# Sparse Autoencoder-based Feature Transfer Learning for Speech Emotion Recognition

Jun Deng[1], Zixing Zhang[1], Erik Marchi[1], Björn Schuller[2,1,3]
[1]Machine Intelligence & Signal Processing Group, MMK,Technische Universität München, Germany
[2]Institute for Sensor Systems, University of Passau, Germany
[3]Department of Computing, Imperial College London, London, U.K.
{jun.deng,zixing.zhang,erik.marchi}@tum.de, bjoern.schuller@uni-passau.de

***Abstract*—In speech emotion recognition, training and test data used for system development usually tend to fit each other perfectly, but further 'similar' data may be available. Transfer learning helps to exploit such similar data for training despite the inherent dissimilarities in order to boost a recogniser's performance. In this context, this paper presents a sparse autoencoder method for feature transfer learning for speech emotion recognition. In our proposed method, a common emotion-specific mapping rule is learnt from a small set of labelled data in a target domain. Then, newly reconstructed data are obtained by applying this rule on the emotion-specific data in a different domain. The experimental results evaluated on six standard databases show that our approach significantly improves the performance relative to learning each source domain independently.**

***Index Terms*—speech emotion recognition; transfer learning; sparse autoencoder; deep neural networks**

## I. Introduction

Speech emotion recognition focuses on using acoustic and linguistic parameters as features and classifiers as tools to predict the 'correct' emotional states [1]. In this task — just as in almost any other pattern recognition task — better performance is usually achieved using training data from the same session or corpus. Building a feature representation is an opportunity to incorporate domain knowledge into the data and can be very application specific. For that reason, much of the actual effort in deploying systems of speech emotion recognition goes into the design of an appropriate representation of the data in order to support promising classification performance. At present, several emotional speech corpora exist, but they are typically highly dissimilar in terms of spoken language, type of emotion such as acted, elicited, or naturalistic, or type of labelling scheme such as categorical or dimensional [2]. When labelling emotional corpora, even worse, there is no certain ground truth but a subjective ambiguous 'gold standard' as given by majority voting of several human raters which may be in considerable disagreement. To reduce human label effort, either to annotate new data or bridge the gap between corpora annotated in different ways, speech emotion recognition is in need of a method to reuse existing corpora and retrieve useful information within corpora for a related target task.

Recently, transfer learning has been proposed to develop methods to transfer useful information in one or more source tasks to a related target task [3]. It has been empirically and theoretically shown that transfer learning can greatly improve the learning performance especially when only a small number of data are available in a target domain [3], [4]. Speech emotion recognition can benefit from using transfer learning as well. As an example, the labelled corpus may be acted speech obtained through previous human labelling efforts. For a classification task on a newly spontaneous corpus where the data's features or data's distributions may be different, there may be a lack of labelled training data. As a result, one may not be able to directly apply the emotion models learnt on the acted speech to the new spontaneous data. In such cases, it would be helpful if we could transfer the classification knowledge into the new domain.

In this paper, we thus propose feature transfer learning based on a sparse autoencoder method — a type of neural network with sparseness constraints on hidden units — for discovering knowledge in acoustic features from small target data to improve performance of speech emotion recognition when applying the knowledge to source data. Our approach to feature learning via a sparse autoencoder consists of two stages: First we learn a representation using a single-layer autoencoder trained on class-specific instances from target data. Then, we apply this representation to source data with respect to the specific class for reconstructing those data, and use it for the classification task in the way of building standard emotional models.

The remainder of this paper is organised as follows. Section II discusses related work. In Section III, we briefly introduce the six chosen databases and acoustic features used. We then present the sparse autoencoder-based method in Section IV. Experiments on the six standard corpora are demonstrated in Section V. Finally, we draw a conclusion and point out future work in Section VI.

## II. Related Work

Recently, increasing attention has been drawn to the study of the emotional content of speech signals, and hence, many systems have been proposed to identify the emotional content of a spoken utterance [1]. Most studies tend to overestimation in this respect: Acted data are often used rather than spontaneous data, results are reported on preselected prototypical data, and true speaker disjunctive partitioning is still less common than simple cross-validation. Even speaker disjunctive evaluation can give only little insight into the generalization ability of

today's emotion recognition engines since training and test data as used for system development usually tend to be similar as far as recording conditions, noise overlay, language, and types of emotions are concerned [2]. For example, if a system builds upon a classifier using features extracted from adults' speech corpora to identify children's emotional state, its performance can be expected as very low. In this example, this comes, as — among other factors — there is a relevant difference of certain low level descriptors (LLDs) such as pitch between adults and children on which emotion phenomena relies heavily.

Transfer learning has been proposed to deal with the significant problem of how to reuse the knowledge learnt before from other data or features. Pan and Yang demonstrated several practical examples to illustrate transfer learning's role [4]. Among the various ways of transfer learning, deep neural networks that have many hidden layers and are trained using new methods have shown to suit well to transfer learning [5]. Previous work [6], [7] further demonstrated that a deep architecture is necessary to represent many functions compactly and such architectures lead to useful representations that ideally disentangle the factors of variation present in the input. Glorot et al. applied a stacked denoising autoencoder with sparse rectifiers to domain adaption in large-scale sentiment analysis [8]. Another successful application of deep neural networks arises in the field of speech recognition to exploit information in neighbouring frames and from using tied context-dependent states for acoustic modelling, which outperforms state-of-the-art methods on a variety of speech recognition benchmarks, sometimes by a large margin [9], [10].

Furthermore, deep neural networks have been also analysed in speech emotion recognition. In [11], a generalised discriminant analysis based on deep neural networks was proposed to learn discriminative features of low dimension for optimisation from a large set of acoustic features for emotion recognition which slightly rises the benchmark. Brueckner and Schuller successfully investigated the applicability of deep belief networks on the Likability Sub-Challenge of the Interspeech 2012 Speaker Trait Challenge [12].

Sparseness plays a key role in learning gabor-like filters, thus Lee et al. presented a sparse variant of the deep belief networks proposed by Hinton et al. [13] which faithfully mimics certain properties of the visual area V2 in the cortical hierarchy [14]. The first layer of the network results in localised, oriented, edges filters. Apparently, the network can effectively discover high-level features in the data but the higher-level features learnt on other data may not be suitable for the target task. Sometimes, such networks may fail due to large difference between source features and target features in speech emotion recognition. We therefore consider introducing a sparse autoencoder as a link to reconstruct source data in accordance with common feature structure learnt on small target data, which results in the information transfer from source domain to target domain.

TABLE I: Emotion categories mapping onto negative and positive valence for six databases.

| Corpus | Negative | Positive |
|---|---|---|
| **FAU AEC** | angry, touchy, emphatic, reprimanding | motherese, joyful, neutral, rest |
| **TUM AVIC** | boredom | neutral, joyful |
| **EMO-DB** | anger, boredom, disgust, fear, sadness | joy, neutral |
| **eNTERFACE** | anger, disgust, fear, sadness | joy, surprise |
| **SUSAS** | high stress, screaming, fear | medium stress, neutral |
| **VAM** | q4, q3 | q2, q1 |

*Abbreviations: q1–q4: quadrants in the arousal-valence plane*

TABLE III: Overview of the standardised feature set provided by the INTERSPEECH 2009 EC.

| **LLDs** (16 × 2) | **Functionals** (12) |
|---|---|
| (Δ) ZCR | mean |
| (Δ) RMS Energy | standard deviation |
| (Δ) F0 | kurtosis, skewness |
| (Δ) HNR | extremes: value, rel, position, range |
| (Δ) MFCC 1–12 | linear regression: offset, slope, MSE |

## III. Databases

To investigate the performance of the proposed method, we consider the INTERSPEECH 2009 Emotion Challenge (EC) two-class task [15] as target. It is based on the FAU Aibo Emotion Corpus (FAU AEC), which contains recordings of children interacting with the pet robot Aibo in German speech. In the training set there are 6 601 instances of positive valence and 3 358 instances of negative valence, and in the test set one finds 5 792 instances of positive valence and 2 465 instances of negative valence.

Besides, another five publicly available and highly popular databases, namely the TUM Audio-Visual Interest Corpus (TUM AVIC) [16], Berlin Emotional Speech Database (EMO-DB) [17], eNTERFACE [18], Speech Under Simulated and Actual Stress (SUSAS) [19], and the "Vera am Mittag" (VAM) database [17], [20] are chosen as source set, which are different from the target database FAU AEC in terms of speaker age, partially spoken language, type of emotion, type of recording situation, and of course annotators and subjects. For the comparability with FAU AEC, we additionally map the diverse emotion groups onto the valence axis in the dimensional emotion model. The mapping defined in [2] for cross-corpus experiments is used to generate labels for binary valence from the emotion categories in order to generate a unified set of labels. This mapping is given in Table I. In addition, Table II summarises the six standard databases.

### A. Acoustic Features

To keep in line with the INTERSPEECH 2009 EC [15], we decided to use its standardised feature set of 12 functionals applied to 2 × 16 acoustic Low-Level Descriptors (LLDs)

TABLE II: Summary of the six chosen databases.

| Corpus | Age | Language | Speech | Emotion | # Valence - | # Valence + | # All | h:mm | #m | #f | Rec | Rate kHz |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **FAU AEC** | children | German | variable | natural | 5 823 | 12 393 | 18 216 | 9:20 | 21 | 30 | normal | 16 |
| **TUM AVIC** | adults | English | variable | natural | 553 | 2 449 | 3 002 | 1:47 | 11 | 10 | studio | 44 |
| **EMO-DB** | adults | German | fixed | acted | 352 | 142 | 494 | 0:22 | 5 | 5 | studio | 16 |
| **eNTERFACE** | adults | English | fixed | induced | 855 | 422 | 1 277 | 1:00 | 34 | 8 | normal | 16 |
| **SUSAS** | adults | English | fixed | natural | 1 616 | 1 977 | 3 593 | 1:01 | 4 | 3 | noisy | 8 |
| **VAM** | adults | German | variable | natural | 876 | 71 | 947 | 0:47 | 15 | 32 | noisy | 16 |

***Age** (adults or children). Number of utterances per binary valence (# **Valence**, Negative (-), Positive (+)), and overall number of chunks (# **All**). Total audio time. Number of female (**#f**) and male (**#m**) subjects. **Recording** conditions (studio/normal/noisy). Sampling **Rate**.*

including their first order delta regression coefficients as shown in Table III. In detail, the 16 LLDs are zero-crossing-rate (ZCR) from the time signal, root mean square (RMS) frame energy, pitch frequency (normalised to 500 Hz), harmonics-to-noise ratio (HNR) by autocorrelation function, and Mel-frequency cepstral coefficients (MFCC) 1–12. Then, 12 functionals — mean, standard deviation, kurtosis, skewness, minimum and maximum value, relative position, and ranges as well as two linear regression coefficients with their mean square error (MSE) — are applied on the chunk level. Thus, the total feature vector per chunk contains $16 \times 2 \times 12 = 384$ attributes. To ensure reproducibility as well, the open source toolkit openEAR toolkit [21] was used to extract the feature set with the pre-defined openEAR configuration for the 2009 challenge.

## IV. Approach

Speech is produced by modulating a relatively small number of parameters of a dynamical system [22], [23], and this implies that its true underlying structure is much lower-dimensional than is immediately apparent in a window that contains hundreds of coefficients [10]. We believe, therefore, that speech emotional features also have such underlying structure if there is a method that can effectively exploit information embedded in a large data set. To allow for feature transfer learning, we use the underlying feature structure learnt from target data to reconstruct other source data accordingly and preserve the data's own information as much as possible. The single-layer sparse autoencoder is used to exploit the underlying feature structure on target data, represented by a set of weight matrices and a bias vector. We input a given source data to the learnt sparse autoencoder structure to reconstruct its own. In this section, we describe briefly the single-layer autoencoder, and then present in detail the sparse autoencoder feature transfer learning.

Let us assume a given target training set of $n_t$ examples $T_t = \{(x_1^t, y_1^t), \ldots, (x_{n_t}^t, y_{n_t}^t)\}$ drawn from some distribution $D$, and a source training set of $n_s$ examples $T_s = \{(x_1^t, y_1^t), \ldots, (x_{n_s}^t, y_{n_s}^t)\}$. Here, the target training set and the source share the same feature space and label space, i. e., each input feature vector $x_i \in \mathbf{R}^n$, and the corresponding class label $y_i \in \{C_1, \ldots, C_L\}$. However, we do not assume that the target data $T_t$ was drawn from the same distribution as the source data $T_s$, which means the classifiers learnt from the source set cannot classify the (target) test data well due to different data distribution. In addition, the size of $T_t$ is often inadequate to train a good classifier for the test data. Transfer learning aims to help improve the learning of the target predictive function in $T_t$ using the knowledge in $T_s$ [4].

### A. Single-layer autoencoder

A single-layer autoencoder, which is a kind of neural network consisting of only one hidden layer, sets the target values to be equal to the input. Deep neural networks use it, as an element, to find common data representation from the input [7], [24]. Formally, in response to an input example $x \in \mathbf{R}^n$, the activation of each neuron, $h_i, i = 1, \ldots, m$ is

$$h(x) = f(W_1 x + b_1), \tag{1}$$

where $f(z) = 1/(1 + \exp(-z))$ is the non-linear activation function applied component-wise, $h(x) \in \mathbf{R}^m$ is the vector of neuron activation, $W_1 \in m \times n$ is a weight matrix, and $b_1 \in \mathbf{R}^m$ is a bias vector. The network output is then:

$$\tilde{x} = f(W_2 h(x) + b_2), \tag{2}$$

where $\tilde{x} \in \mathbf{R}^n$ is a vector of output values, $W_2 \in n \times m$ is a weight matrix, and $b_2 \in \mathbf{R}^n$ is a bias vector.

Given a set of $p$ input examples $x_i, i = 1, \ldots, p$, the weight matrices $W_1$ and $W_2$ and the bias vector $b_1$ and $b_2$ are adapted using back-propagation to minimise the reconstruction error $\sum_{i=1}^{p} \|x_i - \tilde{x}_i\|^2$. Further, we constrain the expected activation of the hidden units to be sparse following the method of [14], i. e., we add a regularisation term that penalises a deviation of the expected activation of the hidden units from a (low) fixed level $\rho$. Thus it turns out to be the following optimisation problem:

$$minimise \sum_{i=1}^{p} \|x_i - \tilde{x}_i\|^2 + \beta \sum_{j=1}^{m} \mathrm{SP}(\rho || \hat{\rho}_j) \tag{3}$$

where $\mathrm{SP}(\rho||\hat{\rho}_j) = \rho \log \frac{\rho}{\hat{\rho}_j} + (1-\rho) \log \frac{1-\rho}{1-\hat{\rho}_j}$ is a sparse penalty term, $\hat{\rho}_j = \frac{1}{p} \sum_{i=1}^{p} h_j(x_i)$ is the average activation of hidden unit $j$ (averaged over the training set), $\rho$ is a sparsity level, and $\beta$ controls the weight of the sparsity penalty term. If the number of hidden units $m$ is less than the number of

**Algorithm 1** Sparse Autoencoder Feature Transfer Learning

**Input:** The two labelled data sets $T_t$ and $T_s$, and the corresponding class labels $C_1, \ldots, C_L$.
**Output:** Learnt classifier $\mathcal{H}$ for the target task.

```
Initialise reconstructed data T̃_s = ∅.
for l = 1 to L do
    Initialise a single-layer autoencoder SA^l(W, b).
    Choose class-specific examples T_t^{C_l} from T_t.
    Train SA^l(W, b) using T_t^{C_l}.
    Choose class-specific examples T_s^{C_l} from T_s.
    Reconstruct data T̃_s^{C_l} = SA^l_Recon(T_s^{C_l}).
    Update the reconstructed data T̃_s = T̃_s ∪ T̃_s^{C_l}.
end for
Learn a classifier H by applying supervised learning
    algorithm s (e. g., SVM) to the reconstructed data T̃_s.
return The learnt classifier H.
```

input units $n$, then the network is forced to learn a compressed and sparse representation of the input.

### B. Sparse Autoencoder Feature Transfer Learning

Since speech can be segmented into units of analysis, such as, phonemes, previous works tends to learn a sparse representation in speech related tasks via stacked single-layer autoencoders. For example, Dahl et al. proposed a context-dependent model for large vocabulary speech recognition that uses deep belief networks for phone recognition [9]. This is not applicable in speech emotion recognition since common units of analysis can be hardly found. However, emotional features are highly correlated in terms of a specific emotion, thus instances with the same emotional state can be assumed to share implicitly a common structure. The single-layer autoencoder has shown the capability of discovering a common structure in the data. Motivated by this, we propose a sparse autoencoder-based feature transfer learning method.

More specifically, for each class in the target training data, we first apply a single-layer autoencoder to class-specific examples $x_i^t \in \mathbf{R}^n$ to learn a set of parameters $W_1$, $W_2$, $b_1$, and $b_2$, as described in Section IV-A. To transfer each of the class-specific examples $x_i^s \in \mathbf{R}^n$ from the source data to the target domain, we then compute features $\tilde{x}_i^s \in \mathbf{R}^n$ based on the learnt set of the parameters by solving:

$$\tilde{x}_i^s = \mathrm{SA}_{\mathrm{Recon}}(x_i^s), \tag{4}$$

where $\mathrm{SA}_{\mathrm{Recon}}(x) = f(W_2 f(W_1 x + b_1) + b_2)$ is the output of the single-layer autoencoder. Equation (4) forces the input $x_i^s$ to reconstruct itself through computing a sparse non-linear combination of the parameters learnt on the target data. The reconstructing procedure, in turn, decreases the difference between the source data and the target data, as well as completes the feature transfer from the source domain to the target domain.

A formal description of the framework is given in Algorithm 1. As can be seen from the algorithm, at each iteration step, class-specific samples in the target set are used to train a single-layer autoencoder denoted $\mathrm{SA}^l(W, b)$ which captures a general mapping structure for the input samples. Then, we move to transfering information from the source to the target domain. For the source set, samples with the corresponding class are reconstructed by using $\mathrm{SA}^l_{\mathrm{Recon}}(T_s^{C_l})$ , as described in Equation (4), according to the mapping structure learnt by the trained autoencoder $\mathrm{SA}^l(W, b)$. Next, like most speech emotion recognition systems, we use these reconstructed features as input to standard supervised classification algorithms $\mathcal{H}$ — here, Support Vector Machines (SVMs). Finally, a test partition is used to evaluate the classifier.

## V. Experiments: Speech Emotion Recognition

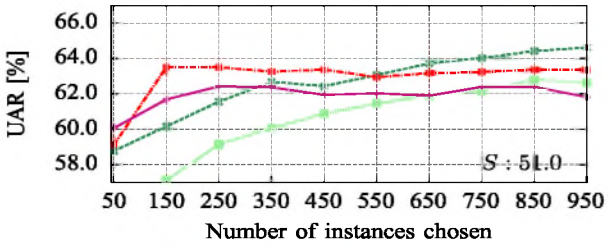
### A. Experimental Setup

In the experiments, we use SVMs as the basic supervised learner. LIBLINEAR [25] with a linear kernel is applied in the experiments to implement the SVM classifiers. The hyper-parameters of all SVMs are chosen by cross-validation on the training set. When training SVMs, furthermore, we always balance training instances between the positive and negative class by SMOTE [26]. For performance evaluation, we choose unweighted average recall (UAR), the sum of the recalls per class divided by the number of classes, which was the competition measure in the Emotion Challenge [15]. For two classes, the chance level thus always resembles 50.0 % UAR. Besides this, here the baseline UAR for the FAU AEC two-class task is 66.9 %.

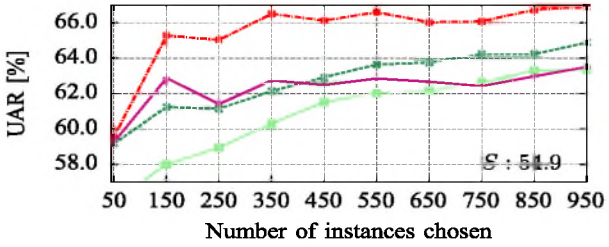
As stated, we treat FAU AEC as target set, which consists of a training and test partition (roughly half and half) naturally given by recordings at different elementary schools. To implement the sparse autoencoder algorithm, a small part of examples (the size ranging from 50 to 950 chunks) are randomly chosen from the FAU AEC training set to obtain a common feature structure, where the same number of instances are chosen from positive valence and negative valence. In the sparse autoencoder learning process, the number of hidden units was fixed to 200, and the sparsity level $\rho$ was set to 0.01. The reported performance in UAR is the average over 20 runs to avoid 'lucky' or 'unlucky' selection. Then, we use the common feature structure to reconstruct each source database, as described in Section IV-B. Finally, FAU AEC test data are classified by the classifier trained on the reconstructed data.

### B. Experimental Results

During the evaluation, we considered a variety of combinations of the target data, the reconstructed data, and the source data, in order to provide a full picture of the suggested method's effects. Figure 1 reports the results for the source data being eNTERFACE and SUSAS. Reconstructed data by the sparse autoencoder, possibly in combination with target data, significantly (one sided z-test) outperform the target data alone. For the eNTERFACE database with induced emotion type, sparse autoencoder data achieves mostly the highest test UAR when the number of chosen instances is in the range of 50 to 550. For instance, the reconstructed data's UAR obtains 63.5 % compared with only the target's UAR

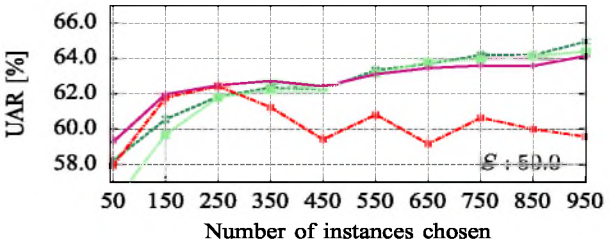



(a) eNTERFACE

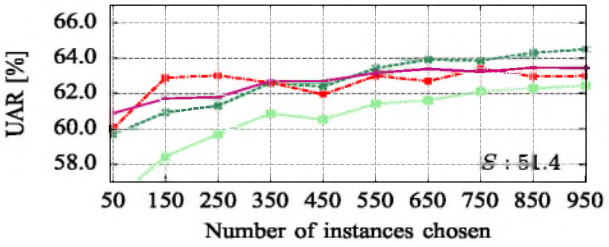



(b) SUSAS

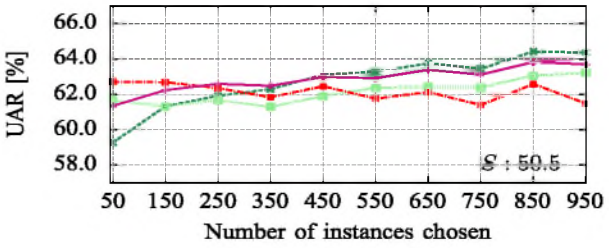

Fig. 1: UAR comparison for the increase of number of instances chosen from the FAU AEC training set for the source data eNTERFACE and SUSAS. S:##.# is the UAR if only using source data. Reconstructed: classifier trained on reconstructed source data by a sparse autoencoder method. Target + Reconstructed: classifier trained on target and reconstructed source data. Target: classifier trained on target data. Target + Source: classifier trained on target and original source data.




(a) EMO-DB




(b) VAM




(c) TUM AVIC


Fig. 2: UAR comparison for the increase of number of chosen examples from the FAU AEC training set for the source data EMO-DB, VAM, and TUM AVIC. Explanations: cf. Figure 1.

of 60.1%, the target and the reconstructed data's UAR of 61.6%, and the target and the source data's UAR of 57.1%, while 150 target instances are used. Afterwards, when the size of target training continues increasing, the performance of target data gradually overtakes the sparse autoencoder data since no more extra information in the eNTERFACE can be transferred to the FAU AEC target domain. In comparison with eNTERFACE, SUSAS's actual stress data, which is collected in a noisy recording, always obtains the highest test UAR. At 150 target instances available, the reconstructed data's UAR reaches 65.2% which is sharply larger than only the target's UAR of 61.2%, the target and reconstructed data's UAR of 62.8%, and the target and source data's UAR of 57.9%. It is worth noting that, with the increase of target training size, its UAR stably goes up to 66.8% at 950 target instances available, which approaches the baseline UAR 66.9% with the whole FAU AEC training set (9958 instances) applied.

Experimental results on the source data EMO-DB, VAM, and TUM AVIC are shown in Figure 2. As for EMO-DB with acted emotion, note that, the sparse autoencoder method cannot transfer more useful information from the source with the increase of target training size. Instead, its performance decreases unexpectedly. However, the method of combining target data with reconstructed data steadily rises in line with the size of the target data. For the source data being VAM, the sparse autoencoder method performs better within the range of target size ranging from 150 to 350. Afterwards, the performance of the sparse autoencoder is still comparable with the method of using target data. For the final source database TUM AVIC in the English language, there is not significant improvement compared to the method of using target data. Nevertheless, it is worth noting that its UAR of the reconstructed data fluctuates around 62.5%, and this UAR value (62.7%) at 50 target instances available is dramatically more than the average UAR values over the other source data

TABLE IV: UAR comparison when 50 instances are chosen from the target data. Average: 51.6% UAR (original) and 59.9% UAR (reconstructed).

| UAR [%] | AVIC | EMO-DB | eNTERFACE | SUSAS | VAM |
|---|---|---|---|---|---|
| **Original** | 50.5 | 50.0 | 51.0 | 54.9 | 51.4 |
| **Reconst.** | 62.7 | 57.9 | 59.1 | 59.5 | 60.2 |

(59.1%). If only a small number of data are available in the target domain, e.g. only 50 instances, Table IV shows UAR values for each source data and the corresponding reconstructed data. As can be seen from Table IV, when those source data as training set are input to a speech emotion recognition system, respectively, only the chance level UAR is obtained for the two-class task of FAU AEC. But the reconstructed data (average UAR 59.9%) significantly outperform the original source data (average UAR 51.6%), which means that knowledge transferred by the sparse autoencoder is useful for the classification in speech emotion recognition. The performance improvement over each original source data are large even though very few target data instances are used.

## VI. Conclusion and Future Work

In this paper, we proposed a sparse autoencoder-based feature transfer learning method of which the basic idea is to use a single-layer autoencoder to find a common structure in small target data and then apply such structure to reconstruct source data in order to complete useful knowledge transfer from source data into a target task. In this method, each single-layer autoencoder focuses on discovering non-linear generalisation of class-specific target instances. We use the reconstructed data to build a speech emotion recognition engine for a real-life task as given by the Interspeech 2009 Emotion Challenge. Experimental results with six publicly available corpora show that the proposed algorithm effectively transfers knowledge and further enhances the classification accuracy.

Future work includes extending the single-layer architecture to a deep architecture in order to further find useful information in emotional features.

## Acknowledgments

The research leading to these results has received funding from the Chinese Scholarship Council (CSC) and the European Community's Seventh Framework Programme (FP7/2007–2013) under grant agreement No. 289021 (ASC-Inclusion).

## References

[1] A. Batliner, B. Schuller, D. Seppi, S. Steidl, L. Devillers, L. Vidrascu, T. Vogt, V. Aharonson, and N. Amir, "The automatic recognition of emotions in speech," *Emotion-Oriented Systems*, pp. 71–99, 2011.

[2] B. Schuller, B. Vlasenko, F. Eyben, M. Wollmer, A. Stuhlsatz, A. Wendemuth, and G. Rigoll, "Cross-corpus acoustic emotion recognition: Variances and strategies," *IEEE Trans. on Affective Computing*, vol. 1, no. 2, pp. 119–131, 2010.

[3] L. Torrey and J. Shavlik, "Transfer learning," *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*, vol. 1, p. 242, 2009.

[4] S. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[5] Y. Bengio, "Deep learning of representations for unsupervised and transfer learning," in *Proc. ICML*, Bellevue, U. S. A., 2011.

[6] Y. Bengio and Y. LeCun, "Scaling learning algorithms towards AI," in *Large-Scale Kernel Machines*. MIT press, 2007.

[7] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Proc. NIPS*, Vancouver, Canada, 2007.

[8] X. Glorot, A. Bordes, and Y. Bengio, "Domain adaptation for large-scale sentiment classification: A deep learning approach," in *Proc. ICML*, Bellevue, U. S. A., 2011.

[9] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 30–42, 2012.

[10] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 82–97, 2012.

[11] A. Stuhlsatz, C. Meyer, F. Eyben, T. Zielke, G. Meier, and B. Schuller, "Deep neural networks for acoustic emotion recognition: raising the benchmarks," in *Proc. ICASSP*. Prague, Czech Republic: IEEE, 2011, pp. 5688–5691.

[12] R. Brückner and B. Schuller, "Likability Classification – A not so Deep Neural Network Approach," in *Proc. INTERSPEECH*, Portland, U. S. A., 2012.

[13] G. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[14] H. Lee, C. Ekanadham, and A. Ng, "Sparse deep belief net model for visual area v2," in *Proc. NIPS*, Vancouver, Canada, 2008, pp. 873–880.

[15] B. Schuller, S. Steidl, and A. Batliner, "The INTERSPEECH 2009 Emotion Challenge," in *Proc. INTERSPEECH*, Brisbane, U. K., 2009, pp. 2794–2797.

[16] B. Schuller, R. Müller, F. Eyben, J. Gast, B. Hörnler, M. Wöllmer, G. Rigoll, A. Höthker, and H. Konosu, "Being Bored? Recognising Natural Interest by Extensive Audiovisual Integration for Real-Life Application," *Image and Vision Computing*, vol. 27, no. 12, pp. 1760–1774, 2009.

[17] F. Burkhardt, A. Paeschke, M. Rolfes, W. Sendlmeier, and B. Weiss, "A database of german emotional speech," in *Proc. INTERSPEECH*, Lisbon, Portugal, 2005.

[18] O. Martin, I. Kotsia, B. Macq, and I. Pitas, "The enterface'05 audio-visual emotion database," *IEEE Workshop on Multimedia Database Management*, 2006.

[19] J. Hansen and S. Bou-Ghazale, "Getting Started with SUSAS: A Speech Under Simulated and Actual Stress Database," in *Proc. EUROSPEECH-97*, Rhodes, Greece, 1997.

[20] M. Grimm, K. Kroschel, and S. Narayanan, "The Vera am Mittag German Audio-visual Emotional Speech Database," in *Proc. ICME*, Hannover, Germany, 2008, pp. 865–868.

[21] F. Eyben, M. Wöllmer, and B. Schuller, "openEAR — Introducing the Munich Open-Source Emotion and Affect Recognition Toolkit," in *Proc. ACII*, Amsterdam, 2009, pp. 576–581.

[22] L. Deng, "Computational models for speech production," in *Computational models of speech pattern processing*. Springer, 1999, pp. 199–213.

[23] ——, "Switching dynamic system models for speech articulation and acoustics," in *Mathematical Foundations of Speech and Language Processing*. Springer, 2004, pp. 115–133.

[24] I. Goodfellow, Q. Le, A. Saxe, H. Lee, and A. Ng, "Measuring invariances in deep networks," in *Proc. NIPS*, Vancouver, Canada, 2009, pp. 646–654.

[25] R. Fan, K. Chang, C. Hsieh, X. Wang, and C. Lin, "Liblinear: A library for large linear classification," *The Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.

[26] N. Chawla, K. Bowyer, L. Hall, and W. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.