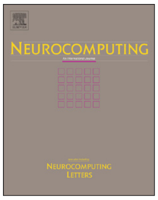
Neurocomputing 139 (2014) 97–106

Contents lists available at ScienceDirect

Neurocomputing

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



Multi-task and multi-view learning of user state

Melih Kandemir [a,*], Akos Vetek [b], Mehmet Gönen [c], Arto Klami [d], Samuel Kaski [d,e,**]

[a] Heidelberg University, HCI, Speyerer Str. 6, D-69115 Heidelberg, Germany
[b] Nokia Research Center Otaniemi, Espoo, Finland
[c] Sage Bionetworks, Seattle, WA, USA
[d] Helsinki Institute for Information Technology HIIT, Department of Computer Science, University of Helsinki, Finland
[e] Helsinki Institute for Information Technology HIIT, Department of Information and Computer Science, Aalto University,
P.O. Box 15400, FI-00076 Aalto, Finland

ARTICLE INFO

Article history:
Received 30 September 2013
Received in revised form
28 January 2014
Accepted 21 February 2014
Communicated by Christos Dimitrakakis
Available online 18 April 2014

Keywords:
Affect recognition
Machine learning
Multi-task learning
Multi-view learning

ABSTRACT

Several computational approaches have been proposed for inferring the affective state of the user, motivated for example by the goal of building improved interfaces that can adapt to the user's needs and internal state. While fairly good results have been obtained for inferring the user state under highly controlled conditions, a considerable amount of work remains to be done for learning high-quality estimates of subjective evaluations of the state in more natural conditions. In this work, we discuss how two recent machine learning concepts, multi-view learning and multi-task learning, can be adapted for user state recognition, and demonstrate them on two data collections of varying quality. Multi-view learning enables combining multiple measurement sensors in a justified way while automatically learning the importance of each sensor. Multi-task learning, in turn, tells how multiple learning tasks can be learned together to improve the accuracy. We demonstrate the use of two types of multi-task learning: learning both multiple state indicators and models for multiple users together. We also illustrate how the benefits of multi-task learning and multi-view learning can be effectively combined in a unified model by introducing a novel algorithm.

1. Introduction

Affective computing seeks to develop more efficient and pleasant user interfaces by taking into account the affective state of the user. For example, the information flow can be tailored by managing interruptions from e-mail alerts and phone calls when the user is in deep thought [7], and the affective state can be used to determine the most suitable time to intervene during a pedagogical game [8]. Apart from adapting the interface, information on the affective state can be used to gain a deeper understanding of how users and computers interact. A prerequisite of affective computing is the ability to recognize users' states of interest, either by observing the users' actions [26] or by analyzing physiological signals measured from the user [25,15,6]. In this work, we study the latter approach and discuss machine learning solutions for inferring the affective state of the user from physiological signals in unobtrusive and loosely controlled user setups.

During recent years, several databases of physiological measurements in affective computing tasks have been released [13,22,31], in an attempt to provide high-quality data for learning and benchmarking state inference models. The state of the art in the field is that the user's state can be inferred relatively accurately in highly controlled experiment setups where the stimuli evoke strong emotional responses [20,24,32]. For less controlled setups, where the ground truth labels come from user evaluations, some recent works have obtained positive results [22,2,9,11,33] but in many cases the prediction accuracies are not yet sufficiently high for practical use in adaptive interfaces.

We introduce two elements from machine learning literature to help improve the user state estimation: multi-view learning and multi-task learning. Both ideas can be incorporated into many of the current state estimation methods (for a recent review see [34]), to obtain better estimates of the user's affective states. We motivate these concepts for affective computing tasks and demonstrate their usefulness in learning user states especially when used in combination.

Multi-view learning studies how data sets having co-occurring observations can be combined. Most affective computing studies monitor the user with several sensors or sensor channels, which

[*] Corresponding author.
[**] Corresponding author at: Helsinki Institute for Information Technology HIIT, Department of Information and Computer Science, Aalto University, P.O. Box 15400, FI-00076 Aalto, Finland.

E-mail addresses: melih.kandemir@iwr.uni-heidelberg.de (M. Kandemir), akos.vetek@nokia.com (A. Vetek), mehmet.gonen@sagebase.org (M. Gönen), arto.klami@cs.helsinki.fi (A. Klami), samuel.kaski@aalto.fi (S. Kaski).

http://dx.doi.org/10.1016/j.neucom.2014.02.057
0925-2312/© 2014 Elsevier B.V. All rights reserved.

can be considered as such co-occurring sets. Multi-view learning refers to various strategies for learning a joint model over all sensor data, to learn how the sources should be combined for building optimal models. In this paper, we work with a specific multi-view learning technique called *multiple-kernel learning* (MKL) [16], which allows using multiple sensors in any kernel-based learning algorithm while automatically revealing which sensors are useful for solving the task. Even though considerable effort has been put into finding out which physiological sensors are related to which affective dimensions, this is still useful for all practical applications with specific sensor hardware. Automatically learning the sensor importance is especially useful when developing practical systems for out-of-laboratory conditions.

The other concept, *multi-task learning* (MTL), studies learning of several prediction tasks together [5]. Within the scope of state inference, MTL takes advantage of the data of other users by learning from the cross-user similarities, without assuming that the users are identical. This helps particularly when the amount of labeled training data is limited. Alternatively, learning each output label, such as arousal and valence, could be considered as a task. Learning predictive models for all of the labels together is then useful assuming that all labels are one-dimensional summaries of a more complex unknown state of the user. The approach will be particularly useful if the dimensions are not independent.

We present a novel kernel-based model that combines both multi-view and multi-task aspects. It can be applied to both of the aforementioned MTL scenarios, and it uses the MKL formulation to make the approach multi-view. We then apply the model to two different data collections to study the accuracy of state recognition. The first collection, taken from Koelstra et al. [22], is an example of a laboratory-quality data. We have collected the other data set ourselves under less constrained conditions.

The main goal of the paper is to illustrate the benefits of the two aforementioned general purpose machine learning techniques in affective computing applications. To this end, we show how combining MTL and MKL within a unified model improves the prediction performance, and also highlight how MKL automatically learns the importance of individual sensors even when solving multiple inference tasks simultaneously. We demonstrate the models with generic features instead of carefully selecting the sensors and features to match the particular affective inference tasks. This highlights the main advantage of the proposed strategy: It allows working with a wide set of sensors and tasks, without requiring much manual labor in incorporating domain-specific knowledge into the solutions.

2. Inferring the user state

Given the input data from P sensors, the user state inference task consists of inferring for each data point a set of labels that jointly characterize the state of the user. We do not assume any particular emotional model, such as [28]. Instead, we simply require the states to be represented by a collection of numerical labels. The labels do not have to be independent; in fact, as will become more apparent later, the multi-task formulation we introduce is specifically tailored to capture correlations between the labels. In the experimental section we use Likert-scale evaluations of valence, arousal, liking, and mental workload as the labels, but the underlying machine learning techniques would apply to any other numerical characterizations of the state dimensions. Even though we resort to binarization of multi-category state labels to overcome data scarcity, extension of the presented techniques to multi-class setups is straightforward.

We study *user-specific* and *user-independent* setups for each learning model. The former is trained on data recorded from a single user and assumes this person to be the eventual user of the system, whereas the latter learns the models from M earlier users and assumes the eventual user to be a new one. User-specific models need to be separately customized to target users. On the other hand, user-independent models do not require any training data from the eventual user, and hence can be pre-trained on large data collections.

For both scenarios, each data sample $\boldsymbol{x}_i$ is represented as a collection of vectors $\boldsymbol{x}_i = \{\boldsymbol{x}_i^{(m)}\}_{m=1}^{P}$, one for each of the P views (here sensors), where $\boldsymbol{x}_i^{(m)} \in \mathbb{R}^{D_m}$ and D_m is the dimensionality of the feature representation for the sensor m. The output, characterization of the user's state, is given as (here binary) vector of labels $\boldsymbol{y}_i = [y_i(1), \ldots, y_i(T)]$, where $y_i(j) \in \{\pm 1\}$ and T is the number of labels.

All learning setups considered in this paper are multi-view, due to the input data coming from P different sensors. MTL, in turn, can be applied in two different ways. When considering the different users as different but related tasks we can learn user-specific models for all users at the same time, separately for each label. In this case, each task takes as input the measurements taken from a different user $\boldsymbol{x}$, and predicts the corresponding label. Even though the models are learned together in the spirit of multi-task learning, the output will be a separate model for each user. Alternatively, we can learn a single user-independent model for all T labels at once, resulting in a MTL setup where the inputs $\boldsymbol{x}$ are the same for all tasks but the output labels are different.

In this paper, we formulate a novel kernel-based algorithm that performs multi-task and multi-view learning in a coupled and efficient manner. In Sections 2.1–2.3 we review the basics of kernel based learning and explain the earlier kernel-based multi-task and multi-view algorithms. Finally, in Section 2.4 we introduce our new model that combines both approaches.

2.1. Support vector machines (SVMs)

We take the standard support vector machine (SVM) [30] as a single-task and single-view building block on which we develop our novel multi-task multi-view learning algorithm. We denote by $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$ a sample of N independent training instances, where $\boldsymbol{x}_i$ is a D-dimensional input vector with the target output y_i, and by $\boldsymbol{\Phi} : \mathbb{R}^D \to \mathbb{R}^S$ a function that maps the input patterns to a preferably higher dimensional space. The support vector machine learns a linear discriminant that predicts the target output of an unseen test instance $\boldsymbol{x}$ as

$$f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{\Phi}(\boldsymbol{x}) + b,$$

where $\boldsymbol{w}$ contains the hyperplane parameters and b is the bias parameter. Using the representer theorem, the discriminant in the dual form becomes

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} \alpha_i \underbrace{\boldsymbol{\Phi}(\boldsymbol{x}_i)^\top \boldsymbol{\Phi}(\boldsymbol{x})}_{k(\boldsymbol{x}_i, \boldsymbol{x})} + b$$

where N is the training set size, $k : \mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}$ is the kernel function that defines a similarity metric for pairs of data instances, and $\boldsymbol{\alpha}$ is the vector of Lagrange multipliers defined in the domain

$$\mathcal{A} = \left\{ \boldsymbol{\alpha} : \sum_{i=1}^{N_r} \alpha_i = 0, \; \alpha_i \in \mathbb{R}, \; \forall i \right\}. \quad (1)$$

For binary classification $y_i \in \{-1, +1\}$ and squared loss, the corresponding objective function is

$$J(\boldsymbol{\alpha}) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \left(k(\boldsymbol{x}_i, \boldsymbol{x}_j) + \frac{\delta_i^j}{2C} \right),$$

where $\delta_i^j = 1$ if $i=j$ and 0 otherwise. In the training phase, $J(\boldsymbol{\alpha})$ is maximized with respect to $\boldsymbol{\alpha}$.

2.2. Multiple kernel learning (MKL)

A good affective computing model utilizes information from all available sensors, correctly weighting each of the sensors according to how useful it is. Instead of manually selecting only a small subset of most useful sensors, we propose to automatically infer the best sensors amongst a possibly very rich set of sensors.

Multiple kernel learning is a multi-view learning solution that automatically learns the importance of the sensors to maximize the predictive accuracy of kernel-methods (see [16] for a survey). The idea is to represent each sensor (view) m by one kernel k_m, and combine them into a single kernel k_η by using a function $f_\eta : \mathbb{R}^P \to \mathbb{R}$ parameterized by $\boldsymbol{\eta}$:

$$k_\eta(\boldsymbol{x}_i, \boldsymbol{x}_j; \boldsymbol{\eta}) = f_\eta(\{k_m(\boldsymbol{x}_i^{(m)}, \boldsymbol{x}_j^{(m)})\}_{m=1}^P; \boldsymbol{\eta}).$$

An optimal $\boldsymbol{\eta}$ is learned from data. The different multiple kernel learning models differ in the way they put restrictions on the kernel weights $\boldsymbol{\eta}$. In this paper, we take a weighted average of the kernels, with nonnegative weights that sum up to one (i.e., convex sum): $k_\eta(\boldsymbol{x}_i, \boldsymbol{x}_j; \boldsymbol{\eta}) = \sum_{m=1}^P \boldsymbol{\eta}_m k_m(\boldsymbol{x}_i^{(m)}, \boldsymbol{x}_j^{(m)})$.

When learning the kernel weights one could also consider some form of regularization for them, for example to favor sparse solutions. There has been no conclusive evidence that sparse solution would be more accurate (see [21]), and hence we learn here the weights of regular MKL without sparsity-inducing regularization.

2.3. Multi-task kernel machines

Multiple learning tasks can be solved more accurately if they are learned together, by encouraging the tasks to share knowledge by having similar parameters [3]. This idea has been employed in SVMs by merging the training instances of all tasks, and learning the following kernel function [14]:

$$\widehat{k}(\boldsymbol{x}_i, \boldsymbol{x}_j) = (1/\gamma + \delta_i^j) k(\boldsymbol{x}_i, \boldsymbol{x}_j) \tag{2}$$

where γ determines the similarity between the samples of different tasks and δ_i^j is 1 if $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are from the same task, and 0 otherwise. Intuitively, the model assumes that samples from all other tasks can also be used for learning the model but their similarity is discounted by a factor of γ. For $\gamma = 0$ the solution reduces to assuming all tasks to be identical, whereas $\gamma = \infty$ is equivalent to learning the tasks separately.

The above multi-task formulation has three disadvantages: (a) it requires all tasks to be in a common input space; (b) it requires all tasks to have the same output space to be able to capture them in a single learner, which makes it not applicable for MTL over labels (multi-output learning); and (c) it requires more time than training separate (hence small-sample) learners for each task.

2.4. Multi-task multiple kernel machines (MT-MKL)

We could obtain a multi-task multi-kernel learning method by simply extending Eq. (2) to multiple kernels:

$$\widehat{k_\eta}(\boldsymbol{x}_i, \boldsymbol{x}_j; \boldsymbol{\eta}) = (1/\gamma + \delta_i^j) k_\eta(\boldsymbol{x}_i, \boldsymbol{x}_j; \boldsymbol{\eta}),$$

and learning the weights $\boldsymbol{\eta}$ as in standard MKL. However, the aforementioned disadvantages would still apply.

We propose a novel MT-MKL model that induces similarity across tasks via kernel combination parameters $\boldsymbol{\eta}$, instead of via the discriminant function as above. It learns a different $\boldsymbol{\eta}_r$ for each task r and regularizes them globally. Assuming a single $\boldsymbol{\eta}$ common to all tasks as in Rakotomamonjy et al. [27] is then a special case of our model, which holds the risk of negative transfer if some of the tasks are only weakly correlated. Parameters of models can be learned by solving the following min–max optimization problem:

$$\underset{\{\boldsymbol{\eta}_r \in \mathcal{E}\}_{r=1}^T}{\text{minimize}} \; \underbrace{\underset{\{\boldsymbol{\alpha}_r \in \mathcal{A}_r\}_{r=1}^T}{\text{maximize}} \; \Omega(\{\boldsymbol{\eta}_r\}_{r=1}^T) + \sum_{r=1}^T J_r(\boldsymbol{\alpha}_r, \boldsymbol{\eta}_r)}_{\mathcal{O}_\eta} \tag{3}$$

where $\mathcal{E} = \{\boldsymbol{\eta} : \sum_{m=1}^P \eta_m = 1, \eta_m \geq 0 \ \forall m\}$ denotes the domain of the kernel combination parameters, $\mathcal{A}_r$ is the domain of the Lagrange multipliers for task r as in Eq. (1), and

$$J_r(\boldsymbol{\alpha}_r, \boldsymbol{\eta}_r) = \sum_{i=1}^N \alpha_i^r - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i^r \alpha_j^r y_i^r y_j^r \left(k_\eta^r(\boldsymbol{x}_i, \boldsymbol{x}_j; \boldsymbol{\eta}_r) + \frac{\delta_i^j}{2C} \right)$$

is the objective function of the kernel-based learner for task r. Similarity between the kernels is enforced by the regularization term $\Omega(\cdot)$ that makes the kernel combination parameters of different tasks related and penalizes their divergence from each other. Among many possible choices of regularizers, we illustrate two: (i) the inner-product regularizer

$$\Omega_1(\{\boldsymbol{\eta}_r\}_{r=1}^T) = -\nu \sum_{r=1}^T \sum_{s=1}^T \boldsymbol{\eta}_r^\top \boldsymbol{\eta}_s,$$

and (ii) the ℓ_2-norm regularizer

$$\Omega_2(\{\boldsymbol{\eta}_r\}_{r=1}^T) = -\nu \sum_{r=1}^T \sum_{s=1}^T \|\boldsymbol{\eta}_r - \boldsymbol{\eta}_s\|_2.$$

The first regularizer, $\Omega_1(\cdot)$, corresponds to the negative total correlation between the kernel weights of the tasks. Although this term is concave, efficient optimization is possible thanks to the bounded feasible sets of the kernel weights. The second alternative, $\Omega_2(\cdot)$, is the standard ℓ_2−norm regularizer that penalizes the distance of kernel weights in the Euclidean space.

The coefficient ν determines the influence of the regularizer on the cost function. A small ν value corresponds to assuming unrelated tasks (and with $\nu = 0$ the model reverts to an independent MKL learner for each task), whereas a large value enforces similar kernel weights across the tasks.

The min–max optimization problem in Eq. (3) can be solved using a two-step iterative algorithm in a similar way to previous work on MKL [35–37]. In the first step, kernel weights $\{\boldsymbol{\eta}_r\}_{r=1}^T$ are given, hence we have T single-task single-kernel learning problems at hand. In the second step, where single-task learners are given, we update $\{\boldsymbol{\eta}_r\}_{r=1}^T$ with respect to $\mathcal{O}_\eta$ by applying projected gradient-descent subject to two constraints on the kernel weights: (i) being positive $(\forall r, \forall m, \eta_r^m \geq 0)$ and (ii) summing up to one $(\forall r, \sum_{m=1}^P \eta_r^m = 1)$. The gradient of the joint objective function of all task learners $\mathcal{O}_\eta$ is

$$\frac{\partial \mathcal{O}_\eta}{\partial \eta_m^r} = -2 \frac{\partial \Omega(\boldsymbol{\eta}_r)}{\partial \eta_m^r} - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i^r \alpha_j^r y_i^r y_j^r \left(k_m^r(\boldsymbol{x}_i^r, \boldsymbol{x}_j^r) + \frac{\delta_i^j}{2C} \right),$$

where the gradient of the regularizer is

$$\frac{\partial \Omega_1(\boldsymbol{\eta}_r)}{\partial \eta_m^r} = -\nu \sum_{s=1}^T \eta_m^s$$

for the inner-product penalty, and

$$\frac{\partial \Omega_2(\boldsymbol{\eta}_r)}{\partial \eta_m^r} = -\nu \sum_{s=1}^T 2(\eta_m^r - \eta_m^s)$$

for the ℓ_2−norm penalty. For faster convergence, step sizes of the gradient-descent can be tuned at each iteration by line search. The

Table 1
List of features extracted from the DEAP data set, which is a subset of the list given in Koelstra et al. [22].

Full-scalp EEG from 32 channels: Spectral powers of theta (4–8) Hz, slow alpha (8–10) Hz, alpha (10–12) Hz, beta (12–30) Hz, and gamma (30+) Hz bands for each electrode

EOG (Electro-oculogram) and *EMG (Electro-myogram):* Energy, mean and variance

GSR (Galvanic Skin Response): Mean, mean of the derivative, mean of the positive derivatives, proportion of negatives in the derivative, number of local minima, and 10 spectral powers within 0–2.4 Hz

Respiration: Band energy ratio, average respiration signal, mean of the derivative, standard deviation, range of greatest breath, 10 spectral powers within 0–2.4 Hz, and average and median peak to peak time

Plethysmograph: Average and standard deviation of Heart Rate Variability (HRV) and interbeat intervals, energy ratio between 0.04–0.15 Hz and 0.15–0.5 Hz, spectral power in 0.1–0.2 Hz, 0.2–0.3 Hz, 0.3–0.4 Hz, 0.01–0.08 Hz 0.08–0.15 Hz, and 0.15–0.5 Hz components of HRV

Skin temperature: Mean, mean of the derivative, and spectral power in 0–0.1 Hz and 0.1–0.2 Hz

iterations are then repeated until convergence. The proposed method can be summarized as in Algorithm 1. See Gonen et al. [17] for the empirical performance of the method on tasks other than affective state inference.

Algorithm 1. The proposed Multitask Multiple Kernel Learning (MT-MKL) algorithm.

```
Initialize $\boldsymbol{\eta}_r$ as $(1/P, \ldots, 1/P),\ \forall r$
repeat
    Calculate $\boldsymbol{K}_\eta^r = k_\eta^r(\boldsymbol{x}_i, \boldsymbol{x}_j)_{i,j}^{N_r},\ \forall r$
    Solve a single-kernel machine using $\boldsymbol{K}_\eta^r,\ \forall r$
    Update $\boldsymbol{\eta}_r$ in the direction of $-\partial\mathcal{O}_\eta/\partial\boldsymbol{\eta}_r,\ \forall r$
until convergence
```

3. Tasks, setups, and measures

We demonstrate off-line analysis with the kernel-based inference models in two different application scenarios. The first uses high-quality data from Koelstra et al. [22], and acts as an example of how good models can be learned when the stimuli are relatively carefully chosen, the user is monitored with an extensive set of sensors, and the labeling has been done with care. We then make a step towards a setup that would be closer to what could be used for practical affective interfaces, using a smaller set of relatively unobtrusive sensors and letting computer scientists that are not experts in psychological experiments, such as ourselves, design the data collection and labeling schemes.

For assessing model performance, we use

- *accuracy*: The proportion of correct predictions,
- *AUC*: Area under receiver operating characteristics curve, and
- *macro-F_1 score*: The average of the harmonic mean of precision and recall over all output categories.

For user-specific models we compute the leave-one-sample-out estimate, learning N different models using $N-1$ data points for training and evaluating with the left-out sample. For user-independent models we use a leave-one-user-out procedure, learning M different models using all the data from $M-1$ users and testing with the left-out user. For both setups, we compare the performance of three kernel-based learners: SVM, MKL, and MT-MKL. For MT-MKL, we consider the following four alternatives:

- *MT-MKL (U1)*: Users are taken as tasks and $\Omega_1(\cdot)$ is used for kernel weight regularization.
- *MT-MKL (U2)*: Users are taken as tasks and $\Omega_2(\cdot)$ is used for kernel weight regularization.
- *MT-MKL (L1)*: Label categories are taken as tasks and $\Omega_1(\cdot)$ is used for kernel weight regularization.
- *MT-MKL (L2)*: Label categories are taken as tasks and $\Omega_2(\cdot)$ is used for kernel weight regularization.

For MT-MKL (L1) and MT-MKL (L2) we evaluate both user-specific and user-independent learning setups, but for MT-MKL(U1) and MT-MKL(U2) only the user-specific setup is applicable.

It would also be possible to consider multi-task learning over both the users and the label categories, so that each user+label pair would form a single task. However, such tasks would not be exchangeable but instead the structure between the tasks should be taken into account in the learner; for instance, the tasks corresponding to the same user should be regularized more towards each other than the tasks corresponding to different users. Hence, we do not consider such a setup further in this paper, but instead focus on the setups where all tasks *a priori* equally related to each other.

We picked the hyperparameters C and ν by cross-validation. The C was selected from the set $\{10^{-3}, 10^{-2}, \ldots, 10^{+3}\}$ for all models. For MT-MKL variants, the regularization parameter ν was picked from the set $\{10^{-4}, 10^{-3}, \ldots, 10^{+4}\}$. We used the baseline method SVM to choose either linear or Gaussian kernel, using the same choice for all MKL methods as well. For both cases, the kernels were normalized to make the MKL weights more easily interpretable.

4. Experiment 1: high-quality laboratory data

The first data set, named by the authors as DEAP, is taken from Koelstra et al. [22]. In the experiment, 32 healthy participants watched 40 music videos of 1 min each and self-reported their emotional response to each video in four dimensions: valence, arousal, dominance, and liking (where liking refers to whether the user liked the video). The original label scales (from 1 to 9) were binarized by thresholding at level 5. The subjects were monitored through measurements with an extensive set of sensors, including full-scalp EEG and six peripheral sensors. We extracted 216 features from the measurements for each video (see Table 1), a subset of the features used by Koelstra et al. [22]. We also utilized a dimensionality reduction procedure similar to Koelstra et al. [22]. We computed linear discriminant analysis (LDA) on the training data for each label separately, then selected the top 25% of features for each sensor, ranking them by the eigenvalue in the LDA solution.

4.1. Prediction performance

The user-specific learning setup is the same as the one used in Koelstra et al. [22]; hence, we are able to compare the performance of our methods also with the naive Bayes model used there. In particular, we compare our results against their best variant using only physiological signals as inputs (they got better results when incorporating also content-based features, which would not generalize to any other type of content). We also present the

Table 2

Test accuracy, AUC, and macro-F_1 score of the models on the DEAP data set. The top table shows the results for the *user-specific* setup and the bottom table for the *user-independent* setup. The value of the best performing model (not counting baselines) has been boldfaced in each column. 'Random' and 'Majority' are baselines, SVM is a traditional kernel-based learner, MKL denotes a multi-view SVM, and the MT-MKL are multi-task multi-view learners. For the user-specific setup the third row shows the best results reported in Koelstra et al. [22, Table 7]. MT-MKL(L1) and MT-MKL(U1) correspond to multi-tasking over labels and users using the regularizer $\Omega_1(\cdot)$, respectively. MT-MKL(L2) and MT-MKL(U2) denote the same but uses the regularizer $\Omega_2(\cdot)$.

Models	Valence			Arousal			Liking			Average		
	Acc.	AUC	F1	Acc.	AUC	F1	Acc.	AUC	F1	Acc.	AUC	F1
Random	0.50	0.52	0.49	0.45	0.50	0.45	0.42	0.50	0.42	0.46	0.51	0.45
Majority	0.51	0.50	0.32	0.58	0.50	0.35	0.65	0.50	0.39	0.58	0.50	0.35
DEAP	0.63	N/A	0.61	0.62	N/A	0.58	0.59	N/A	0.54	0.61	N/A	0.58
SVM	0.64†	0.60†	0.62†	0.61	0.53	0.53†	**0.65**	**0.61**†	**0.57**†	0.63†	0.58†	0.57†
MKL	0.63†	0.60†	0.60†	0.61	0.56†	0.54†	0.64	0.54	0.53†	0.63†	0.56†	0.56†
MT-MKL(U1)	**0.66**†	**0.64**†	**0.63**†	0.63	0.58†	**0.57**†	0.64	0.56†	0.55†	0.64†	**0.59**†	**0.58**†
MT- MKL(U2)	0.62†	0.64	0.53†	0.61	**0.67**	0.57†	0.64	0.60	0.52†	0.63†	0.64†	0.54†
MT-MKL(L1)	0.65†	0.64†	0.61†	**0.65**	0.55	0.57†	0.65	0.53	0.56†	**0.65**†	0.57†	0.58†
MT-MKL(L2)	0.63†	0.61†	0.58†	0.63	0.52	0.51†	0.65	0.52	0.51†	0.64†	0.55†	0.53†
Random	0.49	0.48	0.48	0.52	0.49	0.48	0.55	0.48	0.48	0.52	0.48	0.48
Majority	0.57	0.50	0.28	0.59	0.50	0.29	0.67	0.50	0.33	0.61	0.50	0.30
SVM	0.57	0.58†	0.55†	0.56†	**0.56**†	0.51†	**0.67**	**0.54**†	0.45†	0.60	**0.56**†	0.50†
MKL	0.59	**0.61**†	0.55†	0.56†	0.54†	**0.53**†	0.66†	0.48	0.51†	0.60	0.54†	**0.53**†
MT-MKL(L1)	0.59	0.60†	0.55†	**0.58**	0.55†	0.52†	0.66	0.50	**0.51**†	**0.61**	0.55†	0.53†
MT-MKL(L2)	**0.60**†	0.60†	**0.56**†	0.56	0.54	0.46†	0.65	0.51	0.42†	0.60	0.55†	0.48†

† Significantly above majority voting (paired t-test, $p < 0.05$). Not calculated for DEAP since performance scores for individual cross-validation trials are not publicly available.

baseline results of majority voting (choosing the label that is most frequent in the training data[1]) and random guessing according to the relative frequency of the labels in training data.

We performed our analysis on the same three emotional dimensions as Koelstra et al. [22]: valence, arousal, and liking. Average (over the users) test accuracies, AUC, and macro-F_1 scores of our method and the baselines are given in Table 2 (top). Multi-tasking in either way (over labels, or users) using the inner-product regularizer brings decent improvement over simpler models for all labels except liking. MT-MKL(U1) and MT-MKL(L1) either outperform or are tied with the naive Bayesian model introduced by Koelstra et al. [22]. While MT-MKL(U2) gives comparable results to MT-MKL(U1), the ℓ_2-norm regularizer performs worse for multi-tasking over labels (MT-MKL(L2)).

We evaluated our methods also on the user-independent setup for completeness, even though the authors of Koelstra et al. [22] avoided this setup due to high inter-user variation in their data. Table 2 (bottom) reveals that the accuracy is lower than in the user-specific case, as was expected. Nevertheless, the relative performance of the models is roughly retained, and we still outperform the chance level.

4.2. Sensor importance

An advantage of MKL is that it gives a direct estimate of sensor importance in the form of the kernel weights $\boldsymbol{\eta}$. It is particularly useful for relative ranking of the sensors.

Fig. 1(a) shows the kernel weights found by MT-MKL(U1) for arousal, averaged over the users. EEG is the dominant important sensor, which is sensible considering that a 32-channel sensor is much more data-rich than the other singular sensors. The result is consistent with Koelstra et al. [22] who obtained better results with EEG than they did with all peripheral sensors combined. GSR and respiration sensors are the two most informative peripheral sensors, supporting previous studies such as Alzoubi et al. [1] and Gunes et al. [18]. It is noteworthy that this is an automatic side result of the method which required no extra effort from the experimenter.

Fig. 1(b) shows the weights for individual users with $\nu = 0$ (the regular MKL model) and Fig. 1(c) shows the weights obtained with the multi-task version that chooses the optimal regularization. We see that the multi-task learning solution makes the weights more similar, regularizing the individual solutions learned from limited data, but that it still allows the models for some users to rely more on GSR that is useful for those particular users. The earlier multi-task solution by Rakotomamonjy et al. [27] would force those users to comply with the consensus.

5. Experiment 2: towards real-world usage

In this second example, we took a step towards the kind of data available in real-world applications. We designed an experiment with simpler sensors and with fairly low degree of control for the naturalistic stimulus, but still performed the experiments off-line in a controlled environment.

5.1. Experimental setup

We constructed an experiment where users performed a pre-specified set of tasks that reasonably resemble typical tasks of daily computer use. The tasks were presented as a series of HTML pages, and the users interacted with the system using a mouse and a keyboard. A typical page in the experiment showed a question or puzzle the user was asked to answer, inducing typical processes such as decision-making and problem solving. Submitting the answer took the user to the next page. The experimental setup and the web-interface were designed from a user-centric perspective. To this end, we interviewed with three pilot users, and adjusted the setup based on the findings.

5.1.1. Measurements

We collected data from six healthy male university students with four devices (see Fig. 2): accelerometer, heart rate belt, eye tracker, and electroencephalograph (EEG). A 3D acceleration vector was measured from the nape of the user at 15 Hz frequency. The

[1] Note that Koelstra et al. [22] defined majority voting as the most frequent label in the whole data. This would not correspond to a valid classifier, since it uses test data.

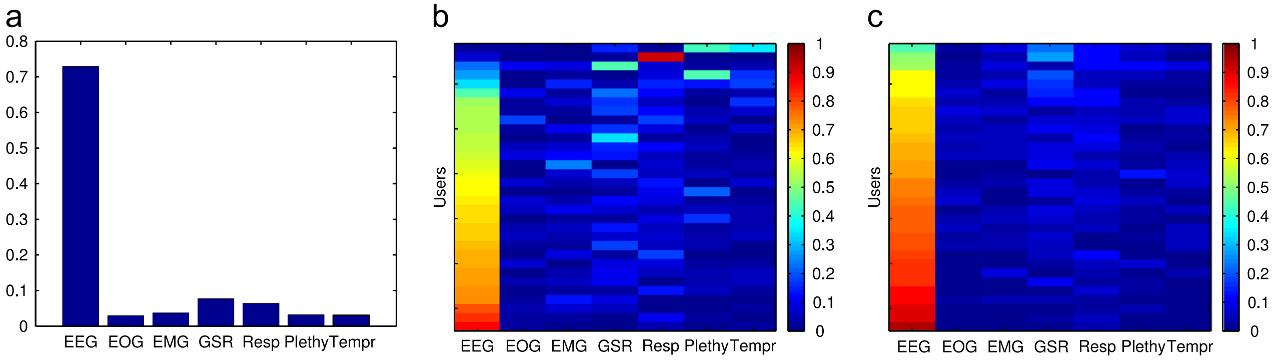


Fig. 1. (a) Average (over the users) kernel weights found by MT-MKL(U1) for inferring arousal, showing that EEG is clearly the most useful sensor. (b) Sensor weights found by MKL for each individual user sorted by the weight of the EEG sensor. (c) Sensor weights found by MT-MKL(U1). Weight increases as the color goes from blue through yellow to red. This figure is best viewed in colors. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

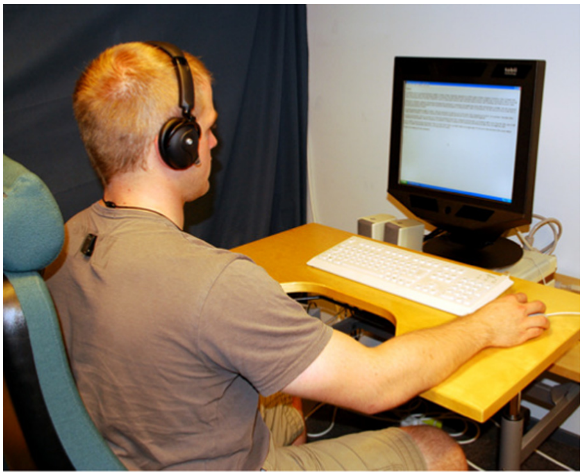
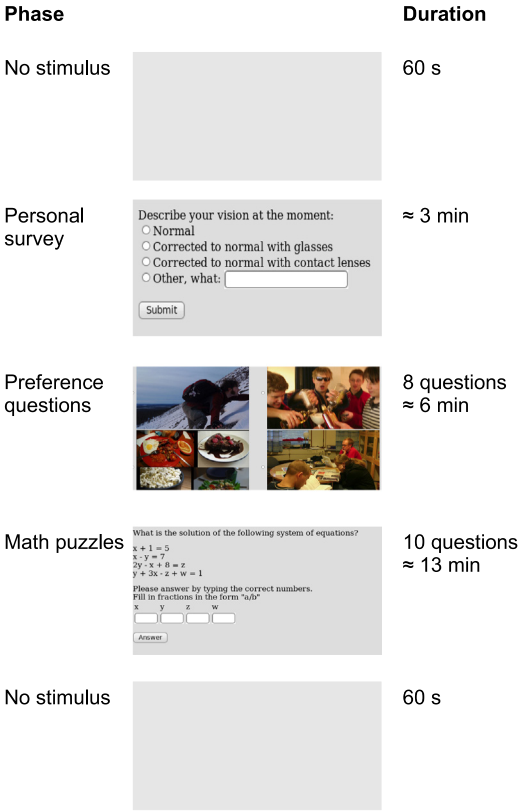


Fig. 2. A test user wearing the sensors. The headset is a one-channel EEG device, the eye-tracker is integrated in the desktop monitor, and the accelerometer can be seen attached to the nape of the user. A heart rate belt is under the shirt.

heart rate belt recorded RR-intervals (the time between two consecutive R waves in the electrocardiogram (ECG)) at 2 Hz frequency. The eye tracker followed the pupil diameter with an infrared camera attached to a PC monitor at 50 Hz frequency. The EEG device measured one-channel EEG from the FP1 location of the International 10–20 system at 512 Hz frequency.

5.1.2. *Interface and user tasks*

The experimental setup consisted of five different phases, as summarized in Fig. 3. The first and last phases were baseline measurements, where the participant was presented with no stimulus and was instructed to relax and sit still. In the second phase, the subject filled in a background survey which included open answer and multiple choice questions about age, gender and language proficiency. The third part contained eight multiple choice preference questions, where the choices were presented as four images. The fourth phase consisted of 10 arithmetic and logic puzzles of increasing difficulty, designed to elicit mental workload. After each puzzle, the user was given feedback on whether his answer was correct. During the experiment, unexpected events and interruptions such as simulated failures in submitting forms and incorrect performance feedback were inserted to evoke frustration and arousal.



Describe your vision at the moment:
○ Normal
○ Corrected to normal with glasses
○ Corrected to normal with contact lenses
○ Other, what:

What is the solution of the following system of equations?

x + 1 = 5
x - y = 7
2y - x + 8 = z
y + 3x - z + w = 1

Please answer by typing the correct numbers.
Fill in fractions in the form "a/b"

x y z w

Fig. 3. A flow diagram of the experiment, showing sample screenshots of the user interface. The experiment lasted 25 min on average, including transitions between the phases.

5.1.3. *Labeling affective states and mental workload*

We obtained the ground-truth state labels from a 7-point numerical scale. The scale is a simplified version of the Self Assessment Manikin [4] for arousal and valence, and corresponds to one-dimensional Mental Load sub-scale of NASA's Task Load Index (NASA TLX) [19] for mental workload. The labels were collected by self-evaluation, similar to D'Mello and Graesser [12]. The user was shown each page again immediately after the experiment, this time including three sets of radio button selectors, one for each label.

We analyzed this data set as similar as possible to the DEAP data set to keep the outcomes comparable. We extracted one data point of 38 features (8 EEG and 30 peripheral) from the time period of each question/puzzle (see Table 3), and formed the views

Table 3
List of features for the second experiment.

3D body motion (calculated separately for each dimension), and pupil diameter: Mean and standard deviation, mean of the derivative, mean, median, and maximum peak-to-peak interval, standard deviation of fixation duration

EEG: Spectral power in 0.5–2.75 Hz, 3.5–6.75 Hz, 7.5–9.20 Hz, 10.0–11.75 Hz, 13.0–16.75 Hz, 18.0–29.75 Hz, 31.0–39.75 Hz, and 41.0–49.75 Hz

ECG: Mean and standard deviation of the HRV, energy ratio between 0.04–0.15 Hz and 0.15–0.5 Hz, spectral powers in 0.1–0.2 Hz, 0.2–0.3 Hz, 0.3–0.4 Hz, 0.01–0.08 Hz, 0.08–0.15 Hz, and 0.15–0.5 Hz components of HRV

Table 4
Test accuracy, AUC, and macro-F_1 score of the models for Experiment 2. The top table shows the results for the user-specific setup and the bottom table for the user-independent setup. The value of the best performing model in each column has been boldfaced. See Table 2 for explanations of the methods.

Models	**Valence**			**Arousal**			**Mental Wkld**			**Average**		
	Acc.	**AUC**	**F1**	**Acc.**	**AUC**	**F1**	**Acc.**	**AUC**	**F1**	**Acc.**	**AUC**	**F1**
Random	0.51	0.51	0.50	0.47	0.47	0.46	0.52	0.47	0.46	0.50	0.49	0.47
Majority	0.58	0.50	0.29	0.47	0.50	0.35	0.73	0.50	0.37	0.67	0.50	0.33
SVM	0.63	0.65	0.58†	0.69	0.54	0.48†	0.75	0.74†	0.58†	0.69	0.64†	0.55†
MKL	0.62	0.66†	0.58†	0.69	0.45	0.46	0.78	0.78†	**0.68**†	0.70	0.63†	0.58†
MT-MKL(L1)	**0.67**	**0.70**†	**0.64**†	0.62	**0.57**	**0.53**†	**0.79**	**0.79**†	0.64†	0.69	**0.69**†	**0.60**†
MT-MKL(L2)	0.64	0.65†	0.60†	0.63	0.51	0.54	0.77	0.77†	0.65†	0.68	0.65†	0.60†
MT-MKL(U1)	0.61	0.66†	0.58†	**0.70**	0.49	0.51	0.77	0.79†	0.66†	0.69	0.65†	0.58†
MT-MKL(U2)	0.63	0.65†	0.60†	0.69	0.48	0.46	0.77	0.77†	0.65†	**0.70**	0.63†	0.57†
Random	0.46	0.46	0.45	0.48	0.48	0.47	0.52	0.52	0.48	0.49	0.49	0.47
Majority	0.55	0.50	0.27	0.48	0.50	0.33	0.58	0.50	0.29	0.60	0.50	0.30
SVM	0.53	0.50	0.52†	0.65	0.65†	**0.49**†	0.53	0.63	0.49†	0.57	0.59†	0.50†
MKL	0.54	0.53	0.52†	0.68	0.63†	0.40†	0.54	0.70†	0.50†	0.59	0.62†	0.47†
MT-MKL(L1)	**0.60**	**0.58**	**0.58**†	**0.69**	**0.65**†	0.40†	0.64	**0.76**†	0.59†	**0.65**	**0.66**†	0.52†
MT-MKL(L2)	0.58	0.57	0.55†	0.67	0.61†	0.42†	**0.66**	0.76†	**0.62**†	0.64	0.65†	**0.53**†

† Significantly above majority voting (paired t-test, $p < 0.05$).

by grouping features according to the sensors they come from. As in the previous experiment, we binarized the output labels. We infer low vs high level using the mid-point as the discretization threshold.

5.2. Prediction performance

Table 4 shows the accuracies, AUC, and F_1 scores of all models and baselines. For the user-specific setup (top) MT-MKL(L1), the multi-task solution over labels, outperforms the other models in majority of the performance metrics. The fact that MT-MKL(L1) is better than MT-MKL(U1) could be due to that inter-subject variance is too large to benefit from information transfer across users given only 33 samples per user. Regularizing the kernel weights with $\Omega_1(\cdot)$ yields marginally better performance than with $\Omega_2(\cdot)$. Standard SVM performs fairly well, since it is less likely to overfit on small data sets compared to the more complex alternatives. For the user-independent setup (bottom) the results are similar; MT-MKL variants outperform the rest in general, whereas standard SVM is good for arousal. The choice of the kernel weight regularizer does not have a significant effect on performance. Again the accuracy of all methods is, on average, lower than in the user-specific case.

5.3. Sensor importance

The kernel weights of MT-MKL(U1), averaged over users for each task, are given in Fig. 4. Body motion and pupil diameter sensors have higher contribution to affect inference than EEG and ECG, supporting previous work [10,29]. The result provides further evidence towards using them in future real-world applications, especially as both are relatively unobtrusive. Another intuitive result is that the single-channel EEG is far less useful than the full-scalp EEG used in the first experiment.

To further illustrate how MKL automatically infers the sensor importance we conducted a semi-artificial study where we complemented the four real sensor streams with artificial noise sensors. The weights given for the real sensors, the ones conveying information on the user state, should then be large while the weights for the noise sensors should be driven towards zero. We created the noise sensors by randomly shuffling the indices of the actual sensor data, in order to break the correlation with the output labels while still retaining the nature of each sensor data. We compare the total weight MKL gives for the true sensors with the alternative approach that directly assigns the sensor weights based on averaged feature-weights of linear regression (implemented as Bayesian ℓ_1-regularized regression [23]). Irrespective of the norm used for averaging the weights, the MKL solutions are superior especially for a high number of noisy sensors, as demonstrated in Fig. 5.

6. Computational time

For practical application of affective computing models the computational time is also important. All of the models discussed in this paper are reasonably fast to train, especially compared to the time it takes to collect the sensor data, and after the training the time needed for making the predictions is negligible. Hence, all would be practically feasible for affective computing systems.

Table 5 reports average training durations per unit learning task. For multitask methods we divide the durations by the number of tasks they jointly learn, to provide a fair comparison to single-task methods. These durations include the time taken by the cross-validation procedure needed for choosing the hyperparameters. The

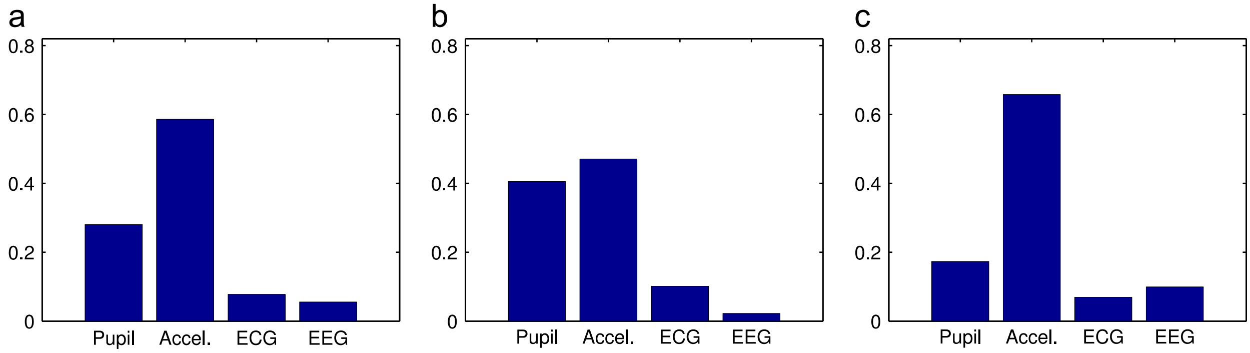


Fig. 4. Sensor weights on the second experiment, averaged over the users for the MT-MKL(U1) model, reveal that body motion (acceleration) is the most important sensor, followed by pupil measurements. (a) Valence. (b) Arousal. (c) Mental Wkld.

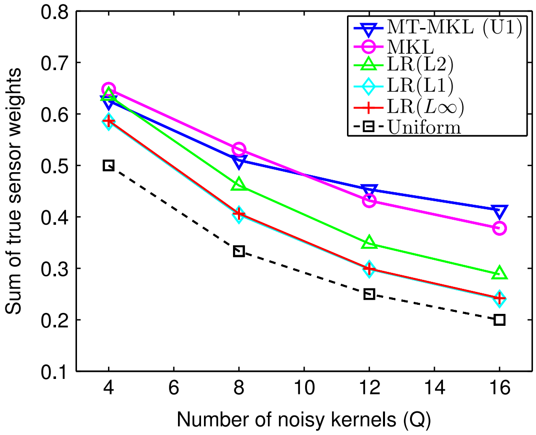


Fig. 5. The relative importance assigned for the four true sensors when learning the model with Q noise sensors not associated with the affective labels. Both MT-MKL(U1) and MKL assign much higher weight for the true sensors than the alternative method estimating the sensor weights by averaging linear regression weights, irrespective of the norm (ℓ_1, ℓ_2, or ℓ_∞) used for regularizing the model. The difference is particularly clear for large Q and statistically significant (paired t-test, $p < 0.05$) for $Q \geq 8$. The black dashed line shows the chance level of assigning equal weight to each sensor.

Table 5
Average training durations of the algorithms in comparison per unit learning task in seconds. MT-MKL variants are approximately 3.5 times slower than MKL.

	Experiment 1		Experiment 2	
	User-specific	User-independent	User-specific	User-independent
SVM	0.1	0.9	0.04	0.5
MKL	0.3	14.9	0.24	17.1
MT-MKL(L1)	1.6	93.0	1.29	10.9
MT-MKL(L2)	1.7	90.4	1.22	9.5
MT-MKL(U1)	2.5	N/A	1.26	N/A
MT-MKL(U2)	2.4	N/A	1.28	N/A

MT-MKL methods are generally the slowest because they need to validate over a two-dimensional grid to pick not only C but also the ν parameter.

7. Discussion

In this study, we investigated the benefits of multi-task and multi-view learning for pattern classification problems of affective computing and human–computer interaction. We believe that these concepts fit naturally to the needs of typical affective state recognition setups, especially when used together. We exemplified the concepts by introducing a new kernel-based learning model that combines the two aspects.

Multi-view learning tells how data coming from different sensors should be combined. The MKL technique used in this paper allows automatically learning the importance of individual sensors (or sensor channels), which simplifies the development of robust inference solutions with novel hardware. *Multi-task learning*, in turn, exploits the correlations between multiple state labels while learning the models. It is also useful in case where data are scarce which is a common problem in user-specific modeling setups. Our new model combines both aspects, by mutually regularizing the kernel weights of multiple tasks towards each other.

The primary empirical result of the paper is that the MKL strategies automatically reveal the importance of the sensors, providing intuitive ranking for the sensors in both experiments. We also showed in an artificially constructed example that the MKL strategies are more efficient in ignoring faulty or noisy sensors compared to inferring the importance from a linear regression model. In terms of accuracy, the proposed computational methods are sufficient for inferring the state labels better than chance, but we were not able to demonstrate statistically significant gain compared to the Naive Bayes and SVM, both of which are accurate classifiers for these kinds of setups. The experiments still suggest that a reliable gain could be demonstrated under more extensive testing: The MT-MKL variants give the best accuracy, AUC, and macro-F_1 scores averaged over all of the results.

Among the two alternatives considered for regularizing the kernel weights of learning tasks, the inner-product regularizer $\Omega_1(\cdot)$ was observed to provide marginally more stable performance than $\Omega_2(\cdot)$. A possible reason for this outcome could be that being a first-order term, $\Omega_1(\cdot)$ acts as a stronger regularizer than the second-order $\Omega_2(\cdot)$. This induces a stronger bias to the learner, making it less sensitive to high noise levels, which is typical to affective computing data sets including the ones presented above.

An interesting future direction is to consider real online inference of user states; for real-world use automatic selection of sensor importance is even more critical as the sensors may not work in all conditions, and it is also possible to apply multi-task learning over more diverse setups, for example considering different contexts as tasks. The methods proposed here are computationally light in the inference stage, and the training algorithms are also fairly effective and could possibly be extended for real-time learning as well.

Acknowledgments

We acknowledge support from Nokia Research Center, Academy of Finland (project number 133818 and the Finnish Centre of Excellence in Computational Inference Research (COIN)), and PASCAL2 European Network of Excellence. We gratefully thank Dr. Ville Ojanen, Dr. Jari Kangas, and Maija Nevala, MSc, for their help in designing the experiment and discussing the modeling aspects. Our special thanks go to Maija Nevala for her help with implementing the experimental setup.

References

[1] O. Alzoubi, R.A. Calvo, R.H. Stevens, Classification of EEG for affect recognition: an adaptive approach, in: Proceedings of 22nd Australasian Joint Conference on Advances in Artificial Intelligence, 2009, pp. 52–61.

[2] I. Arroyo, D.G. Cooper, W. Burleson, B.P. Woolf, K. Muldner, R. Christopherson, Emotion sensors go to school, in: Proceedings of Conference on Artificial Intelligence in Education, IOS Press, Amsterdam, The Netherlands, 2009, pp. 17–24.

[3] J. Baxter, A Bayesian/information theoretic model of learning to learn via multiple task sampling, Mach. Learn. 28 (1) (1997) 7–39.

[4] M.M. Bradley, P.J. Lang, Measuring emotion: the self-assessment manikin and the semantic differential, J. Behav. Ther. Exp. Psychiatr. 25 (1) (1994) 49–59.

[5] R. Caruana, Multitask learning, Mach. Learn. 28 (1) (1997) 41–75.

[6] G. Chanel, J.J.M. Kierkels, M. Soleymani, T. Pun, Short-term emotion assessment in a recall paradigm, Int. J. Human–Comput. Stud. 67 (8) (2009) 607–627.

[7] D. Chen, R. Vertegaal, Using mental load for managing interruptions in physiologically attentive user interfaces, in: Extended Abstracts on Human Factors in Computing Systems, 2004, pp. 1513–1516.

[8] C. Conati, H. Maclaren, Empirically building and evaluating a probabilistic model of user affect, User Model. User-Adapt. Interact. 19 (2009) 267–303.

[9] C. Conati, H. Maclaren, Modeling user affect from causes and effects, in: Proceedings of International Conference on User Modeling, Adaptation, and Personalization, UMAP '09, Springer-Verlag, Berlin, Heidelberg, 2009, pp. 4–15.

[10] C. Conati, C. Merten, Eye-tracking for user modeling in exploratory learning environments: an empirical evaluation, Knowl.-Based Syst. 20 (6) (2007) 557–574.

[11] S. D'Mello, A. Graesser, Mind and body: dialogue and posture for affect detection in learning environments, in: Proceedings of Conference on Artificial Intelligence in Education: Building Technology Rich Learning Contexts That Work, IOS Press, Amsterdam, The Netherlands, 2007, pp. 161–168.

[12] S. D'Mello, A. Graesser, Automatic detection of learner's affect from gross body language, Appl. Artif. Intell. 23 (2) (2009) 123–150.

[13] E. Douglas-Cowie, R. Cowie, I. Sneddon, C. Cox, O. Lowry, M. Mcrorie, J.-C. Martin, L. Devillers, S. Abrilian, A. Batliner, N. Amir, K. Karpouzis, The humaine database: addressing the collection and annotation of naturalistic and induced emotional data, in: Proceedings of 2nd International Conference on Affective Computing and Intelligent Interaction, 2007, pp. 488–500.

[14] T. Evgeniou, M. Pontil, Regularized multi-task learning, in: Proceedings of International Conference on Knowledge Discovery and Data Mining, ACM, New York, USA, 2004, pp. 109–117.

[15] A. Girouard, E. Solovey, L. Hirshfield, K. Chauncey, A. Sassaroli, S. Fantini, R. Jacob, Distinguishing difficulty levels with non-invasive brain activity measurements, in: Proceedings of 12th IFIP International Conference on Human–Computer Interaction: Part I, 2009, pp. 440–452.

[16] M. Gönen, E. Alpaydın, Multiple kernel learning algorithms, J. Mach. Learn. Res. 12 (2011) 2211–2268.

[17] M. Gönen, M. Kandemir, S. Kaski, Multitask learning using regularized multiple kernel learning, in: Proceedings of 18th International Conference on Neural Information Processing (ICONIP), Lecture Notes in Computer Science, 2011, pp. 500–509.

[18] H. Gunes, B. Schuller, M. Pantic, R. Cowie, Emotion representation, analysis and synthesis in continuous space: a survey, in: Proceedings of IEEE International Conference on Automatic Face Gesture Recognition and Workshops, 2011, pp. 827–834.

[19] S.G. Hart, L.E. Stavenland, Development of NASA-TLX (Task Load Index): results of empirical and theoretical research, in: Human Mental Workload, Elsevier, Amsterdam, The Netherlands, 1988, pp. 139–183.

[20] J. Kim, E. André, Emotion recognition based on physiological changes in music listening, IEEE Trans. Pattern Anal. Mach. Intell. 30 (12) (2008) 2067–2083.

[21] M. Kloft, U. Brefeld, Sonnenburg, Soren, A. Zien, Lp-norm multiple kernel learning, J. Mach. Learn. Res. 12 (3) (2011) 953–997.

[22] S. Koelstra, C. Mühl, M. Soleymani, A. Yazdani, J.-S. Lee, T. Ebrahimi, T. Pun, A. Nijholt, I. Patras, DEAP: a database for emotion analysis using physiological signals, IEEE Trans. Affect. Comput., 2014, in press.

[23] K. Murphy, Machine Learning: A Probabilistic Perspective, MIT Press, Cambridge, MA, USA, 2012.

[24] R. Picard, E. Vyzas, J. Healey, Toward machine emotional intelligence: analysis of affective physiological state, IEEE Trans. Pattern Anal. Mach. Intell. 23 (10) (2001) 1175–1191.

[25] R.W. Picard, Affective Computing, MIT Press, Cambridge, MA, USA, 1997.

[26] A. Piolat, T. Olive, J. Roussey, O. Thunin, J. Ziegler, SCRIPTKELL: a tool for measuring cognitive effort and time processing in writing and other complex cognitive activities, Behav. Res. Methods 31 (1) (1999) 113–121.

[27] A. Rakotomamonjy, R. Flamary, G. Gasso, S. Canu, $\ell_p - \ell_q$ penalty for sparse linear and sparse multiple kernel multi-task learning, IEEE Trans. Neural Netw. 22 (8) (2011) 1307–1320.

[28] J.A. Russell, A circumplex model of affect, J. Personal. Soc. Psychol. 39 (6) (1980) 1161–1178.

[29] N. Savva, N. Bianchi-Berthouze, Automatic recognition of affective body movement in a video game scenario, in: International Conference on Intelligent Technologies for Interactive Entertainment, 2011, pp. 149–158.

[30] B. Scholkopf, A.J. Smola, Learning with Kernels: Support Vector Machines, Regularization Optimization, and Beyond, MIT Press, Cambridge, MA, USA, 2002.

[31] M. Soleymani, J. Lichtenauer, T. Pun, M. Pantic, A multi-modal affective database for affect recognition and implicit tagging, IEEE Trans. Affect. Comput. 3 (1) (2011) 42–55.

[32] J. Wagner, J. Kim, E. Andre, From physiological signals to emotions: implementing and comparing selected methods for feature extraction and classification, in: Proceedings of IEEE International Conference on Multimedia and Expo, 2005, pp. 940–943.

[33] A. Yazdani, J.-S. Lee, J.-M. Vesin, T. Ebrahimi, Affect recognition based on physiological changes during the watching of music videos, ACM Trans. Interact. Intell. Syst. 2 (1) (2012) 7:1–7:26.

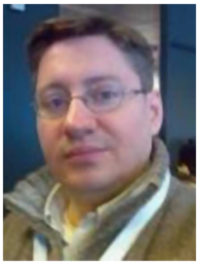
[34] Z. Zeng, M. Pantic, G. Roisman, T. Huang, A survey of affect recognition methods: audio, visual, and spontaneous expressions, IEEE Trans. Pattern Anal. Mach. Intell. 31 (1) (2009) 39–58.

[35] A. Rakotomamonjy, F. Bach, S. Canu, Y. Grandvalet, et al., SimpleMKL, J. Mach. Learn. Res. 9 (2008) 2491–2521.

[36] M. Varma, B.R. Babu, More generality in efficient multiple kernel learning, in: Proceedings of International Conference on Machine Learning (ICML), ACM, New York, USA, 2009, pp. 1065–1072.

[37] Zenglin Xu, Rong Jin, Haiqin Yang, Irwin King, Michael R. Lyu, Simple and efficient multiple kernel learning by group lasso, in: Proceedings of International Conference on Machine Learning (ICML), 2010, pp. 1175–1182.

Melih Kandemir received his B.Sc. and M.Sc. degrees in computer engineering from Hacettepe University, Ankara, Turkey, in 2005 and Bilkent University, Ankara, Turkey, in 2008, respectively. He joined the Statistical Machine Learning and Bioinformatics research group of Aalto University School of Science, Espoo, Finland, in 2008 and earned his Ph.D. degree in 2013. Since 2013, he is with Heidelberg Collaboratory for Image Processing (HCI), Heidelberg University, Heidelberg, Germany. Bayesian modeling, weakly supervised learning, medical image analysis, digital pathology, and neuroinformatics are among his research interests.



Akos Vetek is a Principal Researcher at the Media Technologies Laboratory of Nokia Research Center. His research interests include multimodal interaction, intelligent user interfaces, sensors, and wearables.

Mehmet Gönen received the B.Sc. degree in industrial engineering, the M.Sc. and the Ph.D. degrees in computer engineering from Bogaziçi University, Istanbul, Turkey, in 2003, 2005, and 2010, respectively. He did his postdoctoral work at the Helsinki Institute for Information Technology HIIT, Department of Information and Computer Science, Aalto University, Espoo, Finland. He is currently a Senior Research Scientist at Sage Bionetworks, Seattle, WA, USA. His research interests include support vector machines, kernel methods, Bayesian methods, optimization for machine learning, dimensionality reduction, information retrieval, and computational biology applications.

Arto Klami received his Ph.D. degree in computer science from Helsinki University of Technology in 2008 and worked as a postdoctoral researcher in Aalto University until 2012. Currently he works as an Academy Research Fellow (2013–2018) at Department of Computer Science and Helsinki Institute for Information Technology HIIT in University of Helsinki. His research interests include statistical machine learning, nonparametric Bayesian models, and integrated analysis of heterogeneous data sources.

Samuel Kaski is the director of Helsinki Institute for Information Technology HIIT, a joint research institute of Aalto University and University of Helsinki, and a professor of computer science at the Aalto University. His research field is statistical machine learning and computational data analysis, current application areas being in bioinformatics, neuroinformatics and proactive interfaces. He has published about 150 peer reviewed articles in these fields.