2011 11th IEEE International Conference on Data Mining

A New Multi-Task Learning Method for Personalized Activity Recognition

Xu Sun*†, Hisashi Kashima*, Ryota Tomioka*, Naonori Ueda‡ and Ping Li†
*Department of Mathematical Informatics, The University of Tokyo, Tokyo, JP
†Department of Statistical Science, Cornell University, Ithaca, NY, USA
‡NTT Communication Science Laboratories, Kyoto, JP
xusun@cornell.edu kashima@mist.i.u-tokyo.ac.jp tomioka@mist.i.u-tokyo.ac.jp
ueda@cslab.kecl.ntt.co.jp pingli@cornell.edu

***Abstract*—Personalized activity recognition usually faces the problem of data sparseness. We aim at improving accuracy of personalized activity recognition by incorporating the information from other persons. We propose a new online multi-task learning method for personalized activity recognition. The proposed online multi-task learning method automatically learns the "transfer-factors" (similarities) among different tasks (i.e., among different persons in our case). Experiments demonstrate that the proposed method significantly outperforms existing methods. The novelty of this paper is twofold: (1) A new multi-task learning framework, which can naturally learn similarities among tasks; (2) To our knowledge, this is the first study of large-scale personalized activity recognition.**

I. Introduction

Although there was a considerable literature on sensor based activity recognition, most of the prior work discussed activity recognition in predefined limited environments [1], [2], [3]. For example, most of the prior work assumed the beginning and ending time(s) of each activity are known beforehand, and the constructed recognition system only need to perform simple classifications of activities [1], [2], [3]. However, this is not the case for real-life activity sequences, in which the boundaries of the activities are unknown beforehand [4].

More importantly, to the best of our knowledge, there is *no* previous work that systematically studied personalized activity recognition. Because of the difficulty of collecting training data for activity recognition, most of the prior work simply merge all personal data for training. We will show in our experiments that simply merging the personal data for training an activity recognizer will result in weak performance. Due to the fact that different persons usually have very different activity patterns, it is natural to construct *personalized activity recognizers* (for different persons). However, the new problem is the data sparseness of personalized activity recognition, because usually each person only has very limited amount of labeled training data.

To realize personalized learning in activity recognition, we exploit multi-task learning where each task corresponds to a specific person in activity recognition. We will propose an *online multi-task learning* method for *personalized and continuous* activity recognition.

Table I
Prior accelerometer-based activity recognition studies.

	#Persons	Models	Continuous	Personalize
Bao [1]	20	DTs	×	Limited
Ravi [3]	2	DTs, SVMs	×	×
Pärkkä [2]	16	DTs	×	×
Huynh [5]	1	Bayesian LTM	√	×
Sun [6]	≤ 20	CRFs, LCRFs	√	×
This Work	20	Multi-Task Learner	√	√

II. Related Work and Motivations

A. Activity Recognition

Most of the prior work on activity recognition treated the task as a single-label classification problem [1], [2], [3]. Given a sequence of sensor signals, the activity recognition system predicts a single label (representing a type of activity) for the whole sequence. Ravi *et al.* [3] used decision trees (DTs), support vector machines (SVMs) and K-nearest neighbors (KNNs) models for classification. Bao and Intille [1] and Pärkkä *et al.* [2] used decision trees for classification. A few other works treated the task as a structured classification problem. Huynh *et al.* [5] tried to discover latent activity patterns by using a Bayesian latent topic model (Bayesian LTM). Most recently, Sun *et al.* [6], [4] used conditional random fields (CRFs) and latent conditional random fields (LCRFs) for activity recognition.

To our knowledge, there is only very limited work on the study of personalized activity recognition. A major reason is that most of the previous studies contain only a few participants. The limited number of participants is inadequate for a reliable study of personalized activity recognition. For example, in Ravi *et al.* [3], the data was collected from two persons. In Huynh *et al.* [5], the data was collected from only one person. In Pärkkä *et al.* [2], the data was collected from 16 persons. Because of the difficulty of collecting training data, most of the prior work simply merge all personalized data for training. Personalized activity recognition and how to solve data sparseness in personalized activity recognition were not adequately studied. Table I summarizes prior work on activity recognition.

1550-4786/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDM.2011.14

B. Conditional Random Fields

Conditional random fields (CRFs) are very popular models for structured classification [7]. Assuming a feature function that maps a pair of observation sequence $\mathbf{x}$ and label sequence $\mathbf{y}$ to a global feature vector $\mathbf{f}$, the probability of a label sequence $\mathbf{y}$ conditioned on the observation sequence $\mathbf{x}$ is modeled as follows [7]:

$$P(\mathbf{y}|\mathbf{x},\mathbf{w}) = \frac{\exp\left[\mathbf{w}^\top \mathbf{f}(\mathbf{y},\mathbf{x})\right]}{\sum_{\forall \mathbf{y}'} \exp\left[\mathbf{w}^\top \mathbf{f}(\mathbf{y}',\mathbf{x})\right]}, \tag{1}$$

where $\mathbf{w}$ is a parameter vector.

Given a training set consisting of n labeled sequences, $(\mathbf{x}_i, \mathbf{y}_i)$, for $i = 1 \ldots n$, parameter estimation is performed by maximizing an objective function. For simplicity, we denote $\log P(\mathbf{y}_i|\mathbf{x}_i,\mathbf{w})$ as $\ell(i,\mathbf{w})$. The final objective function is as follows:

$$\mathcal{L}(\mathbf{w}) = \sum_{i=1}^{n} \ell(i,\mathbf{w}) - \frac{||\mathbf{w}||^2}{2\sigma^2}. \tag{2}$$

C. Stochastic Gradient Descent

To speed up the training, people turn to online training methods. A representative online training method is the stochastic gradient descent (SGD) [8]. Suppose $\hat{\mathcal{S}}$ is a randomly drawn subset of the full training set $\mathcal{S}$, the stochastic objective function is then given by

$$\mathcal{L}_{stoch}(\mathbf{w},\hat{\mathcal{S}}) = \sum_{i\in\mathcal{S}} \ell(i,\mathbf{w}) - \frac{|\hat{\mathcal{S}}|}{|\mathcal{S}|}\frac{||\mathbf{w}||^2}{2\sigma^2}.$$

The extreme case is a batch size of 1, and it gives the maximum frequency of updates, which we adopt in this work. In this case, $|\hat{\mathcal{S}}| = 1$ and $|\mathcal{S}| = n$ (suppose the full training set contains n samples). In this case, we have

$$\mathcal{L}_{stoch}(\mathbf{w},\hat{\mathcal{S}}) = \ell(i,\mathbf{w}) - \frac{1}{n}\frac{||\mathbf{w}||^2}{2\sigma^2}, \tag{3}$$

where $\hat{\mathcal{S}} = \{i\}$. The model parameters are updated in such a way:

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \gamma_k \nabla_{\mathbf{w}_k} \mathcal{L}_{stoch}(\mathbf{w},\hat{\mathcal{S}}), \tag{4}$$

where k is the update counter, γ_k is the learning rate [6], [4].

D. Multi-Task Learning

There was quite limited study on systematically combining online learning with multi-task learning. The existing multi-tasking learning methods are mainly focused on matrix regularization (e.g., [9], [10]), and online learning is not well studied in such settings. Two recent studies considered online learning in multi-task setting [11], [12]. Our multi-task learning proposal will be substantially different from them. While Yang *et al.* [12] focused on multi-task feature selection and Agarwal *et al.* [11] focused on online matrix regularization, our proposal relates to neither feature selection nor matrix regularization. We will propose a tighter combination of online learning and multi-task learning, with a new objective function and a novel training method.

III. A New Multi-Task Learning Framework

In this section, we introduce the multi-task learning framework. For every positive integer q, we define $\mathcal{N}_q = \{1,\ldots,q\}$. Let T be the number of tasks (number of persons in activity recognition) which we want to simultaneously learn. For each task $t \in \mathcal{N}_T$, there are n data examples $\{(\mathbf{x}_{t,i}, \mathbf{y}_{t,i}) : i \in \mathcal{N}_n\}$ available. In practice, the number of examples per task may vary but we have kept it constant for simplicity of notation. We use $\mathbf{D}$ to denote the $n \times T$ matrix whose t-th column is given by the vector $\mathbf{d}_t$ of data examples.

A. Model

Our goal is to learn the vectors $\mathbf{w}_1,\ldots,\mathbf{w}_T$ from the data $\mathbf{D}$. For denotational simplicity, we assume that each of the weight vectors is of the same size f (feature dimension), and corresponds to the same ordering of features. We use $\mathbf{W}$ to denote the $f \times T$ matrix whose t-th column is given by the vector $\mathbf{w}_t$. We learn $\mathbf{W}$ by maximizing[1] the objective function,

$$\mathrm{Obj}(\mathbf{W},\mathbf{D}) \triangleq \mathrm{Likelihood}(\mathbf{W},\mathbf{D}) - R(\mathbf{W}), \tag{5}$$

where $\mathrm{Likelihood}(\mathbf{W},\mathbf{D})$ is the averaged likelihood on the tasks, namely,

$$\mathrm{Likelihood}(\mathbf{W},\mathbf{D}) = \sum_{t\in\mathcal{N}_T} \mathcal{L}(\mathbf{w}_t,\mathbf{D}), \tag{6}$$

and $\mathcal{L}(\mathbf{w}_t,\mathbf{D})$ is defined as follows:

$$\mathcal{L}(\mathbf{w}_t,\mathbf{D}) \triangleq \sum_{t'\in\mathcal{N}_T} \left[\alpha_{t,t'}\mathcal{L}(\mathbf{w}_t,\mathbf{d}_{t'})\right]. \tag{7}$$

$\alpha_{t,t'}$ is a real-valued *transfer-factor* between two tasks, with $\alpha_{t,t'} = \alpha_{t',t}$ (symmetric). Intuitively, a transfer-factor $\alpha_{t,t'}$ measures the *similarity* between the t-th task and the t'-th task. For example, in activity recognition, $\alpha_{t,t'}$ estimates the *similarity* of the activity patterns between the person t and the person t'. $\mathcal{L}(\mathbf{w}_t,\mathbf{d}_{t'})$ is defined as follows:

$$\begin{aligned}\mathcal{L}(\mathbf{w}_t,\mathbf{d}_{t'}) &\triangleq \sum_{i\in\mathcal{N}_n} \log P(\mathbf{y}_{t',i}|\mathbf{x}_{t',i},\mathbf{w}_t) \\ &= \sum_{i\in\mathcal{N}_n} \ell_{t'}(i,\mathbf{w}_t),\end{aligned} \tag{8}$$

where $P(\cdot)$ is a prescribed probability function. In this paper, we use the CRF probability function, Eq. (1). The second step is just a simplified denotation by defining $\ell_{t'}(i,\mathbf{w}_t) \triangleq \log P(\mathbf{y}_{t',i}|\mathbf{x}_{t',i},\mathbf{w}_t)$.

[1] *Maximization* is only for simplicity of presentation. Actually, we minimize the $-\log$ of the objective function.

Algorithm Learning with *fixed* transfer-factors (OMT-F)
Input: $\mathbf{W} \leftarrow \mathbf{0}, \mathbf{D}, \mathbf{A}^*$
for $t \leftarrow 1$ to T
for 1 to *convergence*
. **for** 1 to n
. . $\mathbf{g}_t \leftarrow -\frac{1}{n}\nabla_{\mathbf{w}_t}\frac{||\mathbf{w}_t||^2}{2\sigma_t^2}$
. . **for** $t' \leftarrow 1$ to T
. . . Draw $i \in \mathcal{N}_n$ uniformly at random
. . . $\mathbf{g}_t \leftarrow \mathbf{g}_t + \mathbf{A}^*_{t,t'}\nabla_{\mathbf{w}_t}\ell_{t'}(i, \mathbf{w}_t)$
. . $\mathbf{w}_t \leftarrow \mathbf{w}_t + \gamma\mathbf{g}_t$
Output: $\forall t$, $\mathbf{w}_t$ converges to $\mathbf{w}_t^*$; i.e., $\mathbf{W}$ converges to $\mathbf{W}^*$.

Algorithm Learning with *unknown* transfer-factors (OMT)
Input: $\mathbf{W} \leftarrow \mathbf{0}, \mathbf{D}, \mathbf{A} \leftarrow \mathbf{0}$
for 1 to *convergence*
. $\mathbf{W} \leftarrow$ OMT-F$(\mathbf{W}, \mathbf{D}, \mathbf{A})$
. **for** $t \leftarrow 1$ to T
. . **for** $t' \leftarrow 1$ to T
. . . Update $\mathbf{A}_{t,t'}$ using Eq. (15) or Eq. (14)
Output: $\mathbf{A}$ empirically converges to $\hat{\mathbf{A}}$. $\mathbf{W}$ converges to $\hat{\mathbf{W}}$.

Figure 1. Online multi-task learning algorithms (using batch size of 1). The derivation of $\frac{1}{n}$ before the regularization term was explained in Eq. (3).

Finally, $R(\mathbf{W})$ is a regularization term for dealing with overfitting. In this paper, we simply use L_2 regularization:

$$R(\mathbf{W}) = \sum_{t \in \mathcal{N}_T} \frac{||\mathbf{w}_t||^2}{2\sigma_t^2}. \tag{9}$$

To summarize, our multi-task learning objective function is as follows:

$$\text{Obj}(\mathbf{W}, \mathbf{D}) = \sum_{t,t' \in \mathcal{N}_T} \left[\alpha_{t,t'} \sum_{i \in \mathcal{N}_n} \ell_{t'}(i, \mathbf{w}_t)\right] - \sum_{t \in \mathcal{N}_T} \frac{||\mathbf{w}_t||^2}{2\sigma_t^2}.$$

To simplify denotation, we introduce a $T \times T$ matrix $\mathbf{A}$, such that $\mathbf{A}_{t,t'} \triangleq \alpha_{t,t'}$. We also introduce a $T \times T$ functional matrix $\mathbf{\Phi}$, such that $\mathbf{\Phi}_{t,t'} \triangleq \mathcal{L}(\mathbf{w}_t, \mathbf{d}_{t'})$. Then, the objective function can be compactly expressed as follows:

$$\text{Obj}(\mathbf{W}, \mathbf{D}) = \text{tr}(\mathbf{A}\mathbf{\Phi}^\top) - \sum_{t \in \mathcal{N}_T} \frac{||\mathbf{w}_t||^2}{2\sigma_t^2}, \tag{10}$$

where tr means *trace*. In the following content, we will first discussion a simple case that the transfer-factor matrix $\mathbf{A}$ is fixed. After that, we will focus on the case that $\mathbf{A}$ is unknown.

B. Learning with Fixed Transfer-Factors

With fixed transfer-factors, the optimization problem is as follows:

$$\mathbf{W}^* = \underset{\mathbf{W}}{\text{argmax}} \left[\text{tr}(\mathbf{A}^*\mathbf{\Phi}^\top) - \sum_{t \in \mathcal{N}_T} \frac{||\mathbf{w}_t||^2}{2\sigma_t^2}\right]. \tag{11}$$

It is clear to see that we can independently optimize $\mathbf{w}_t$ and $\mathbf{w}_{t'}$ when $t \neq t'$. In other words, we can independently optimize each column of $\mathbf{W}$, and therefore derive the optimal weight matrix $\mathbf{W}^*$. For $\mathbf{w}_t$ (i.e., the t'th column of $\mathbf{W}$), its optimal form is:

$$\mathbf{w}_t^* = \underset{\mathbf{w}_t}{\text{argmax}}\, \psi(\mathbf{w}_t, \mathbf{D}), \tag{12}$$

where $\psi(\mathbf{w}_t, \mathbf{D})$ has the form as follows:

$$\psi(\mathbf{w}_t, \mathbf{D}) = \sum_{t' \in \mathcal{N}_T} \left[\alpha^*_{t,t'}\mathcal{L}(\mathbf{w}_t, \mathbf{d}_{t'})\right] - \frac{||\mathbf{w}_t||^2}{2\sigma_t^2}. \tag{13}$$

This optimization problem is a cost-sensitive optimization problem. We present a cost-sensitive online training algorithm, called *online multi-task learning with fixed transfer-factors (OMT-F)*, for this optimization. The OMT-F algorithm is shown in Figure 1.

Given certain conditions, we can theoretically show that the parameters $\mathbf{W}$ produced by the OMT-F online learning algorithm are convergent towards the maximum $\mathbf{W}^*$ of Eq. (10). For saving space, we omit the details of convergence analysis. We can also see the convergence of the proposed method in the section of experiments.

C. Learning with Unknown Transfer-factors

For many practical applications, the transfer-factors are hidden variables that are unknown. To solve this problem, we present a heuristic learning algorithm, called OMT, to learn transfer-factors and model weights in alternating optimization (see the bottom of Figure 1). Here, the OMT-F algorithm is employed as a subroutine. In the beginning, model weights $\mathbf{W}$ and transfer-factors $\mathbf{A}$ are initialized by $\mathbf{0}$ matrix. $\mathbf{W}$ is then optimized to $\hat{\mathbf{W}}$ by using the OMT-F algorithm, based on the fixed $\mathbf{A}$. Then, in an alternative way, $\mathbf{A}$ is updated based on the optimized weights $\hat{\mathbf{W}}$. After that, $\mathbf{W}$ are optimized again based on updated (and fixed) transfer-factors. This iterative process continues until empirical convergence of $\mathbf{A}$ and $\mathbf{W}$.

In updating transfer-factors $\mathbf{A}$ based on $\mathbf{W}$, a natural idea is to estimate a transfer-factor $\alpha_{t,t'}$ based on the similarity between weight vectors, $\mathbf{w}_t$ and $\mathbf{w}_t$. The similarity between weight vectors can be calculated by using kernels, including the popular Gaussian RBF and polynomial kernels. We can define Gaussian RBF kernel to estimate similarity between two tasks:

$$\alpha_{t,t'} \triangleq \frac{1}{C}\exp(-\frac{||\mathbf{w}_t - \mathbf{w}_{t'}||^2}{2\sigma^2}), \tag{14}$$

Table II
FEATURES USED IN THE ACTIVITY RECOGNITION TASK. $\mathcal{A} \times \mathcal{B}$ MEANS A CARTESIAN PRODUCT BETWEEN TWO SETS; i REPRESENTS THE WINDOW INDEX; y_i AND $y_{i-1}y_i$ REPRESENTS CRF LABEL AND LABEL-TRANSITION. SINCE THE SINGLE-AXIS BASED FEATURES ON THE THREE AXES ARE EXTRACTED IN THE SAME WAY, FOR SIMPLICITY, WE ONLY DESCRIBE THE FEATURES ON ONE AXIS. FOR MULTI-AXIS BASED FEATURES, WE USE 1, 2, AND 3 TO INDEX/REPRESENT THE THREE AXES.

Single-axis based features:
(1) Signal strength features: $\{s_{i-2},\ s_{i-1},\ s_i,\ s_{i+1},\ s_{i+2},\ s_{i-1}s_i,\ s_is_{i+1}\} \times\{y_i,\ y_{i-1}y_i\}$
(2) Mean feature: $m_i \times\{y_i,\ y_{i-1}y_i\}$
(3) Standard deviation feature: $d_i \times\{y_i,\ y_{i-1}y_i\}$
(4) Energy feature: $e_i \times\{y_i,\ y_{i-1}y_i\}$
Multi-axis based features:
(1) Correlation features: $\{c_{1,2,i}, c_{2,3,i}, c_{1,3,i}\} \times\{y_i,\ y_{i-1}y_i\}$

Table III
RESULTS ON THE DATA OF 5 PERSONS, 10 PERSONS, AND 20 PERSONS. OMT IS THE PROPOSED METHOD. *SGD-Single* IS THE PERSONALIZED SGD TRAINING; *SGD-Merged* IS THE MERGED SGD TRAINING.

#Person = 5	Ov. Accuracy (St. Deviation)
SGD-Merged	57.65 (±1.06)
SGD-Single	68.19 (±0.19)
OMT, C=80 (prop.)	**69.84** (±0.86)
#Person = 10	Ov. Accuracy (St. Deviation)
SGD-Merged	63.25 (±0.16)
SGD-Single	68.34 (±0.25)
OMT, C=40 (prop.)	**72.80** (±0.61)
#Person = 20	Ov. Accuracy (St. Deviation)
SGD-Merged	62.53 (±1.12)
SGD-Single	62.46 (±0.56)
OMT (prop.)	**63.90** (±0.40)

where C is a real-valued constant for tuning the magnitude of transfer-factors. Intuitively, a big C will result in "weak multi-tasking" and a small C will make "strong multi-tasking". σ is used to control the variance of a Gaussian RBF function. Alternatively, we can use polynomial kernel (normalized) to estimate similarities between tasks:

$$\alpha_{t,t'} \triangleq \frac{1}{C}\frac{\langle \mathbf{w}_t, \mathbf{w}_{t'}\rangle^d}{||\mathbf{w}_t||^d \cdot ||\mathbf{w}_{t'}||^d}, \tag{15}$$

where $\langle \mathbf{w}_t, \mathbf{w}_{t'}\rangle$ means inner product between the two vectors (i.e., $\mathbf{w}_t^\top \mathbf{w}_{t'}$); d is the degree of the polynomial kernel; $||\mathbf{w}_t||^d \cdot ||\mathbf{w}_{t'}||^d$ is the normalizer; C is a real-value constant for controlling the magnitude of transfer-factors. Actually the normalized polynomial kernel is natural and easy to understand. For simplicity, we typically set $d = 1$. In preliminary experiments, we find the polynomial kernel works better (more robust) than the RBF kernel. Hence, we will focus on the polynomial kernel in the experiments.

D. Accelerated OMT Learning

The OMT learning algorithm can be further accelerated using more frequent update of the transfer-factors, $\mathbf{A}$. The naive OMT learning algorithm waits for the convergence of the model weights $\mathbf{W}$ (in the OMT-F step) before updating the transfer-factors $\mathbf{A}$. In practice, we can update transfer-factors $\mathbf{A}$ before the convergence of the model weights $\mathbf{W}$. For example, we can update transfer-factors $\mathbf{A}$ after running only one iteration of the OMT-F algorithm. This can bring a much faster empirical convergence of the OMT learning. We will adopt this accelerated version of the OMT learning for experiments. In the experiment section, we will compare the (accelerated) OMT method with a variety of strong baseline methods.

IV. EXPERIMENTS ON ALKAN DATA

We use the ALKAN dataset [13] for experiments. This dataset contains 2,061 sessions, with totally 3,899,155 samples (in a temporal sequence). The data was collected by *iPod* accelerometers with the sampling frequency of 20HZ. A sample contains 4 values: time stamp and triaxial singals. For example, {539.266(s), 0.091(g), -0.145(g), -1.051(g)} [2]. There are five kinds of activity labels: `act-0` means *"walking/running"*, `act-1` means *"on elevator/escalator"*, `act-2` means *"taking car/bus/train"*, `act-3` means *"standing/sitting/discussing/at-dinner"*, and `act-4` means *"other (more trivial) activities (e.g., dressing)"*.

We randomly selected 85% of samples for training, 5% samples for tuning hyper-parameters (development data), and the rest 10% samples for testing. Following [6], the evaluation metric are sample-accuracy (%) (the number of correctly predicted samples divided by the total number of samples). We also considered other evaluation metrics, like precision and recall, in preliminary experiments. However, we found precision and recall tended to be misleading in this task, because an activity segment is very long (typically contains thousands of time-windows), and small difference on the boundaries of segments can cause very different precision and recall. On the other hand, the accuracy metric is much more reliable in this scenario.

A. Feature Engineering

Following prior work in activity recognition [1], [2], [3], [5], we use acceleration features, mean features, standard deviation, energy, and correlation features (see Table II). We denote the window index as i. The mean feature is simply the averaged signal strength in a window: $m_i = \frac{\sum_{k=1}^{|w|} s_k}{|w|}$, where $s_1, s_2, \ldots$ are the signal magnitudes in a window. The energy feature is defined as $e_i = \frac{\sum_{k=1}^{|w|} s_k^2}{|w|}$. The deviation feature is defined as $d_i = \sqrt{\frac{\sum_{k=1}^{|w|}(s_k - m_i)^2}{|w|}}$, where the m_i is the mean value defined before. The correlation feature is defined as $c_{1,2,i} = \frac{covariance_{1,2,i}}{d_{1,i}d_{2,i}}$, where the $d_{1,i}$ and $d_{2,i}$ are the deviation values on the i'th window of the axis-1 and the

[2]In the example, 'g' is the *acceleration of gravity*.

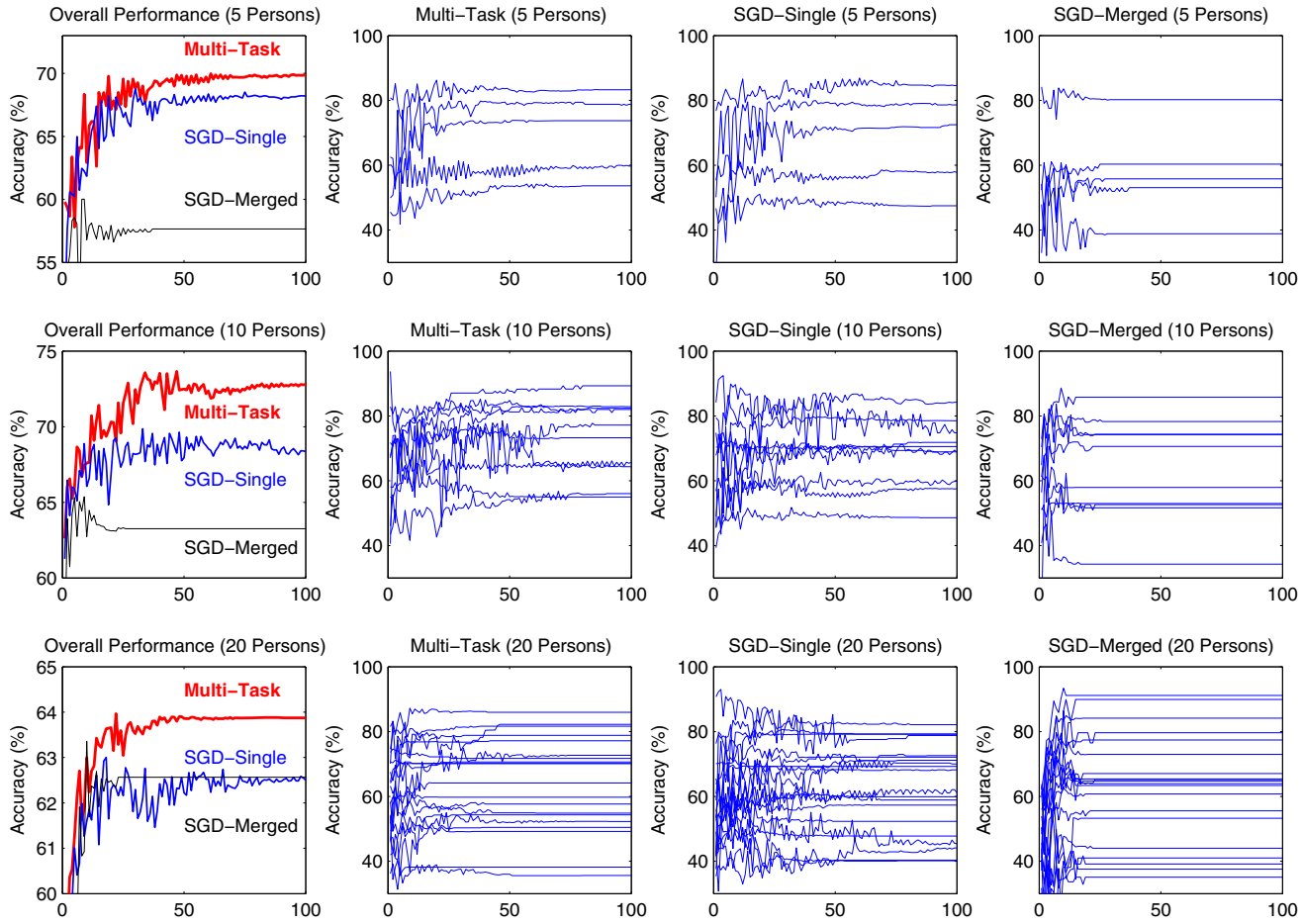

Figure 2. Overall and personal accuracy curves of the different methods (5-person, 10-person, and 20-person, respectively). The overall accuracy curves are used to compare the OMT method with baselines. The personal accuracy curves are used for showing the diversity and distribution of the performance among persons.

axis-2, respectively. The $covariance_{1,2,i}$ is the covariance value between the i'th windows of the axis-1 and the axis-2. We defined correlation feature between other axis pairs in the same manner.

B. Experimental Setting

Two baselines are adopted, including the SGD-Single training for each single person (using only this person's data for training), and the SGD-Merged training (merging all the training data of different persons to train a unified model).

We employed an L_2 prior for all methods, by setting the variance $\sigma = 2$. For the OMT method, its hyper-parameters (i.e., C and d) are tuned by using development data. In preliminary experiments, we find using $d = 1$ worked well. For C, we test $C = 5, 10, 20, 40, 80, 160$ on development data, and choose the optimal one. We will show detailed values of C in experimental results.

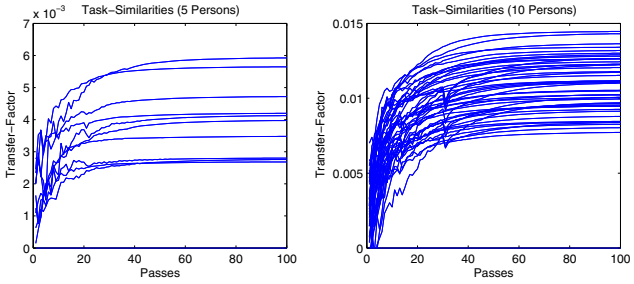

Figure 3. Convergence of the learned OMT transfer-factors between personal data pairs. A curve (i, j) corresponds to $\alpha_{i,j}$ between the persons i and j (that is, the task-similarity between person i and j). For simplicity, we omit the (i, j) information (because we only focus on the convergence here). Similar tendencies were also observed on the 20-person data.

C. Results and Discussion

To study multi-task learning with different scales, we perform experiments on 5-person, 10-person, and 20-person

data in an incremental way (see Table III). As we can see, the OMT method significantly outperformed baselines.

Note that the *overall* accuracies of 5-person, 10-person, and 20-person datasets are not directly comparable to each other, simply because the datasets are different. For example, the 20-person dataset contains the newly-added 15 persons (compared with the 5-person dataset), and the newly-added 15 persons may have more noisy data. Nevertheless, the personal accuracies for specific persons are comparable among different scales.

1) Overall and Personal Curves: In Figure 2, we show the accuracy curves by varying the number of training passes. From the overall curves, we can clearly see the superiority of the OMT method over other methods in different scales.

We can see the personal curves are very diversified, and simply merging their data for unified SGD training is frustrating. The OMT method is an ideal solution for this diversified situation.

2) Convergence of Transfer-Factors: In Figure 3, we show curves of the transfer-factors. As we can see, the transfer-factors were convergent as the OMT learning went on.

In principle, the proposed method should be able to learn even negative transfer-factors among tasks (e.g., if task *a* and task *b* have opposite patterns). However, in this dataset, we did not observe negative similarities. We observed that all transfer-factors were non-negative. This is also good in another aspect (convex analysis): it indicates that the objective function of multi-task learning will be convex and its optimum will be unique.

V. Conclusions and Future Work

We studied personalized activity recognition, and proposed a new multi-task learning method, which can naturally learn similarities among different tasks (persons). Experiments demonstrated that personalized activity recognition with multi-task learning performed much better than single-person based learning and merged learning. Note that the proposed multi-task learning method is a general technique, and it can be easily applied to other tasks. As future work, we plan to apply this method to other large-scale data mining tasks.

Acknowledgments

X.S., H.K., and N.U. are supported by the FIRST Program of JSPS. P.L. is partially supported by the National Science Foundation (DMS-0808864) and the Office of Naval Research (YIP-N000140910911).

References

[1] L. Bao and S. S. Intille, "Activity recognition from user-annotated acceleration data." in *Pervasive Computing*, ser. Lecture Notes in Computer Science 3001. Springer, 2004, pp. 1–17.

[2] J. Präkkä, M. Ermes, P. Korpipää, J. Mäntyjärvi, J. Peltola, and I. Korhonen, "Activity classification using realistic data from wearable sensors." *IEEE Transactions on Information Technology in Biomedicine*, vol. 10, no. 1, pp. 119–128, 2006.

[3] N. Ravi, N. Dandekar, P. Mysore, and M. L. Littman, "Activity recognition from accelerometer data." in *Proceedings of AAAI'05*, 2005, pp. 1541–1546.

[4] X. Sun, H. Kashima, R. Tomioka, and N. Ueda, "Large scale real-life action recognition using conditional random fields with stochastic training." in *PAKDD (2011)*, ser. Lecture Notes in Computer Science, J. Z. Huang, L. Cao, and J. Srivastava, Eds., vol. 6635. Springer, 2011, pp. 222–233.

[5] T. Huynh, M. Fritz, and B. Schiele, "Discovery of activity patterns using topic models," in *Proceedings of the 10th international conference on Ubiquitous computing*. ACM, 2008, pp. 10–19.

[6] X. Sun, H. Kashima, T. Matsuzaki, and N. Ueda, "Averaged stochastic gradient descent with feedback: An accurate, robust, and fast training method." in *Proceedings of the 10th International Conference on Data Mining (ICDM'10)*, 2010, pp. 1067–1072.

[7] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the 18th International Conference on Machine Learning (ICML'01)*, 2001, pp. 282–289.

[8] L. Bottou, "Online algorithms and stochastic approximations," *Online Learning and Neural Networks. Saad, David. Cambridge University Press*, 1998.

[9] A. Argyriou, T. Evgeniou, and M. Pontil, "Convex multi-task feature learning." *Machine Learning*, vol. 73, no. 3, pp. 243–272, 2008.

[10] Y. Xue, D. Dunson, and L. Carin, "The matrix stick-breaking process for flexible multi-task learning," in *Proceedings of the 24th international conference on Machine learning (ICML'07)*. Corvalis, Oregon: ACM, 2007, pp. 1063–1070.

[11] A. Agarwal, A. Rakhlin, and P. Bartlett, "Matrix regularization techniques for online multitask learning," EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2008-138, Oct 2008.

[12] H. Yang, I. King, and M. R. Lyu, "Online learning for multi-task feature selection." in *Proceedings of CIKM'10*. ACM, 2010, pp. 1693–1696.

[13] Y. Hattori, M. Takemori, S. Inoue, G. Hirakawa, and O. Sudo, "Operation and baseline assessment of large scale activity gathering system by mobile device," in *Proceedings of DICOMO'10*, 2010.