on mobile and ubiquitous systems: Computing, networking and services* (pp. 1–10). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

Wen, J., & Wang, Z. (2016). Sensor-based adaptive activity recognition with dynamically available sensors. *Neurocomputing, 218*, 307–317.

Wen, J., & Zhong, M. (2015). Activity discovering and modelling with labelled and unlabelled data in smart environments. *Expert Systems with Applications, 42*(14), 5800–5810.

Wen, J., Zhong, M., & Indulska, J. (2015b). Creating general model for activity recognition with minimum labelled data. In *Proceedings of the 2015 acm international symposium on wearable computers* (pp. 87–90). ACM.

Wen, J., Zhong, M., & Wang, Z. (2015c). Activity recognition with weighted frequent patterns mining in smart environments. *Expert Systems with Applications, 42*(17), 6423–6432.

Yan, Z., Subbaraju, V., Chakraborty, D., Misra, A., & Aberer, K. (2012). Energy-efficient continuous activity recognition on mobile phones: An activity-adaptive approach. In *Wearable computers (iswc), 2012 16th international symposium on* (pp. 17–24). IEEE.

Ye, J., Dobson, S., & McKeever, S. (2012). Situation identification techniques in pervasive computing: A review. *Pervasive and mobile computing, 8*(1), 36–66.

Zappi, P., Lombriser, C., Stiefmeier, T., Farella, E., Roggen, D., Benini, L., & Tröster, G. (2008). Activity recognition from on-body sensors: Accuracy-power trade-off by dynamic sensor selection. *Wireless sensor networks*.

Zhan, K., Faux, S., & Ramos, F. (2014). Multi-scale conditional random fields for first-person activity recognition. In *Pervasive computing and communications (percom), 2014 ieee international conference on* (pp. 51–59). IEEE.

Zhao, Z., Chen, Y., Liu, J., Shen, Z., & Liu, M. (2011). Cross-people mobile-phone based activity recognition. *Twenty-second international joint conference on artificial intelligence*.

Zheng, J., Liu, S., & Ni, L. M. (2014). Effective mobile context pattern discovery via adapted hierarchical dirichlet processes. In *Mobile data management (mdm), 2014 ieee 15th international conference on: vol. 1* (pp. 146–155). IEEE.

Zheng, V. W., & Yang, Q. (2011). User-dependent aspect model for collaborative activity recognition. In *Ijcai proceedings-international joint conference on artificial intelligence: vol. 22* (p. 2085).