Personalized Machine Learning for Facial Expression Analysis

by Michael A. Feffer

Submitted to the
Department of Electrical Engineering and Computer
Science in Partial Fulfillment of the Requirements for the
Degree of

Master of Engineering in Electrical Engineering and Computer

Science at the

Massachusetts Institute of

Technology June, 2018

Author: _____

Department of Electrical Engineering and Computer Science
May 25, 2018

Certified by: _____

Ognjen (Oggi) Rudovic, PhD
Marie Curie Research Fellow
Thesis Supervisor

Certified by: _____

Rosalind Picard, ScD
Professor of Media Arts and Sciences
Thesis Supervisor

Accepted by: _____

Katrina LaCurts, PhD
Chair, Master of Engineering Thesis Committee

Personalized Machine Learning for Facial Expression Analysis

by Michael A. Feffer
Submitted to the Department of Electrical Engineering and Computer Science

May 25, 2018
In Partial Fulfillment of the Requirements for the Degree of Master of Engineering in Electrical Engineering and Computer Science

ABSTRACT

For this MEng Thesis Project, I investigated the personalization of deep convolutional networks for facial expression analysis. While prior work focused on population-based ("one-size-fits-all") models for prediction of affective states (valence/arousal), I constructed personalized versions of these models to improve upon state-of-the-art general models through solving a domain adaptation problem. This was done by starting with pre-trained deep models for face analysis and fine-tuning the last layers to specific subjects or subpopulations. For prediction, a "mixture of experts" (MoE) solution was employed to select the proper outputs based on the given input. The research questions answered in this project are: (1) What are the effects of model personalization on the estimation of valence and arousal from faces? (2) What is the amount of (un)supervised data needed to reach a target performance? Models produced in this research provide the foundation of a novel tool for personalized real-time estimation of target metrics.

Acknowledgements

First, I would like to thank my two supervisors, Dr. Ognjen (Oggi) Rudovic and Prof. Rosalind Picard, for their persistent and continued support and mentorship throughout my time working on this thesis. Through their teachings and guidance, I grew tremendously as a researcher and learned more about personalized machine learning, affective computing, and machine learning in general. Completing my research would have a much harder task without their encouragement and wisdom. Additional thanks goes to John Busche, who processed and sanitized the datasets used in my experiments. No machine learning can take place without data, so it was thanks to his scripts that I was able to make progress on my research in a timely fashion. I would also like to thank the other members of the Affective Computing Group for talking to me about their research and welcoming me into the group. All of their projects are fascinating, and especially given that computer science is sometimes seen as a soulless discipline driven by fame and money, it was extremely refreshing to see these tireless researchers devoted to working on projects aimed towards improving people's health and well-being.

I would also like to thank all of my friends that I have met during my time here at MIT, both inside and outside of the Institute. Every day at MIT is a blend of a living nightmare and a living dream; the exact proportions depend on the particular day. These people were with me to enjoy the days where MIT felt like a paradise and help me through the times when it felt considerably less so. Especially this past year, where I definitely pushed the limits of what was possible for me during my last academic school year at the Institute and was presented with unforeseen academic and non-academic challenges alike, their support was valuable in giving me the strength to keep moving forward.

Last but not least, I would like to thank my family for supporting me in everything from day one. I thank my mother, Carline Crevecoeur, as well as my father, Michael J. Feffer, for giving me the foundation of knowledge through my early education that eventually enabled me to learn math and computer science to the extent that I would be able to contribute to the fields of affective computing and personalized machine learning. More importantly, they also instilled in me a strong resolve that helped overcome the obstacles both life and MIT placed in my path, and I could always count on them for words of comfort and reassurance when rugged persistence alone was not enough. I also want to thank my four siblings, Danielle, Nicholas, Joseph, and Jacqueline Feffer, for being my greatest friends and closest companions. They inspire me to be the best version of myself while also reminding me to take it easy once in a while, both of which were key in making this research possible.

Table of Contents

List of Figures

List of Tables

1. Introduction

In recent years, machine learning has become increasingly popular for performing analysis and generating predictions based on data in a variety of different areas, especially in healthcare and fields devoted to improving general health and wellness [22]. In the realm of affective computing, it has been applied to many problems, including those of engagement and emotional state prediction [18] [21]. AI Systems that could accurately perform this analysis have a variety of different possible applications, such as human-robot interactions. As robots become increasingly complex and increasingly integrated into daily life, it will be important for them to perceive and understand not only human biometrics but also human emotional metrics. Enhanced perception of human emotions would allow them to avoid taking certain actions that would worsen a human's emotional state at a minimum and perhaps even influence them to take actions that could improve a human's emotional state. However, these systems are capable of impacting human life without being hosted in the artificial brain of a robot. Another possible application is the estimation of emotion expressions of individuals for the purpose of illustrating how they are being perceived. Many individuals with autism have difficulty expressing emotions in ways that can be easily understood by those without it [23], so affected individuals can use machine learning systems that can report perceived emotion expressions in order to learn how to express themselves so that they can be more easily understood by those around them.

These are only two out of many beneficial applications of this technology. Unfortunately, there have been difficulties training such systems in practice for several reasons. Most machine learning applications are successful because of their ability to generalize to unseen data and because they were trained on a plethora of easily obtainable training examples. Proposed systems to help solve human emotion perception have neither of these properties. People express emotions differently, even when they are part of the same culture or have the same mental state. Therefore, learning a general predictor or classifier with data from one set of people has issues detecting emotion expressions not only of people from a disjoint set but also of specific people within the training set. Moreover, it is difficult to obtain the massive amount of training data usually employed when training machine learning systems in this context due to the complexity of the labels. Whereas nearly any volunteers from the general public are capable of generating labels for digit classification or object recognition, emotion expression labels and measures must be given and calculated by trained professionals [18]. This has made obtaining training data very expensive, which in-turn has made it additionally difficult to train machine learning systems in this area.

In response to both of these issues, personalized machine learning has risen as a potential solution [29]. Personalized machine learning involves learning a general classifier or predictor based on data from a group of people and then fine-tuning the model to the profiles of specific individuals to create better results than the general model [21]. It has seen remarkable success in health and medical contexts [21] [22], and it has the potential to alleviate the issues that have plagued the problem of creating a machine learning system for emotion expression analysis.

My thesis work focused on a specific type of personalized machine learning through domain adaptation in this context. Specifically, I investigated the personalization of deep convolutional neural networks for facial expression analysis of face images by fine-tuning these networks for specific scenarios. The scenario primarily explored in my work is the estimation of valence and arousal from still images of faces. I mainly worked with two databases of images professionally

annotated with valence and arousal measures. The first was the AffectNet database, a diverse selection of images found on the Internet that have been annotated with the emotion categories and valence and arousal measures [18]. The second was the multimodal affect database REmote COLlaborative and Affective (RECOLA) database [24], used in the Audio/Visual Emotion Challenge and Workshop (AVEC) 2016 [25] (referred to as “AVEC” throughout this thesis). It is important to note that this latter database is actually composed of several videos, but preprocessing was done to extract the frames and corresponding labels to use for still image analysis.

Through this research, the main research questions I have answered are:
(1) What are the effects of model personalization on the problem of the estimation of valence and arousal from still images of faces?
(2) What is the amount of (un)supervised data needed to reach a target performance?
Models produced in this research provide the foundation of a novel tool for personalized real-time estimation of target metrics. In the sections that follow, I describe the methods I have explored and used in order to create the described machine learning models.

2. Related Work

In this section, I will first focus on formal definitions of domain adaptation and survey current implementations, after which I will discuss implementations of personalized machine learning and a neural network technique referred to as the “mixture of experts” with which I have experimented to produce effective analysis models. Most of the personalization and domain adaptation works discussed below try to solve problems in other areas, but they solve these problems in ways from which I have drawn inspiration while improving upon the accuracy and performance of existing techniques with regard to valence and arousal estimation.

2.1. Domain Adaptation

In most machine learning experiments, it is assumed that the training data and test data come from similar-enough domains that training on the training data can reasonably produce good performance on the test data. However, in problems like the one explored in this thesis project, this assumption cannot be made because the domains from which the training and test data originate are too different. “Domain adaptation” is a type of technique that attempts to remedy this difference so that models trained in one domain can perform effectively in another [1]. It is also a way in which one can perform personalization of a model (which will be covered in more detail in the next subsection).

Broadly speaking, there are two types of domain adaptation: unsupervised and semi-supervised. In unsupervised domain adaptation, no data from the new domain are available to help perform domain adaptation [2]. This usually means that features or probabilistic models learned from training in the old domain are used to perform domain adaptation. However, sometimes a limited amount of labeled data from the new domain are available to help calibrate the model. This is referred to as semi-supervised domain adaptation, since some labeled data are able to help perform the operation [2].

The changes made to the classification or prediction scheme in order to perform the adaptation are also ways to categorize the type of domain adaptation employed. Instance-based (or cost-based) adaptations include ones that reweight the losses of individual examples to change the loss function when training to better adapt to the new data [2]. Feature-based adaptations include ones that change or introduce features when training to better adapt to new data. Lastly, model-based adaptations include ones that change the underlying classification model or algorithm in order to adapt.

Jiang and Beijbom provided surveys of recent domain adaptation work as well as more in-depth definitions of domain adaptation than those given above [1][2]. The recent work in the rest of the papers described in this section fall roughly into three groups based on how the paper authors specifically performed domain adaptation. Methods in the first group map source features to a new feature space, typically one that is somehow linked to features in the target domain. In one paper, Blitzer et. al. described a method of domain adaptation for natural language processing (NLP) involving mapping source features to a subspace that is linked to target features and utilizing a new classifier that uses this projection to perform classification [3]. Kodirov et. al. employed a similar method for a visual recognition task with zero-shot learning that involved mapping features to a shared semantic space of lower dimension than the original feature space [8]. On the contrary, Yamada described an approach that involves mapping features to a higher dimensional space and reweights training examples "based on the ratio of test and train marginals in that space" [12].

A second group of papers can be thought of as an extension of the first. Instead of using a mapping function however, the paper's authors employed kernel functions to avoid computing the vectors in the high-dimensional spaces. One of the first versions of this was "frustratingly easy" for an NLP task, but it required a fully-supervised form of domain adaptation instead of an unsupervised one [5]. Gong et. al. used a "Geodesic Flow Kernel (GFK)" in order to implicitly map features to higher dimensional spaces where the two domains are more similar [7]. Tuia and Camps-Valls described a similar method called "Kernel Manifold Alignment" that aligns the two domains with a limited amount of labeled data [11].

One final group of papers describes methods that employ deep learning to either perform the domain adaptation automatically or learn a mapping function from the source domain to the target domain. Ganin et. al. used a "gradient reversal layer" to allow a deep neural network to learn domain adaptation through backpropagation, a normal part of the training process [6]. A team at IBM in collaboration with the National University of Singapore implemented a domain adaptation through a Deep Domain Adaptation Network (DDAN) on top of an AlexNet implementation to allow a network trained with high-quality images to describe clothing features in lower-quality images [4]. Long and Taigman explored the idea of "transfer networks" to perform domain adaptation [10] [11]. Long used residual networks to learn adaptive classifiers and transferable features while Taigman used a GAN to learn a mapping function [10] [11].

Since my research involved deep learning, I primarily looked to papers from the last group for inspiration, but papers in the other groups were still useful to keep in mind.

2.2. Personalized Machine Learning through Domain Adaptation

In the realm of personalized machine learning, there are a few pieces of related work that utilize domain adaptation to perform the personalization. Zen, Sangineto et. al. described methods of domain adaptation used to develop person-specific classifiers for facial action unit (AU) detection and pain recognition, two areas closely related to my research problem [15] [16]. Their methods employed support vector machines (SVMs) instead of deep neural networks, but their approaches were useful. Wachinger et. al. used a form of supervised machine learning with multinomial regression to personalize Alzheimer's disease classification [17]. Again, while deep learning was not used to classify here, their method of domain adaptation was useful for inspiration.

2.3. Mixture of Experts

"Mixture of experts" refers to a deep learning technique that involves training multiple "expert" subnetworks [13]. These networks produce the same type of output for the same type of input, but they are trained on different subsets of data so they are fine-tuned to specific contexts. At test time, all of them are given the same input, and an output for the overall network is created from combining the expert outputs in a certain way or otherwise somehow selecting the "best" output. Jacobs et. al. described a method of selecting an output involving a gating network that performs softmax activation to assign probabilities of randomly selecting the outputs of each of the expert networks for the current training example [13]. A more recent technique for using a mixture of experts approach involves a mixture of experts layer in a recurrent neural net (RNN) architecture that also has a gating network that performs softmax activation [14]. However, the outputs of this gating network are used to scale the outputs of the experts, the products of which are added together to yield the final output [14]. I employed a mixture of experts technique most similar to this last approach in my finished models.

3. Motivation: Personalized Machine Learning for Emotion Expression Detection through Domain Adaptation

Overall, my goal was to use personalization and domain adaptation to improve upon current deep learning techniques that have been pioneered to estimate emotion expressions from face images. Specifically, I developed techniques and evaluated their performance on two datasets.

The first set of data with which I worked is a publicly available dataset called the AffectNet dataset [18]. The dataset contains 450,000 cropped and centered images of faces and non-faces from the Internet that have been manually annotated by professionals with emotion expression labels [18]. In the case of a face, the annotators labeled the image with one of eight discrete facial expression labels and also calculated continuous valence and arousal values of the cropped and centered face in the image (a "non-face" expression label and values of -2 were used for valence and arousal in the case the image did not have a face) [18]. With this data, the original creators of the AffectNet database trained an AlexNet, a type of convolutional neural network (CNN), to

detect valence and arousal based on the input image [18]. They reported root-mean-squared error (RMSE) and the concordance correlation coefficient (CCC) for both valence and arousal when evaluated on the test set [18].

As part of my preliminary work for this project, I combined the training and validation sets from the dataset to produce my own training-validation-test split in a 60-20-20 ratio to train and evaluate a ResNet with 50 layers (a "residual network", another type of CNN as described in [20]) to perform the same task with a two-output dense (fully-connected) layer with linear activation for each output. I calculated RMSE and the intraclass correlation coefficient (ICC) for both valence and arousal (a metric similar to CCC but not identical), and my results are shown alongside theirs in the table below. (Note: the original paper authors did not release the test set, so this was the most similar experiment I could run.)

Table 1 - Preliminary Results and Baseline Comparisons. AffectNet results taken from cited paper.

	RMSE Valence	RMSE Arousal	ICC Valence	ICC Arousal
Train	0.283425	0.261155	0.819049	0.556734
Val	0.300042	0.270832	0.795662	0.516009
Test	0.302472	0.270152	0.792487	0.518108
AffectNet [18]	0.394	0.402	0.541 (CCC)	0.450 (CCC)

These initial results suggested that simply using residual networks, which have deeper and more complicated (yet powerful) architecture than traditional CNNs like AlexNet, may already improve upon state-of-the-art performance, but I wanted to see whether performing "personalization" with respect to the type of emotion involved could further bolster accuracy.

The second set of data was the REmote COLlaborative and Affective (RECOLA) database [24], used in the Audio/Visual Emotion Challenge and Workshop (AVEC) 2016 [25] (again, this database was referred to as "AVEC" throughout this thesis). This was a multimodal affect dataset containing face-centered videos of subjects speaking into a microphone while being recorded by a webcam [24]. These videos were professionally annotated with valence and arousal measures that reflected changes in the given subject's perceived emotion expression over time [24].

Since the neural networks I explored during my research were not recurrent, it was not possible to make use of a sequence of images and therefore the videos needed to be split into individual frames. More importantly, these frames needed to be centered on the face of the given user for optimal analysis. To accomplish this, the Faster R-CNN, a publicly available neural network empirically found to accurately detect faces in images [26], was utilized to detect the face bounding boxes in each frame and crop the frames to those bounding boxes. Labels were matched to frames based on the timestamps from the video with which the original labels and frames were associated.

I used videos from 18 subjects in my experiments with this dataset. Each video was approximately 5 minutes long and had a framerate of 25 frames per second, meaning that theoretically, this

process would result in 7500 frames per subject. However, while the Faster R-CNN was robust, it was not perfect, and occasionally it was not able to detect the face in the given frame. The end result was that the frame extraction process dropped frames and each subject only had around 7000 frames on average. Nevertheless, this dataset was effective in helping illustrate the domain adaptation problem and the failure of a "one-size-fits-all" predictor. The experiments performed with this dataset showed that domain adaptation and personalization can help a fine-tuned predictor outperform a general one.

4. Goals and Methods

For my thesis, I have worked on building personalized models through domain adaptation for the purpose of accurately detecting valence and arousal measures based on facial expressions. The goals of my research were to determine the following features and aspects of my final model:

(1) the effects of model personalization on performance in terms of valence and arousal prediction
(2) the amount of (un)supervised data needed to reach target performance

My starting models of choice were existing pre-trained convolutional networks, trained on millions of images from different contexts. A ResNet with 50 layers was my base network used in all experiments, as described in [20]. My starting implementation was loaded with optimal weights from the ImageNet classification task [30], but I retrained and fine-tuned this network with data from the two aforementioned datasets as needed.

At the same time, I researched different personalization strategies leading to the models that can automatically adapt to the user's face using a limited amount of labeled data or no labels at all, via unsupervised and semi-supervised domain adaptation methods. The method I primarily explored is a "mixture of experts" method. The baseline neural networks only had general output layers for valence and arousal given inputs from analysis done in higher layers of the network. However, I experimented with strategies to train multiple output layers that are fine-tuned to different types of input as well as a "selector layer" that can distinguish between these different types of input and select the corresponding output layer to use for a given input. The selector used in my experiments was implemented as a dense layer that received input from the ResNet (just as the experts did) with n outputs that used softmax activation (n being the number of experts). These outputs produced a categorical probability distribution used to weight the outputs of each corresponding expert (i.e. output k from the selector layer weighted both the valence and arousal outputs from expert k) and added them to the corresponding scaled valence and arousal outputs from the other experts to produce the overall output of the network.

Depending on how the network is trained, resulting models can outperform state-of-the-art general models. Furthermore, if the selector and experts are tuned properly, then for an input from the target domain, the selector can weight expert outputs based on the likelihood that the domain in which the expert has been trained is similar to the domain from which this new input has originated. This can be considered a form of unsupervised domain adaptation, since it is not necessary to use labels from the target domain for retraining. In the case of the AffectNet data, I

“personalized” experts by training one expert for each type of emotion, and for the AVEC dataset, I personalized experts by training one expert for each of 9 different subjects.

While this approach was primarily used on image data, its personalized learning strategy should be applicable to other modalities, such as voice, physiology and body expressions. It should also be a novel personalization strategy for model adaptation to a new subject as more data of that subject become available.

5. Benchmarks and Evaluation

Network architectures were evaluated using root-mean-squared error (RMSE), the intraclass correlation coefficient (ICC), and the concordance correlation coefficient (CCC) of sets of predictions on various datasets relative to ground truth. These three metrics together allow for comparisons with results in papers from other researchers in this field. For instance, the authors of the AffectNet database paper used RMSE and CCC for their results [18].

Additionally, for both the AffectNet and AVEC datasets, I used a ResNet with a general output layer as a baseline predictor. In both cases, this predictor was fine-tuned such that the ResNet weights and output layer weights were trained together on the dataset, followed by a fine-tuning of the output layer. RMSE, ICC, and CCC values of any future networks from any datasets were directly compared to the same values of this baseline model from those same datasets. This baseline model was important for both the AffectNet and AVEC datasets. For the AffectNet dataset, it was noted previously that just by using a ResNet instead of an AlexNet, one can achieve better performance than the values specified in the original paper. Using this model as a baseline controls for using a ResNet instead of an AlexNet and allows one to determine whether personalization through domain adaptation makes a measurable positive impact. There were few other baselines that were directly comparable for the AVEC dataset, so it was near impossible to examine the results of personalization otherwise.

6. Results

6.1. Experiments on the AffectNet Dataset

AffectNet was the first dataset with which I worked. It contained a very large number of images with faces expressing different emotions. This made it an ideal candidate for helping determine a training strategy for fitting networks not only for this dataset but also for all future datasets. When working with all datasets, the models I trained involved layers appended to the end of the ResNet to perform regression from features extracted by the ResNet to valence and arousal output. Regardless of whether there was only one such “regression layer” performing this regression (referred to as a “shared network” or “one-size-fits-all” solution in other parts of this thesis) or a mixture of experts architecture, the presence of these additional layers raised questions regarding how the two parts of these models (the ResNet and regression layer(s)) should be trained. Questions such as “does training the ResNet improve performance, and if so, by how much?” and “what is the proper training order of the components (ResNet+regression training followed by

more regression training versus the reverse order)?" needed to be answered before any experimentation exploring personalization and domain adaptation solutions could take place.

To answer these questions, I followed-up on my preliminary findings detailed in a previous section and tested the models resulting from different training strategies using the 60-20-20 split of the AffectNet dataset. Each model was composed of a ResNet and a shared network in the form of a two-output dense layer with linear activation for valence and arousal outputs, and the resulting model was trained via gradient descent to minimize mean-squared error for both outputs on the AffectNet data. The training strategies employed tested how the combinations and ordering of two training configurations improved or hindered final network performance. One of these configurations allowed for training of the final dense layer alone by "freezing" the ResNet weights during training (i.e. setting the gradients to zero during backpropagation and gradient descent), and the other configurations allowed all layer weights to be tuned together. Combining these configurations in different ways yielded four different training strategies, all of which were used to perform training of the combined network, and the results have been reproduced in the table below.

Table 2 - Training Strategy Evaluation

	ICC						RMSE					
Strategy	Valence			Arousal			Valence			Arousal		
	Train	Val	Test	Train	Val	Test	Train	Val	Test	Train	Val	Test
A	0.509	0.504	0.505	0.222	0.212	0.213	0.443	0.444	0.446	0.315	0.316	0.316
B	0.806	0.782	0.780	0.542	0.503	0.506	0.287	0.303	0.305	0.263	0.273	0.272
C	0.813	0.791	0.788	0.537	0.499	0.501	0.294	0.309	0.312	0.266	0.275	0.274
D	**0.819**	**0.796**	**0.792**	**0.557**	**0.516**	**0.518**	**0.283**	**0.300**	**0.302**	**0.261**	**0.271**	**0.270**

Strategy A: Training shared network only
Strategy B: Training shared network only followed by training shared network and ResNet together
Strategy C: Training shared network and ResNet together
Strategy D: Training shared network and ResNet together followed by training shared network only

From this experiment, it was clear that fine-tuning the ResNet offered clear advantages over tuning the shared network alone. Moreover, this experiment also demonstrated that tuning the ResNet and shared network together followed by fine-tuning of the shared network offered slightly better performance than performing those training operations in the reverse order. Therefore, the training algorithms utilized in future experiments were derived in light of these findings and typically involved optimizing the ResNet first whenever possible.

Having discovered the best way to perform ResNet and regression fitting, the next step was to perform "personalization" to emotion type as described in a previous section. Two methods of personalization were explored: one in which the mixture of experts was trained in a supervised fashion (involving training the selector layer to recognize emotion expressions given the emotion label for each image and training each expert on the subset of data characterized by the emotion

for which they were responsible) and one in which the mixture of experts was trained in an unsupervised fashion (involving randomly initializing the selector and experts together and optimizing parameters such that the weighted sum of expert outputs is optimized relative to ground-truth). Both of these networks were trained and evaluated using the split in the AffectNet data utilized previously. The selector in each case was implemented as a dense layer that received input from the ResNet with softmax activation on the outputs as described in a previous section. Additionally, to evaluate whether training experts in a supervised fashion by training on subsets of the data was a sensible thing to do, results were also obtained by simulating a "perfect" selector and choosing the expert corresponding to the ground-truth emotion represented in the given frame and measuring its output against the ground-truth valence and arousal values (instead of using a selector to weight and sum all of the expert outputs together). The results of these configurations are included alongside the results with the best "one-size-fits-all" network in the table that follows.

Table 3 - Results of Emotion Personalization using AffectNet Data

	ICC						RMSE					
Strategy	Valence			Arousal			Valence			Arousal		
	Train	Val	Test	Train	Val	Test	Train	Val	Test	Train	Val	Test
A	0.819	0.796	0.792	0.557	0.516	0.518	0.283	0.300	0.302	0.261	0.271	0.270
B	**0.928**	**0.925**	**0.924**	**0.831**	**0.821**	**0.821**	**0.197**	**0.201**	**0.202**	**0.181**	**0.185**	**0.185**
C	0.823	0.801	0.798	0.563	0.522	0.523	0.3	0.32	0.32	0.264	0.274	0.273
D	0.823	0.818	0.821	0.570	0.562	0.568	0.29	0.29	0.288	0.236	0.238	0.237

Strategy A: Best general model obtained from previous experimentation
Strategy B: Training experts in isolation with simulated ideal selector
Strategy C: Training experts in isolation with learned weighted sum selector
Strategy D: Training experts and weighted sum selector together

Based on these results, it appears that a perfect selector would allow a personalized mixture of experts model to greatly outperform a general model. However, none of the models with a selector obtained via standard learning algorithms had any appreciable benefit over the general model (and rounding to one fewer significant digit yields virtually no difference). Evidently, based on the dramatic improvement that results from simulating a perfect selector, there appears to be an issue with how the current selector is learned. Exploring other methods of learning a selector and other ways to model a selector that can perform the required weighted sum is an area of future work and research.

6.2. Experiments on the AVEC Dataset

After determining the best performing training strategy from working with the AffectNet dataset, I turned towards working with the AVEC dataset, where actual personalization could be performed given that data came from individual subjects. The experiments performed with this dataset indeed proved that there are clear benefits of personalization when trying to adapt to data of unseen individuals because a personalized model has greater flexibility. The results of working

on the AVEC dataset are covered in greater detail in the paper that follows [31]. In helping produce this paper, I was responsible for running the experiments described and gathering results to use for analysis. Analysis of results was done by both Dr. Rudovic and myself. Limitations of our work and ideas for future work were contributed by Dr. Rudovic and Prof. Picard. All three of us had input into the model approach.

A Mixture of Personalized Experts for Human Affect Estimation

Michael Feffer, Ognjen (Oggi) Rudovic, and Rosalind W. Picard

MIT Media Lab, Massachusetts Institute of Technology, Cambridge, MA 02139, USA
{mfeffer,orudovic,roz}@mit.edu,

Abstract. We investigate the personalization of deep convolutional neural networks for facial expression analysis from still images. While prior work has focused on population-based ("one-size-fits-all") approaches, we formulate and construct personalized models via a mixture of experts and supervised domain adaptation approach, showing that it improves greatly upon non-personalized models. Our experiments demonstrate the ability of the model personalization to quickly and effectively adapt to limited amounts of target data. We also provide a novel training methodology and architecture for creating personalized machine learning models for more effective analysis of emotion state.

Keywords: mixture of experts, domain adaptation, personalized machine learning, residual networks

1 Introduction

In recent years, machine learning has become increasingly popular for performing analysis and generating predictions based on data in a variety of different areas, especially in healthcare and fields devoted to improving health and wellness [1, 2]. In the realm of affective computing, it has been applied to tasks such as automated analysis of persons' engagement, personality, and affective states during human-computer [3] and human-robot interaction [4]. For instance, as robots become increasingly complex and integrated into daily life, it will be important for them to perceive and understand not only human biometrics but also human emotional metrics. Enhanced perception of human emotions could enable robots to avoid actions that would worsen a human's emotional state and perhaps even influence them to act in a way that could improve a human's well-being. Moreover, in the advent of powerful machine learning capabilities for mobile devices, it is possible nowadays to perform emotion analysis through smartphone cameras. Therefore, a smartphone application could be programmed to detect a user's emotions and recommend strategies for dealing with negative emotions or actively attempt to improve the user's mood. Lastly, emotion analysis could be used for emotion and engagement detection and coaching for individuals with autism, who have inherent difficulties in reading others' emotions and expressing their own in a way that can be easily understood by neurotypicals.

Most machine learning approaches are successful because the models produced can generalize to unseen data and were trained on a plethora of existing data. However, most of the existing approaches ignore the fact that people express affect differently, even when they are part of the same culture. Therefore, learning a general predictor or classifier (the traditional "one-size-fits-all" approach) with data from one set of people typically underperforms when tested on people from a disjoint set but also on specific people within the training set. Moreover, it is difficult to obtain a large amount of training data for each target subject because providing labels for these data is costly in terms of time and resources [5]. Thus, improving machine learning approaches so that they can efficiently leverage small amounts of training data to adapt to each target subject is of large importance for increasing the model's effectiveness. To address these challenges, a number of works attempted model personalization [4]. The goal of model personalization is to leverage the individual-specific data in order to adapt a general classifier (the "one-size-fits-all" approach) learned from data of previously seen people (source subjects) to the profiles of specific individuals (target subjects). This has shown great improvements in a number of machine learning tasks related to human-data analysis (e.g., [6, 2, 1]).

In this paper, we have focused on a specific type of model personalization for estimation of facial affect (valence and arousal) using the notion of an ensemble of models and domain adaptation [7]. Specifically, we use the Mixture-of-Experts (MoEs) approach [8] to model the facial expression data (face images) of source subjects, for whom it is assumed that a large amount of image labels for the facial affect is readily available. We adopt MoEs where each expert represents one of the source subjects, which has greater modeling flexibility and improved ability to capture the large variation in facial expressions of different subjects compared to a single expert, which typically captures the average variation. While this approach performs better fitting of the source subjects, it is not guaranteed that this performance translates to previously unseen subjects (target), as confirmed in our experiments. To this end, we perform a supervised adaptation of the learned MoEs model using a varying portion of labeled data of all of the target subjects. We show: (i) that this approach achieves improved performance on the target subjects compared to a single expert model, and (ii) that it also outperforms the same model trained solely on the data of target subjects used to adapt the model. The latter is due to the ability of our approach to efficiently leverage the data of the source subjects. We demonstrate this in the context of deep neural networks that we tuned for "end-to-end" estimation of valence and arousal from still images of faces from the multimodal affect database REmote COLlaborative and Affective (RECOLA) database [9], used in the Audio/Visual Emotion Challenge and Workshop (AVEC) 2016 [10]. Note, however, that the focus of this work is not to outperform existing models in affect estimation but instead to examine how personalization and supervised domain adaptation can bolster current affect estimation models, an intersection of research areas that has yet to be explored.

In the sections that follow, we describe our approach to personalizing deep convolutional neural networks and a MoEs model. We first discuss related work both regarding the domain adaptation and MoEs. Then, we describe our mixture of personalized experts approach, followed by its experimental validation and derived conclusions.

2 Related Work

MoEs refers to a learning approach that involves training multiple "expert" subnetworks. These networks produce the same type of output for the same type of input, but they are trained on different subsets of data so that they are fine-tuned to specific contexts. At test time, all of them are given the same input, and an output for the overall network is created by combining the expert outputs in a certain way or somehow selecting the "best" output. Among the first proponents of this technique, [8] introduce a method of selecting an output by using a gating network that performs softmax activation to assign probabilities of randomly selecting the outputs of each of the expert networks for the current training example, after which an output is randomly chosen via a probability draw based on that distribution. Since then, it has been studied extensively for over twenty years, and in that time, expert models with other classifiers such as SVMs [11] and Gaussian Processes [12, 13], have also been researched. With regard to deep neural networks, a myriad of different network architectures have been explored, ranging from hierarchical and ensemble experts [14] to networks with infinite numbers of experts [15] and nested mixtures of experts that allow for deep learning [16]. Although our work is built upon the same framework of MoEs, it differs from existing approaches as we personalize the experts to each subject. Furthermore, we combine the learning of MoEs with supervised domain adaptation (DA) [7] in order to efficiently adapt the model to previously unseen subjects. While there is a large body of work on DA, we do not intend here to improve upon existing DA methods. We rather use its notion when adapting the target MoE model that we propose for the model personalization to target subjects. For a detailed review of existing DA approaches, see [7]. Also, even though model personalization has been researched in several previous contexts (e.g., self-reported pain analysis [6] and robot therapy for autism [4]), none of these methods have been explored using a mixture of experts architecture in the context of model personalization. To this end, we have adopted this approach in our personalized framework. We have taken particular inspiration from [16] to utilize a mixture of experts approach that outputs a weighted combination of expert outputs based on a gating network learned form source subjects, as will be elaborated in the following sections.

3 Methodology

3.1 Notation

We consider the following setting: we are given a number of training subjects (source), denoted as $P^{(s)} = \{p_1^{(s)}, \ldots, p_{n_s}^{(s)}\}$, where $id^{(s)} = 1, \ldots, n_s$ represents

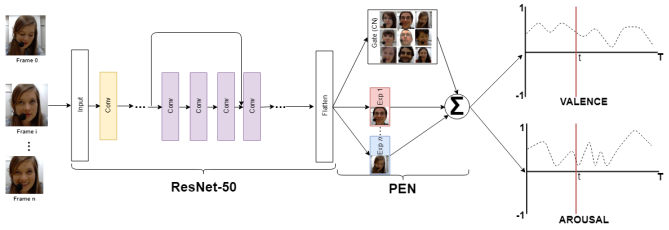


Fig. 1: The architecture of the proposed approach. The input is a subject's video and the outputs are his/her estimated valence and arousal levels. We first applied Faster R-CNN [17] to extract the face region from each raw image frame. The extracted faces were passed through a ResNet-50 [18], fine-tuned on source subjects' data. The obtained deep features were used as input to our personalized expert network (PEN) for automatic estimation of valence and arousal. This also contains a "gating network" (CN) that assigns different weights to each expert in the PEN during inference of new test images.

the subject id, and n_s is the number of the subjects. Data of these source subjects are used to learn a shared model optimized on these subjects. We are also given a number of test subjects (target) which were previously unseen by the shared model. These are denoted as: $P^{(t)} = \{p_1^{(t)}, \ldots, p_{n_t}^{(t)}\}$, and $id^{(t)} = 1, \ldots, n_t$. Then, starting from the shared model, the goal is to achieve the best possible performance on $P^{(t)}$. Furthermore, the data of each subject is stored as $p_i = \{X_i, Y_i\}$[1], where input features of the subject i are given by: $X_i = [x_{i1}; \ldots; x_{iN_i}] \in \mathcal{R}^{N_i \times D_x}$, where N_i is the number of available examples of the subject, and D_x is the input feature size. Note that these examples may be temporally correlated (in the case of video data) or be randomly sampled from independent observations of the subject. Likewise, the output labels (in our case, the levels of valence and arousal of the subject), are given as: $Y_i = [y_{i1}^v\, y_{i1}^a; \ldots; y_{iN_i}^v\, y_{iN_i}^a] \in \mathcal{R}^{N_i \times D_y}$, where $D_y = 2$. In what follows, we first describe how the shared model is learned from $P^{(s)}$, and used to estimate target affective states on $P^{(t)}$. Then, we propose an expert model, where each expert corresponds to one of the source subjects. The key to our approach is the model personalization step, where we propose a learning strategy designed to adapt the expert model to target subjects, using a varying portion of their (previously seen) data. Lastly, we provide details of the learning and inference in the personalized expert model.

3.2 Shared Model

We start by learning a shared model that is trained on data of all source subjects, without taking into account their id. This model is based on a deep architecture composed of the layers of a residual network (ResNet) [18], a pre-trained

[1] For notational simplicity, we drop here the dependence on the source/target subjects

deep network commonly used to extract the most informative features for object classification [18]. We used the ResNet-50 architecture composed of multiple three-layer "bottleneck" building blocks (containing 50 layers in total) described in the original ResNet paper [18]. When beginning training, we initialized the layer weights corresponding to weights that yield published optimal performance on the ImageNet dataset [19]. However, we use all of the layers of the network but the last (i.e. the softmax layer) as it was optimized for classification of various object categories such as "laptop" and "orange". We instead replace the last layer with an data-uninformed fully-connected dense layer with linear activation, which we use to fine-tune the ResNet weights for the target task: the estimation of valence and arousal from face images (see Fig. 1). This architecture receives as input the face images of source subjects (x) and passes the most discriminative (deep) facial features (z) to the regression layer in the output through the following mapping: $x \rightarrow z \rightarrow \tilde{y}$, where $\tilde{y}$ are the estimated levels of valence and arousal. The training of the shared model is divided into two stages. First, the whole network (ResNet included) is optimized for estimation of y. Then, we freeze all of the (fine-tuned) layers of the ResNet ($W^{r-net} \in \mathcal{R}^{D_x \times D_z}$) and additionally fine-tune the last (regression) layer ($W^s \in \mathcal{R}^{D_z \times D_y}$). We experimented with different learning strategies and found that this one performed the best. This is because of the large number of parameters that need to be tuned simultaneously. Due to this, the network underfits the last layer, so we overcome this by additional tuning of the last layer. The resulting network, referred to as the shared network (SN), is used to initialize the expert network (EN), which is then adapted to the population of target subjects as described below.

3.3 Personalized Expert Network (PEN)

The learning of the expert network is accomplished using the Mixture-of-Experts (MoEs) approach, where an expert network (EN) is comprised of a set of layers called "experts" that are denoted as $e_1, \ldots, e_n$. Furthermore, an EN is also comprised of a "gating network" denoted as CN (which in our case is a person selector network). Its output is used to weight the contribution (relevance) of each expert during the inference stage. In our personalized model setting, during the model training on source subjects, each expert corresponds to one training subject (thus, n_s experts). This personalization yields a personalized expert network (PEN). Each expert is modeled using a feed-forward network with fully connected linear activations, as used for the SN, but each with their own parameters. Thus, the following mapping is learned: $x \rightarrow z \rightarrow \tilde{y}^e = [\tilde{y}_1^e \ \ldots \ \tilde{y}_{n^{(s)}}^e]$, where $\tilde{y}_i^e$ are the valence/arousal estimates by the i-th expert. Likewise, the CN aims to learn the mapping: $x \rightarrow z \rightarrow h \rightarrow \tilde{c} = [\tilde{c}_1 \ \ldots \ \tilde{c}_{n^{(s)}}]$, where $\tilde{c}_i$ is the (normalized) weight for the i-th expert during the model learning. More specifically, the CN is designed as a two-layer network. The first layer is a fully-connected feed-forward linear activation network. The linear activations are then passed through a softmax layer, providing the probability that the input sample comes from one of n_s source subjects and thus assuring that the outputs of the CN sum to one. More formally, given an input x to the ResNet, the output of the

PEN is defined as:

$$\tilde{y} = \sum_{i=1}^{n_s} \tilde{c}_i \cdot \tilde{y}_i^e = \tilde{c} \otimes \tilde{y}^e, \tag{1}$$

where $\tilde{y}$ is the weighted combination of the individual experts. The output of the CN, given the activations (z) of the ResNet is obtained as:

$$\tilde{c} = \text{softmax}(h) \text{ and } h = \text{fcl}(z; W^s), \tag{2}$$

where z is passed through the fully connected layer (fcl), the output of which is fed into the softmax function to yield a categorical probability distribution over the source subjects. Note that during training of the PEN, the targets for the CN output are the subjects' ids encoded via the 1-hot encoding (e.g., $c = [0, 1, 0, \ldots, 0]$ for subject $i = 2$). Similarly, each expert $i = 1, \ldots, n^{(s)}$ produces estimates for target affective dimensions as:

$$\tilde{y}_i^e = \text{fcl}(z; W_i^e), \tag{3}$$

where z is multiplied by a trainable weight matrix W_i^e for expert i.

The network personalization is attained by using the prior knowledge about the source subjects: each expert is supposed to represent one of the subjects, and CN performs the selection of that expert during the model learning. Therefore, given the training data of $n^{(s)}$ source subjects, the overall loss is the sum of losses due to differences between y and $\tilde{y}$ (the weighted combination of expert outputs) as well as losses from the CN. However, in practice this loss does not enforce the sparsity on the selector's weights ($\tilde{c}$), which may result in the learned PEN expending too much modeling power of each expert on trying to fit the data of all source subjects. In turn, we may end up with an expert that is unable to specialize in individual characteristics of the subject, resulting in an average model that is suboptimal. To overcome this, we introduce the L-1 sparsity constraint on the output of the CN, but this cannot be done directly because the outputs of that layer always sum to one. Instead, we enforce the sparsity on the activations of the fcl of the SN. This leads to the following joint loss being optimized during the parameter learning:

$$\alpha = \alpha^y + \lambda_0 \alpha_c^s + \lambda_1 \alpha_r^s, \tag{4}$$

where α^y is the mean-squared error (MSE) loss between the PEN estimates and the ground-truth labels for valence and arousal (y). (λ_0, λ_1) are regularization parameters that are optimized on the validation data. They control the trade-off between the model performance and the penalty terms: α_c^s, which ensures that each expert focuses on its corresponding subject, and α_r^s further ensures this through the sparsity constraint. These individual losses are defined as:

$$\alpha^y = \frac{1}{N^{(s)}} \sum_{i=1}^{n^{(s)}} \sum_{j=1}^{N_i^{(s)}} (y_j^i - \tilde{y}_j^i)(y_j^i - \tilde{y}_j^i)^T, \tag{5}$$

and $N_i^{(s)}$ and $N^{(s)}$ are the number of training data per source subject and overall, respectively. The selector loss is given by:

$$\alpha_c^s = \frac{1}{N^{(s)}} \sum_{i=1}^{n^{(s)}} \sum_{j=1}^{N_i^{(s)}} H(c_j^i, \tilde{c}_j^i), \tag{6}$$

where $H(\cdot, \cdot)$ is the cross-entropy loss that is commonly used for discrete variables, as is the case here. Finally, the standard L-1 sparsity is enforced via:

$$\alpha_r^s = \frac{1}{N^{(s)}} \sum_{i=1}^{n^{(s)}} \sum_{j=1}^{N_i^{(s)}} |h_j^i|_{L_1}. \tag{7}$$

Note that this loss treats each activation/image frame independently, and therefore, no structure (prior information) about the source subjects is directly modeled. Nevertheless, this should still result in the learned activations being sparse, on average, for different face images of the subjects.

3.4 PEN: Supervised Adaptation to Target Population

Once the PEN parameters are optimized for the source subjects, our goal is to achieve the best performance on target subjects that the PEN has not seen before. In the traditional supervised machine learning approach, this would be evaluated using the network learned solely from the data of the source subjects. However, this typically leads to the learned model attaining a lower performance on target subjects than on the source subjects, as expected. Also, since the PEN is "tuned" to the latter, it may even more easily overfit those subjects, leading to the comparable to or worse performance than that of the SN trained on the target subjects. This is despite the modeling flexibility introduced by the local (person) experts, which allows the PEN to better fit the source subjects. Conversely, in the proposed personalized learning approach, we adopt the supervised domain adaptation approach [20], where the target subjects are treated as another domain assumed to be different from the one used to train the PEN. To investigate this, we pool the data of target subjects and use a varying portion of this data (with approximately equal amounts of data from each subject) to "tune" the PEN to those subjects. This is driven by two assumptions. First, we should be able to adapt the PEN to target subjects using a (significantly) smaller number of target data than originally used from the source. This is mainly because the network has been pre-trained on the latter and should be able to adapt to new subjects more easily. Second, while the same assumption may hold for the SN, the proposed PEN is more flexible in its modeling power (due to the multiple experts). This in turn should allow it to better adapt to the previously unseen subjects by focusing on their individual characteristics, thus, avoiding the limitations of the "one-size-fits-all" approach. Formally, this is attempted by fine-tuning the PEN to the population of target subjects via the following adaptation loss:

$$\alpha_{ad} = \alpha^{y^t} + \lambda_1 \alpha_r^{s,t}, \tag{8}$$

This supervised adaptation loss uses a small number of labeled data of target subjects (x^t, y^t) to adjust the PEN parameters. It is important to note that we do not use the ids of target subjects during the model adaptation, thus, the parameters of the CN are optimized in an unsupervised fashion by setting $\alpha_c^s = 0$. However, we still impose the sparsity constraint on the CN parameters. We also do not further optimize the ResNet parameters - these are rather used as the feature extractor during the adaptation stage. The benefits of this are two-fold. First, we preserve the privacy of the target subjects[2]. This is important in the context of many applications where the user does not want to reveal his/her identity, while still being able to receive estimates of target affective states. Second, instead of completely "overwriting" the previously learned PEN, the adapted model performs the actual adaptation of the model rather than learning it from scratch. This is motivated by the assumption that the PEN model has seen a large amount of labeled data from source subjects, thus encapsulating valuable knowledge about the affect expressions of different subjects. In this way, more robust adaptation of the PEN to new subjects is expected. Ideally, when $n^{(s)} > n^{(t)}$, PEN should be able to specialize one expert to each target subject, while compensating the lack of target subject data with the knowledge extracted from the source subjects.

3.5 Learning, Inference and Implementation Details

We first summarize the learning and inference in the proposed models, followed by the implementation details of our deep architecture. We start with the learning of the SN. The learning of the model parameters $\{W^{r-net}, W^s\}$ is performed in two steps: (i) The joint fine-tuning of the ResNet with the parameter optimization in the appended fcl, used for estimation of valence and arousal. (ii) The additional fine-tuning of the fcl parameters $\{W^s\}$, while freezing the ResNet parameters i.e., $\partial W^{r-net} = 0$. For (i) W^{r-net} was initialized to weights corresponding to optimal training on the ImageNet dataset, and W^s was initialized randomly. Our implementation appended the SN to the last flattening layer of the ResNet and trained all of these layers together. For (ii) we took our best model from (i) (based on validation set performance) and froze every layer in the architecture besides the SN. After more training, we saved this fully-adapted architecture and used it as the starting point for expert layers going forward.

To learn the PEN parameters, we first initialized each expert using the weights of the SN as $W^{e_i} \leftarrow W^s$, $i = 1, \ldots, n^s$. This ensures that individual experts do not overfit the corresponding source subjects, which could adversely affect their generalization to target subjects (we describe this below). The initial learning of the CN was done in isolation from the rest of the network. Only the outputs of the fine-tuned ResNet were used as input (z), and the $W^{c,0}$ was optimized by minimizing the loss $\alpha_c^s + \lambda_1 \alpha_r^s$ using the 1-hot encoding of the source

[2] For instance, only the ResNet features of target subjects need be provided as input to the adapted model, as original face images cannot be reconstructed from those features.

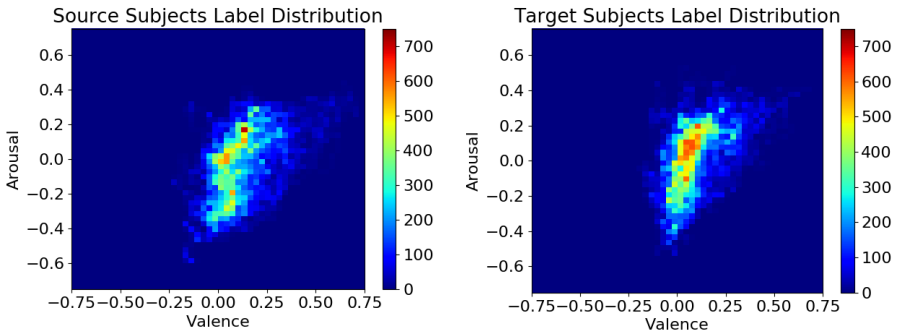
Algorithm 1 Personalized Experts Network (PEN)

Source Learning Input: Source persons data $P^{(s)} = \{p_1^{(s)}, \ldots, p_{n_s}^{(s)}\}$
step 1: Fine-tune ResNet weights (W^{r-net}) and optimize SN (W^s)
step 2: Freeze ResNet weights and fine-tune SN (W^s)
step 3: Initialize the experts ($W_i^e \leftarrow W^s$) and optimize PEN (W^s, W^c)
Target Adaptation Input: Target persons data $P_{ad}^{(t)} \leftarrow n\%$ of $P^{(t)} = \{p_1^{(t)}, \ldots, p_{n_t}^{(t)}\}$
step 1: Fine-tune PEN weights (W^s, W^c) using adaptation data $P_{ad}^{(t)}$

Inference Input: Unseen target persons data $P_{un}^{(t)} = P^{(t)} \cap P_{ad}^{(t)}$
Output: $(\tilde{y}^v, \tilde{y}^a) \leftarrow \text{PEN}(P_{un}^{(t)})$

subjects' ids as ground-truth labels (c). Then, the joint learning of the PEN was performed by minimizing the loss in Eq.(1). We noticed that the model with the individually tuned experts to data of each subject (thus, in isolation from the selector) generalized worse to target subjects than when only the joint learning of the selector and the experts was attempted. For this reason, we report our results only for the latter setting. Also, we noticed that doing the joint learning for a large number of epochs even led to overfitting of source subjects (on their left-out portion of the data). For this reason, we used the early stopping strategy, which prevented the model from overfitting after only five epochs. This model was subsequently used for further adaptation to target subjects by minimizing the PEN loss (Eq.(8)) on the adaptation data of target subjects – see Sec. 4. These learning and inference steps are summarized in Alg. 1.

We implemented the PEN architecture using the Keras API [21] with the Tensorflow [22] back-end. For the soft-max and fcls, we used the existing implementations. The layer sizes were 2048 x 9 (for a total of 18441 parameters including the 9 offsets) for the CN and 2048 x 2 (for a total of 4098 parameters including the 2 offsets) for a given expert. The reweighting part of the weighted sum was performed via a custom Lambda layer that took the tensors output by the CN and the experts as input and scaled the outputs of each expert by the corresponding CN output. Afterwards, the scaled components were summed via an Addition layer and output as the overall network output. During training with source data, mean-squared error was used to train with this overall output, and categorical cross-entropy loss was used to train with the output of the CN. However, to implement the PEN loss, we created a custom loss function that performed L-1 regularization on the pre-softmax- activation CN output by taking the mean of the absolute value of the pre-activation tensor and summing it with the categorical cross-entropy loss. The parameter optimization was then performed using the standard back-propagation algorithm and Adadelta optimizer with the default parameters. The details of the employed validation settings are provided in the description of the experiments.



(a) (b)

Fig. 2: The joint distribution of the labels of affective dimensions: valence and arousal, in source and target subjects. By personalizing the PEN using adaptation data of target subjects, we reduce the difference between two distributions.

4 Experiments

To demonstrate the effects of the model personalization using the proposed PEN approach, we used image sequences from the multimodal affect database REmote COLlaborative and Affective (RECOLA) database [9] , used in the Audio/Visual Emotion Challenge and Workshop (AVEC) 2016 [10] This database contains four modalities or sensor-signals: audio, video, electro-dermal activity (EDA), and electro-cardiogram (ECG). The data are synchronized with the video modality and coded by five human experts. Specifically, the gold standard labels (i.e., the aligned codings) for two affective dimensions - valence and arousal - are provided by the database creators. The time-continuous codings for each dimension are provided on a scale from -1 to +1. For our experiments, we used the publicly available data partitions from AVEC 2016, namely, training (9 subjects) and development (9 subjects) sets. We refer to these as source and target persons. The video of each person is 5 mins long (25fps), resulting in $\sim$7k image frames per person after the face detection using Faster R-CNN [17]. We used the processed face images as input to the models, and the output was the estimated levels of valence and arousal per frame. The models' performance was evaluated in terms of Root-mean-square-error (RMSE) and concordance correlation coefficient (CCC), both of which were used as competition measures in AVEC and were computed on the pairs of model estimates and the gold-standard labels.

We performed the following experiments: (i) the SN and PEN models trained and tested on the source subjects ($P^{(s)}$), in order to evaluate the modeling power of the latter when fitting the data. We denote these models as s-SN and s-PEN. (ii) These models were then adapted using the adaptation data ($P_{ad}^{(t)}$), a varying portion of the data from target subjects, and evaluated on the non-overlapping data of the target subjects. The data of target subjects were split into the adap-

tation and test data at random. Specifically, we formed $P_{ad}^{(t)}$ by incrementally sampling the following amount of data: $n = 5\%, 10\%, 20\%, 30\%$ and 50% from each target subject and then combining the data across the subjects. For example, from 7k frames per target subject, $n = 5\%$ led to $P_{ad}^{(t)}$ of size 350*9=3150 images, and for $n = 50\%$ the $P_{ad}^{(t)}$ size was ten times larger.

For testing, we always used the same (nonoverlapping) 50% of target subject data. The goal of these comparisons was to assess the models' behavior when using a different amount of target subject data to adapt the deep networks. The main premise here is that due to the modeling flexibility of the PEN, it would be able to better adapt to the target subjects than would SN. To show the benefits of using the data of source subjects to learn the (non-adapted) models, we also tested the SN and PEN model architectures trained from scratch on the 5 different adaptation sets $P_{ad}^{(t)}$ formed from the $n\%$ of target data as described above and evaluated on the test set, i.e. the left-out 50% of target subject's data. (iii). We refer to these settings as target SN (t-SN) and target PEN (t-PEN), respectively.

To form the base models, we first trained the s-SN together with the ResNet (see Alg.1). This was accomplished using 80% of the source data (evenly sampled from each subject) while the remaining 20% were left out for the model validation. We found that 10 epochs were sufficient to fine-tune the ResNet without overfitting it due to its large number of parameters and the limited number of data used to tune the network. Further optimization of the SN and PEN configurations was performed using 30 epochs for training the models, which was enough for the models' loss to converge. To select the regularization parameters when training s-PEN, we cross-validated (λ_0, λ_1) using the following values $\{10^{-4}, 10^{-3}, .., 0, 1, 10, 100\}$, with 10^{-3} performing the best for both. During model adaptation and training of the target models, we used the same regularization parameters. Note that for these models, no subject id was provided during the adaptation/training, as we assumed that these are available only for the source subjects.

Table 4 compares our networks initially trained with source data, s-SN and s-PEN, with the t-SN and t-PEN, trained from scratch using only $n\%$ of target data, as mentioned above[3]. The results show that the initial training of the SN and PEN on the source data improved performance on target test data, compared to the t-SN and t-PEN. This evidences that the proposed models were able to efficiently leverage the data of the source subjects during the estimation. Moreover, the s-PEN approach eventually outperforms the SN architecture after as little as 10% of target (adaptation) data, due to its more flexible architecture that allowed it to easier adapt to target population data. Furthermore, we observe that with even as few as 5% of target subjects' data, both models' performance improves largely, with the s-PEN improving more advantage as more adaptation data become available. We assume that these (supervised) adaptation data are sufficient to constrain the feature/label space of the source models,

[3] Note, however, that the Resnet used to extract the features for these models was fine-tuned using the labeled source data.

	n[%]	0	5	10	20	30	50
Valence	s-SN	**0.72**	**0.80**	0.80	0.82	0.82	0.82
	s-PEN	0.71	**0.80**	**0.82**	**0.84**	**0.85**	**0.86**
	t-SN	N/A	0.71	0.77	0.8	0.81	0.82
	t-PEN	N/A	0.75	0.79	0.81	0.82	0.83
Arousal	s-SN	**0.66**	**0.79**	0.80	0.81	0.82	0.82
	s-PEN	0.65	**0.79**	**0.81**	**0.83**	**0.84**	**0.85**
	t-SN	N/A	0.73	0.77	0.79	0.81	0.81
	t-PEN	N/A	0.76	0.79	0.8	0.81	0.82

Table 4: Performance on target test data in terms of CCC after adapting the networks with n% of (non-overlapping) target data.

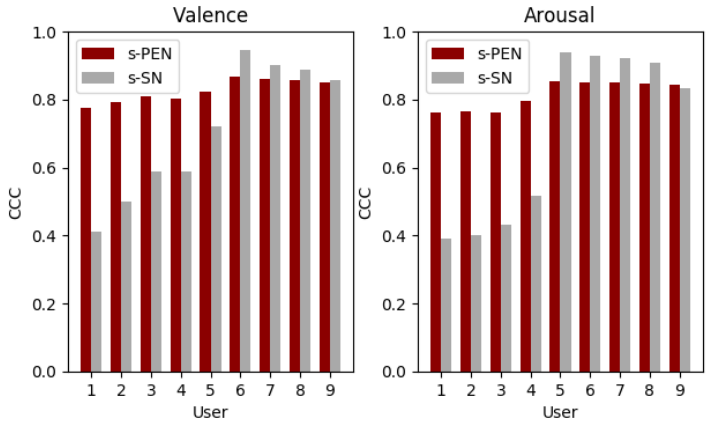


Fig. 3: Per-subject valence and arousal estimation performance on target test data of source-trained models adapted with limited target data. The s-PEN model has more consistent performance than the s-SN model over all of the target subjects.

rendering more efficient models for the target population. We also note the high performance of the t-SN and t-PEN, even with only 5% of the target data. We attribute this to the fact that they use the same ResNet feature extractor that was fine-tuned (via the s-SN) with the source labeled data. At the same time, we also note that s-SN and s-PEN have been trained on significantly more data. This, in turn, allowed the s-PEN to outperform the t-PEN by larger margin than is the case with their SN versions.

To analyze the effects of the model personalization to the target population, in Fig. 3 we show the CCC values of the s-SN and s-PEN models per target subject. We averaged the per-subject CCC values for both valence and arousal across all of the adaptation data sizes, and sorted the subjects based on the absolute difference in the models' performance for each subject. We observe

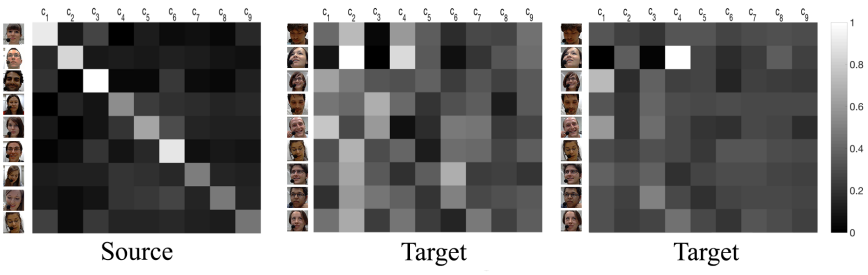


Fig. 4: Sparse Combinations of Experts via CN. Left: the selector learns the weighting of the outputs for the source subjects during source training. Center: the selector weights are effectively random yet sparse for the target subjects from source training. Right: after some fine-tuning on target, the selector begins to lose its sparse weighting despite regularization.

that on 4/9 subjects, s-PEN largely outperforms s-SN in estimation of both valence and arousal levels. On the remaining subjects, s-SN outperforms t-PEN - however, these differences are less pronounced. For instance, the s-PEN model consistently produces average CCC values for each subject that are around 0.8. This experiment shows limitations of the "one-size-fits-all" machine learning architecture on certain subjects. By contrast, the modeling flexibility of the s-PEN allows it to better fit the data distribution of the target population.

We depict the effects of our custom loss function based on the the L-1 (sparsity) regularization as well as how the CN was learning a combination of the source subjects in Fig. 4. These three plots show the progression of the average outputs of the CN weights per subject as our algorithm advances. As seen in the first image, the source weights are nearly an identity matrix after training only on the source subjects. This allowed the s-PEN to specialize each expert to one source subject, while also sharing the knowledge. However, when evaluated on target subjects, the selector produces different weights after unsupervised fine-tuning to those subjects (i.e. it is ignorant of the subject ids). Compared to the third image, these "subject" weights are still more sparse than when the training is done using data of target subjects. This is because the former uses the pre-trained selector network, resulting in the more sparse weights. By contrast, without leveraging this information, the s-PEN finds it more challenging to specialize in target subjects (as measured by the distribution of its CN weights $(c_1 \dots c_9)$) after the fine-tuning, despite the regularization of its weights (perhaps, due to the lack of subject ids). On the other hand, we found that by increasing the level of regularization adversely affects the model's performance, diminishing the role of the valence-arousal estimation loss.

Finally, in Table 5, we have included the s-PEN model's performance alongside the performance reported recently by [23], where a ResNet-50 with Gated Recurrent Unit (GRU) networks for sequence estimation is used for estimation of valence and arousal from videos of the same target subjects. We show that

Model	Valence	Arousal	Avg.
s-PEN (0% of fine-tuning data)	0.71	0.65	0.68
s-PEN (5% of fine-tuning data)	0.80	0.79	0.80
End2You	0.58	0.41	0.50
AVEC 2016 Baseline	0.61	0.38	0.50

Table 5: Comparison to End2You and AVEC 2016 Baseline [23].

the proposed s-PEN exceeds their reported performance by large margin. However, these results may not directly be comparable because of possibly different evaluation settings.

5 Conclusions

In summary, we propose a novel strategy for the personalization of deep convolutional neural networks for the purpose of valence and arousal estimation from face images. The key to our approach is the personalization of the mixture of experts architecture using a limited amount of data of target subjects. These personalized models have clear advantages over the compared single-expert ("one-size-fits-all") models in terms of how well are able to adapt to the target population when using limited amounts of labeled data from target subjects. Given the limitations involved in obtaining annotated valence and arousal data due to cost of expert labor and large variations in levels of expressiveness between people, model personalization can be key in working with limited data with many different domains. The audio-visual data we used come from sessions limited to 5 minutes, yielding ∼7k image frames per subject, and we randomly sampled and split this data into non-overalapping training, adaptation, and test sets. However, ideally the system would have access to multiple sessions, allowing the proposed model to actively personalize as the interactions progress, and give us enough sessions so that we could draw samples that are further apart in time and less likely to be correlated. While minimizing correlation is not as critical in this work as in non-personalized situations, future work should explore how different methods of pseudo-random sampling of frames for constructing the adaptation and the hold-out test sets affect the results.

Acknowledgments

The work of O. Rudovic has been funded by the European Union H2020, Marie Curie Action - Individual Fellowship no. 701236 (EngageMe).

References

1. Peterson, K., Rudovic, O., Guerrero, R., Picard, R.W.: Personalized gaussian processes for future prediction of alzheimer's disease progression. NIPS Workshop on Machine Learning for Healthcare (2017)

2. Jaques, N., Rudovic, O., Taylor, S., Sano, A., Picard, R.: Predicting tomorrow's mood, health, and stress level using personalized multitask learning and domain adaptation. In: IJCAI Workshop. (2017)
3. Zeng, Z., Pantic, M., Roisman, G.I., Huang, T.S.: A survey of affect recognition methods: Audio, visual, and spontaneous expressions. IEEE TPAMI (2009)
4. Rudovic, O., Lee, J., Dai, M., Schuller, B., Picard, R.: Personalized machine learning for robot perception of affect and engagement in autism therapy. arXiv preprint arXiv:1802.01186 (2018)
5. Mollahosseini, A., Hasani, B., Mahoor, M.H.: Affectnet: A database for facial expression, valence, and arousal computing in the wild. IEEE TAC (2017)
6. Martinez, D.L., Rudovic, O., Picard, R.: Personalized automatic estimation of self-reported pain intensity from facial expressions. In: IEEE CVPR'W. (2017)
7. Csurka, G.: Domain adaptation for visual applications: A comprehensive survey. Advances in Computer Vision and Pattern Recognition (2017)
8. Jacobs, R.A., Jordan, M.I., Nowlan, S.J., Hinton, G.E.: Adaptive mixtures of local experts. Neural computation **3**(1) (1991)
9. Ringeval, F., Sonderegger, A., Sauer, J., Lalanne, D.: Introducing the recola multimodal corpus of remote collaborative and affective interactions. In: IEEE FG (Workshops). (2013)
10. Valstar, M., Gratch, J., Schuller, B., Ringeval, F., Lalanne, D., Torres Torres, M., Scherer, S., Stratou, G., Cowie, R., Pantic, M.: Avec 2016: Depression, mood, and emotion recognition workshop and challenge. In: Proc. of the 6th International Workshop on Audio/Visual Emotion Challenge, ACM (2016)
11. Collobert, R., Bengio, S., Bengio, Y.: A parallel mixture of svms for very large scale problems. In: NIPS. (2002)
12. Shahbaba, B., Neal, R.: Nonlinear models using dirichlet process mixtures. JMLR (2009)
13. Theis, L., Bethge, M.: Generative image modeling using spatial lstms. In: NIPS. (2015)
14. Yao, B., Walther, D., Beck, D., Fei-Fei, L.: Hierarchical mixture of classification experts uncovers interactions between brain regions. In: NIPS. (2009)
15. Rasmussen, C.E., Ghahramani, Z.: Infinite mixtures of gaussian process experts. In: NIPS. (2002)
16. Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G., Dean, J.: Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In: ICLR. (2017)
17. Jiang, H., Learned-Miller, E.: Face detection with the faster r-cnn. In: IEEE FG. (2017)
18. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE CVPR. (2016)
19. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NIPS. (2012) 1097–1105
20. Jiang, J.: A literature survey on domain adaptation of statistical classifiers. URL: http://sifaka. cs. uiuc. edu/jiang4/domainadaptation/survey **3** (2008)
21. Chollet, F., et al.: Keras (2015)
22. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al.: Tensorflow: A system for large-scale machine learning. In: OSDI. (2016)
23. Tzirakis, P., Zafeiriou, S., Schuller, B.W.: End2you–the imperial toolkit for multimodal profiling by end-to-end learning. arXiv preprint arXiv:1802.01115 (2018)

7. Contributions

In this thesis, I made two contributions to the field of personalized machine learning. They are as follows:

1. I created a training algorithm that involves training the ResNet and regression layer(s) together followed by freezing the ResNet weights and fine-tuning the weights of layers that appear deeper in the network. This training strategy was created upon noticing that fitting the ResNet and regression layer(s) together boosts performance relative to fitting the regression layer(s) alone.

2. I demonstrated that personalization improves upon state-of-the-art performance given by one-size-fits-all models (provided that the selector for weighting the experts is optimized properly).

8. Limitations

Some of the limitations of the research performed in producing this thesis have been mentioned in the previous sections. Nevertheless, I will briefly summarize all of the limitations of my work below (including those limitations previously mentioned):

1. **Selector Training:** In the case of the AffectNet dataset, when simulating a selector with perfect knowledge (i.e. it predicts every given emotion with 100% accuracy), the resulting predictor surpasses the one-size-fits-all model by a large margin. Other experiments performed have deduced that emotion expression classification needs to be 80% accurate to improve arousal estimates and 85% accurate to improve valence estimates. However, I have been unable to learn a selector that meets these criteria via gradient descent. More analysis and research into why this is happening should be addressed in future work. Such future work could even produce novel tools and methods of approaching the problems addressed in this research (e.g. multiple models currently exist for estimating emotion expressions from images that could potentially be used in-place of this learned selector; while professional valence and arousal annotators typically disagree regarding the exact values of valence and arousal in a given image, even members of the general public would less likely disagree regarding the emotion expressions contained in an image, suggesting that a tool that takes an emotion and face as an input and returns the contained valence and arousal estimates based on the given emotion would be useful).

2. **Temporal Correlation:** When working with AVEC, videos were split into frames and randomly shuffled to form datasets as described in a previous section. However, from a machine learning standpoint, this could cause issues in that very similar frames (such as two frames from the same second) could appear in both the training and validation sets, meaning that our models could have essentially seen frames during evaluation that were part of its training set. Therefore, this could have resulted in performance values that were

better than what the values should have been due to not taking temporal correlation into account. Future work should be done to shuffle in a way that mitigates this issue.

3. **Training Set Size Differences:** When evaluating the effects of learning with a small amount of target data, models originally trained with source data and then fine-tuned with target data were compared to models learned from target data alone. It was found that training on source first before fine-tuning with target data improves performance, but it is important to note that this phenomenon could be simply due to the sheer number of extra training examples a network trained on source would observe compared to the small amount of examples a network trained only on limited target data would observe. At the same time, target-trained networks performed very well with even small amounts of target data; this could be attributed to the fact that even though such networks had regression layers that had not been trained on source data, they still used features from ResNet architectures that had weights from training on source data. More analysis of the effects of adaptation with target data versus training entirely from scratch with the same amount of target data should be performed in light of these observations.

4. **Population Adaptation versus Person Adaptation:** Experiments with AVEC were performed in a way such that one expert was trained per source subject, but when it came to adapting to target subjects, all of the experts were adapted to *all* of the target subjects together (i.e. their data were pooled and the experts and selector were together adapted to all of this pooled data to minimize differences between weighted sum of expert outputs and the ground truth). This is still a valid method of semi-supervised domain adaptation, given that this method does not utilize the subject ids during training, but this cannot be called an instance of *personalized* domain adaptation because the target domain in this case is a *population* of target subjects and not an individual target subject. Future work should be done to observe how well this training algorithm performs when the target domain consists of data from a single person instead of data from a population.

5. **ResNet Feature Vector Size:** While working with both the AffectNet and AVEC datasets, the experts and selector layers (or just the sole shared network in the case of the general model) received input directly from the ResNet. The output of the ResNet has width 2048, meaning that any layer performing regression to valence and arousal outputs is a 2048 x 2 weight matrix, and the selector layer is of size 2048 x n (where n is the number of experts). While it is possible to find locally optimal solutions for these matrices via gradient descent, this does not allow for much flexibility while finding the solution. Therefore, more experiments should be done that add additional dense layers with nonlinear activation functions (such as ReLU) that shrink the feature dimension in-between the ResNet and any layers previously receiving ResNet features as input to examine the effects of having better-processed features along with a smaller number of weights to use in the last layer.

9. Future Work

There are at least two additional areas where future work can take place aside from addressing the limitations discussed above. One is to work with more datasets (especially since the use of only two datasets could be seen as a limitation as well). An additional database with which experimentation and evaluation would be useful is the Automatic Sentiment Analysis in the Wild (SEWA) database [27]. This database can be thought of as a larger version of the AVEC dataset, with which more experiments and different types of experiments can be performed. First, the SEWA database contains data from 398 subjects that are together from 6 different cultures [27]. From a personalization perspective, it would be interesting to examine how the training strategy employed with these other datasets performs. Specifically, one would be able to test how well models obtained through "personalization" to a culture, personalization within a culture, and personalization irrespective of culture perform against each other.

A second dataset that would be great for testing against the methods used in this paper was obtained partly through a collaboration between two of the MIT Media Lab groups: the Affective Computing Group and the Personal Robots Group. Through this collaboration as well as with the Affective Computing Group's own resources, the Affective Computing Group obtained videos of children interacting with robots, specifically a Tega robot created by the Personal Robots Group and a NAO robot owned by the Affective Computing Group. Two types of children participated as subjects in this study: children diagnosed with autism and children without autism. These videos were professionally annotated with engagement levels and valence and arousal measures, and they were additionally annotated via the OpenFace toolkit with information such as locations of the facial landmarks and bounding boxes [19]. All of this data and information is currently stored in the Affective Computing Group's data storage.

Due to the differences between the ways in which children of the two groups express emotions and engagement, a model for predicting engagement, valence, and/or arousal given video frames that was trained on all of the data would be expected to perform poorly. If a type of personalization through the techniques discussed were to be applied, one would hope that a model would be able to control for the differences between types of children and accurately predict engagement and expressed emotions.

Aside from training and testing models on more datasets, there is another technique that would be interesting to explore as a potential personalization and domain adaptation method. This technique is known in the academic literature as "student-teacher learning". It involves first training a complex network (the "teacher") on a dataset and obtaining predictions from that network. Then, a simpler network (the "student") is trained on the same dataset except that the labels used are the *predictions* of the more complex network. By training the student to predict like the teacher on the given data, this method accomplishes the task of "model compression", which trains a simple model to perform like a complex one.

Student-teacher learning and model compression typically are not domain adaptation techniques. However, a recent paper discusses a "novel domain adaptation method for DNN acoustic models

based on the knowledge distillation framework" [28]. The research in this paper comes from a different context in that it deals with audio data instead of image data and uses classification models to predict discrete classes instead of regression models to predict over a continuous range of values. At the same time, it should be possible to derive an analogous algorithm from the methods used in their work that can be utilized with the problems explored here. Adapting their methods to problems of valence and arousal estimation from face images is definitely an area for future work.

10. Conclusion

As machine learning becomes more powerful, its applications grow more numerous as well. Facial expression analysis is one of the latest applications of machine learning in the realm of affective computing. If machine learning systems were able to measure engagement and expressed emotions based on facial expressions, then as a minimum, human-robot interactions could be greatly improved, and systems that improve users' emotional states could be more reliably developed. Unfortunately, accurate estimation of expressed emotions based on facial expressions is difficult to do both due to the limited amount of labeled training data and the numerous differences between various types of people. Personalized machine learning appears to be one solution to this problem, and domain adaptation may be the way to personalize models to give more accurate results both on seen and unseen data. However, the exact manner in which the domain adaptation should or could be performed is still an open question.

My research sought to find the best ways to perform this model adaptation with the end goal of creating a personalized machine learning system that can accurately predict valence and arousal given images of facial expressions. The end system's change in accuracy relative to a baseline model and required amount and type of training data provided important insight into ways in which personalization can be achieved in related fields. Additionally, my research showed the benefits and drawbacks of using a mixture of experts approach to perform this domain adaptation. Based on the results from working with the AffectNet and AVEC datasets, the mixture of experts approach shows promise in that models that employ it perform as well as baseline models if not better, but more research needs to be done with regard to exploring how to effectively train the selector. This is definitely something to research when working with more datasets and when changing the network architecture to reduce the number of features. At any rate, the methods discussed here have contributed a new technique and foundation of a solution to the problem of the estimation of valence and arousal from face images.

References

1. Jiang, Jing. “A Literature Survey on Domain Adaptation of Statistical Classifiers.” http://sifaka.cs.uiuc.edu/jiang4/domain_adaptation/survey/
2. Beijbom, Oscar. “Domain Adaptations for Computer Vision Applications.” CoRR. 2012.
3. Blitzer, John. “Domain Adaptation with Coupled Subspaces.” PMLR. 2011. Version 15. Pages 173-181
4. Q. Chen, J. Huang, R. Feris, L. M. Brown, J. Dong and S. Yan, “Deep domain adaptation for describing people based on fine-grained clothing attributes,” *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, 2015, pp. 5315-5324.Shabo A, Scarpa M. Bridging the informatics gap between bench and bedside: implications to neurodegenerative diseases. In Neurodegenerative Diseases: Integrative PPPM Approach as the Medicine of the Future. 2013:301-8.
5. Daumé, Hal III. “Frustratingly Easy Domain Adaptation.” CoRR. 2009
6. Ganin, Yaroslav and Lempitsky, Victor. “Unsupervised Domain Adaptation by Backpropagation.” PMLR. 2014.
7. B. Gong, Y. Shi, F. Sha and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," *2012 IEEE Conference on Computer Vision and Pattern Recognition*, Providence, RI, 2012, pp. 2066-2073.
   doi: 10.1109/CVPR.2012.6247911
8. Kodirov, Elyor, Tao Xiang, Zhen-Yong Fu and Shaogang Gong. “Unsupervised Domain Adaptation for Zero-Shot Learning.” *2015 IEEE International Conference on Computer Vision (ICCV)* (2015): 2452-2460.
9. Long, Mingsheng, Jiamin Wang, and Michael I. Jordan. “Unsupervised Domain Adaptation with Residual Transfer Networks.” CoRR. 2016.
10. Taigman, Yaniv, Adam Polyak, and Lior Wolf. “Unsupervised Cross-Domain Image Generation.” CoRR. 2016.
11. *Tuia* D, Camps-Valls G (2016) *Kernel Manifold Alignment* for Domain Adaptation. PLoS ONE 11(2): e0148655. https://doi.org/10.1371/journal.pone.0148655.
12. Yamada, M., Sigal, L. & Chang, Y. “Domain Adaptation for Structured Regression.” Int J Comput Vis (2014) 109: 126. https://doi.org/10.1007/s11263-013-0689-x
13. Jacobs, Robert A., Michael I. Jordan, Steven J. Nowlan and Geoffrey E. Hinton. “Adaptive Mixtures of Local Experts.” *Neural Computation* 3 (1991): 79-87
14. Shazeer, Noam, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. “Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer.” CoRR. 2017.
15. Zen, Gloria, Enver Sangineto, Elisa Ricci and Nicu Sebe. “Unsupervised Domain Adaptation for Personalized Facial Emotion Recognition.” *ICMI* (2014).
16. Sangineto, Enver. Gloria Zen, Elisa Ricci and Nicu Sebe. “We are not All Equal: Personalizing Models for Facial Expression Analysis with Transductive Parameter Transfer.” *ACM Multimedia* (2014).
17. Wachinger C, Reuter M, Alzheimer's Disease Neuroimaging Initiative, Australian Imaging Biomarkers and Lifestyle flagship study of ageing. “Domain adaptation for Alzheimer's disease diagnostics**.** *Neuroimage.” 2016 Oct 1;139:470-9.*
18. Ali Mollahosseini, Behzad Hasani, and Mohammad H. Mahoor, “AffectNet: A New Database for Facial Expression, Valence, and Arousal Computation in the Wild”, IEEE Transactions on Affective Computing, 2017.

19. T. Baltrušaitis, P. Robinson and L. P. Morency, "OpenFace: An open source facial behavior analysis toolkit," *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Lake Placid, NY, 2016, pp. 1-10.
20. He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Deep Residual Learning for Image Recognition.” CoRR. 2015.
21. Jaques, N., Rudovic, O., Taylor, S., Sano, A., and Picard, R. "Predicting Tomorrow's Mood, Health, and Stress Level using Personalized Multitask Learning and Domain Adaptation," Proceedings of Machine Learning Research, 48, 17-33. August 2017.
22. Peterson, K., Rudovic, O., Guerrero, R., Picard, R. "Personalized Gaussian Processes for Future Prediction of Alzheimer's Disease Progression," NIPS Workshop on Machine Learning for Healthcare, Long Beach, CA, December 2017.
23. Rudovic, O., Lee, J., Mascarell-Maricic, L., Schuller, B., Picard, R. "Measuring Engagement in Robot-assisted Autism Therapy: A Cross-cultural Study," Frontiers in Robotics and AI,Ê4, 36, July 2017.
24. Ringeval, Fabien, Andreas Sonderegger, Juergen Sauer, and Denis Lalanne. "Introducing the RECOLA multimodal corpus of remote collaborative and affective interactions." In *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*, pp. 1-8. IEEE, 2013.
25. Valstar, Michel, Jonathan Gratch, Björn Schuller, Fabien Ringeval, Denis Lalanne, Mercedes Torres Torres, Stefan Scherer, Giota Stratou, Roddy Cowie, and Maja Pantic. "Avec 2016: Depression, mood, and emotion recognition workshop and challenge." In *Proceedings of the 6th International Workshop on Audio/Visual Emotion Challenge*, pp. 3-10. ACM, 2016.
26. Jiang, Huaizu, and Erik Learned-Miller. "Face detection with the faster R-CNN." In *Automatic Face & Gesture Recognition (FG 2017), 2017 12th IEEE International Conference on*, pp. 650-657. IEEE, 2017.
27. http://www.sewaproject.edu/
28. Asami, Taichi, Ryo Masumura, Yoshikazu Yamaguchi, Hirokazu Masataki, and Yushi Aono. "Domain adaptation of DNN acoustic models using knowledge distillation." In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pp. 5185-5189. IEEE, 2017.
29. Rudovic, Ognjen, Jaeryoung Lee, Miles Dai, Bjorn Schuller, and Rosalind Picard. "Personalized machine learning for robot perception of affect and engagement in autism therapy." *arXiv preprint arXiv:1802.01186* (2018).
30. Deng, Jia, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. "Imagenet: A large-scale hierarchical image database." In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 248-255. IEEE, 2009.
31. Feffer, M., Rudovic, O., Picard, R. "A Mixture of Personalized Experts for Human Affect Estimation." *The 14th International Conference on Machine Learning and Data Mining (MLDM)*. July 2018.

Appendix

Code Guide

Function	Description
`fit_model`	The purpose of this is to fine tune the Resnet to target context (dataset) and also extract the deep features for training more sophisticated models for affect estimation.
`get_preds_from_model`	Given model weights, obtains output features and predicted valence and arousal values and saves them to a csv file
`fit_reduction_network`	Given ResNet features and corresponding labels, trains a series of dense layers and gets dimension-reduced output features and saves them to a file
`train_experts`	Given reduced features and corresponding labels, trains a mixture of experts architecture (in a supervised or unsupervised way based on settings) and saves output features and predictions to a file

CSV File Structure

Each row in a CSV file consists of entries corresponding to an input example that can be grouped as follows (from left to right):

```
pid | features | labels | predicted labels | group flag
```

Column Group	Description
`pid`	Unique identifier
`features`	ResNet output features
`labels`	Ground-truth labels for example
`predicted labels`	Regression layer predictions
`group flag`	Train/Val/Test group indicator