TOWARDS DISTRIBUTED RECOGNITION OF EMOTION FROM SPEECH

Wenjing Han[1,2], *Zixing Zhang*[1], *Jun Deng*[1], *Martin Wöllmer*[1], *Felix Weninger*[1], *Björn Schuller*[1]

[1]Institute for Human-Machine Communication, Technische Universität München, Germany
[2] School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China

ABSTRACT

This paper introduces an approach for performing distributed speech emotion recognition in a client-server architecture. In this architecture, the client side deals only with feature extraction, compression and bit-stream formatting, while the server side performs bit-stream decoding, feature decompression and emotion recognition, which requires more computational resources. Taking into account the trade-off between the required transmission bandwidth and recognition accuracy, we propose to employ a vector quantization approach based on independent codebooks for feature sub-spaces. Extensive test runs are conducted to reveal the impact of quantization parameters on the compression rate and recognition performance. In the result, by using a quantization strategy involving 32 subvectors and 9 bit codeword length, almost 30 times compression can be reached without a considerable increase of the error rate.

Index Terms— distributed speech emotion recognition, split vector quantization

1. INTRODUCTION

With the emergence and exponential growth of Internet, a great deal of effort has been made to integrate speech processing into Internet technology with the purpose of facilitating the interface for human users, as well as decreasing the demand for computing resources on the client side. So far, there exist several speech-based Internet applications which have been well explored and even further applied in practice, such as distributed speech recognition (DSR) focusing on understanding users' instructions by the use of the front-end speech recognition technology [1, 2], and distributed speaker verification which can simplify users' authentication operations by verifying their identity from short spoken phrases in a client-server paradigm instead of forcing them to provide passwords and personal identification numbers (PINs) [3]. Yet, in contrast to the attention paid to the above speech-based applications, we are unaware of a study that deals with speech emotion recognition (SER) over the Internet. Actually, there are at least two benefits for developing distributed SER (DSER): 1) the development of DSER would facilitate integration of SER technology into large-scale end-user applications, 2) DSER could help enhancing 'social competence' of state-of-the-art Internet interfaces. This paper focuses on the first perspective of DSER, espeically in the context of wireless networks with mobile terminals. A further advantage of DSER is that models stored on the server can be updated periodically on the server side rather than by the end-user.

Comparing to stand-alone SER, DSER involves diverse technologies including data compression, network data transmission protocols, distributed computing etc. Furthermore, similarly to other distributed applications, it must address three following criteria: 1) the solution must be inexpensive to implement on the client side, 2) the required data transmission bandwidth for emotion recognition must remain at a low level, and 3) the recognition accuracy must be (at least) approximately equal to state-of-the-art SER.

To implement a DSER system meeting the above criteria, the first problem that arises is how to distribute the components of the recognizer over the Internet. To this end, the classic client-server architecture, as adopted in the widely adopted European Telecommunications Standards Institute (ETSI) standard for DSR [4], can be a natural choice. Based on an architecture of this kind, we separate the emotion recognition processing into two parts — the feature extraction and compression front-end executed on the client side and the emotion recognition on the remote back-end, and only need to send the parameterized representation of speech instead of speech coding from client to server.

Another crucial problem is the establishment of a suitable feature compression strategy. Compared to speech recognition front-ends where a low number of frame-wise features is extracted at a high frame rate (typically, 100 13-dimensional vectors per second), emotion recognition often involves computation of functionals from frame-wise features to capture temporal variation across time periods of approximately one second. This can result in very high dimensional feature vectors; for instance, in the (relatively small) baseline feature set for the INTERSPEECH 2009 Emotion Challenge, the dimension of the features per speech sample is 384 [5]; other recent studies in emotion recognition (e. g., [9]) employ thousands of features. Thus, the required bandwidth for uncompressed transmission of emotion recognition features is comparable to speech recognition applications. In this paper, to address feature compression, we propose to employ a sub-division into feature subspaces which are encoded independently (Split Vector Quantization algorithm, SVQ). We then focus on the balance between the required transmission bandwidth and satisfactory recognition performance.

The outline of this paper is as follows. Section 2 describes the fundamental DSER architecture assumed in this study. Section 3 gives a brief description of the FAU Aibo Emotion Corpus used in our experiments. Section 4 presents the experimental setup and results. Finally, conclusions are drawn in Section 5.

2. FUNDAMENTAL ARCHITECTURE OF DSER

There are three alternative strategies in the design of general distributed speech processing. The first is client-only processing in which most speech processing is done on the client side and then the results are transmitted to the server. The second is server-only processing in which the speech signal is transmitted to the server in the form of a waveform signal and then all processing is done on the server side. The third is client-server processing. In this model front-end processing including feature extraction is located on the client

Zixing Zhang, Jun Deng and Wenjing Han are supported by a grant from the People's Republic of China.

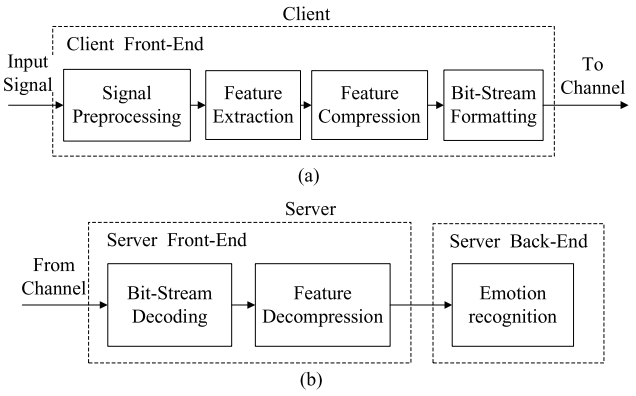

Fig. 1. The fundamental architecture of DSER. Figure (a) shows blocks implemented on the client side and (b) shows blocks implemented on the server side.

side, speech features are transmitted to the remote server, and finally the remaining processing of speech are performed on the server side [1].

The former two models both have obvious disadvantages. The client-only model requires clients that are powerful enough to perform computationally expensive speech processing. As to the server-only model, pure voice transmission requires high-speed network bandwidth, and low bandwidth connections will cause recognition performance degradation. In contrast, the client-server model moves the heavy computing burden from the client to the server and only requires a low bandwidth for transmission of small size feature data. In fact, this model has been integrated in the ETSI DSR standard [4].

Based on the above discussion, we decided to build our DSER architecture base on a client-server model. The proposed fundamental architecture of DSER is illustrated by Figure 1. This architecture consists of two parts. In the client part, which is shown in Figure 1(a), acoustic features are calculated from the input signal in the feature extraction block. Then, features are compressed and further processed for channel transmission. On the server side (see Figure 1(b)), bit-stream decoding and decompression are applied before further processing in the emotion recognition back-end. As a first perspective, this paper mainly puts emphasis on the feature extraction, compression, decompression and speech emotion recognition components. A detailed description of the first three components is given in this section, while the discussion on the last component is deferred to Section 4.

2.1. Feature Extraction

The audio feature set used is the INTERSPEECH 2009 Emotion Challenge feature set with 384 features [5] brute forced by functional application to low-level descriptors (LLD), extracted by our openSMILE [6] toolkit. In detail, the 16 LLD chosen are: zero-crossing-rate (ZCR) from the time signal, root mean square (RMS) frame energy, pitch frequency (normalized to 500 Hz), harmonics-to-noise ratio (HNR) by the autocorrelation function, and mel-frequency cepstral coefficients (MFCC) 1-12 in full accordance to HTK-based computation. From each of these, the delta coefficients are computed in addition. Next, twelve functionals including mean, standard deviation, kurtosis, skewness, minimum and maximum value, relative position, and range as well as two linear regression coefficients with their mean square error (MSE) are applied on a chunk basis as depicted in Table 1. Thus, the total feature vector per chunk contains $16 \times 2 \times 12 = 384$ attributes.

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

Table 1. Feature set: low-level descriptors (LLD) and functionals.

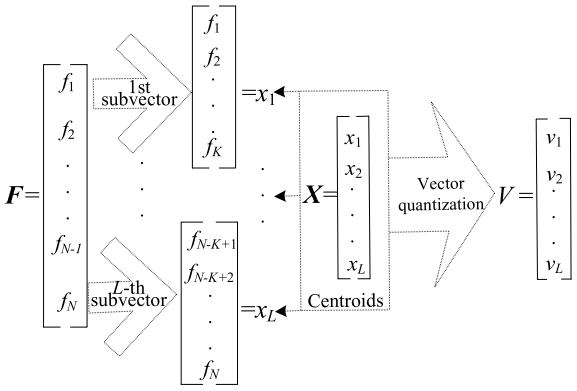

Fig. 2. Diagram of the Split Vector Quantization (SVQ) algorithm.

2.2. Feature Compression

To reduce the number of bits needed to represent each front-end feature vector, the SVQ algorithm is employed in this paper. The principle is to use a subspace quantization scheme, as shown in Figure 2. Specifically, the feature vector F is first split into L subvectors, $X = [x_1, \cdots, x_L]$, and then the subvectors are encoded by using separate codebooks, $V = [v_1, \cdots, v_L]$. The resulting set of index values is then used to represent the corresponding speech chunk. The closest quantization centroid is found using a weighted Euclidean distance to determine the index:

$$d_i^j = x_i - v_i^j, \quad i = 1, \ldots, L; j = 1, \ldots, N_i, \tag{1}$$

$$idx_i = \underset{0 \le j \le (N_i - 1)}{\arg\min} (d_i^j) W_i (d_i^j), \tag{2}$$

where v_i^j denotes the jth codevector in the codebook v_i, d_i^j denotes the Euclidean distance between subvector x_i and codevector v_i^j, N_i is the size of the codebook, W_i is the (possibly identity) weight matrix to be applied for the codebook v_i, and the idx_i denotes the codevector index chosen to represent the vector x_i. The indices are then retained for transmission to the back-end.

2.3. Feature Decompression

Using the indices received from channel, estimates of the front-end features are extracted with a codebook lookup on the server side:

$$\hat{x_i} = v_i^{idx_i}, \tag{3}$$

where $\hat{x_i}$ denotes the estimate of x_i.

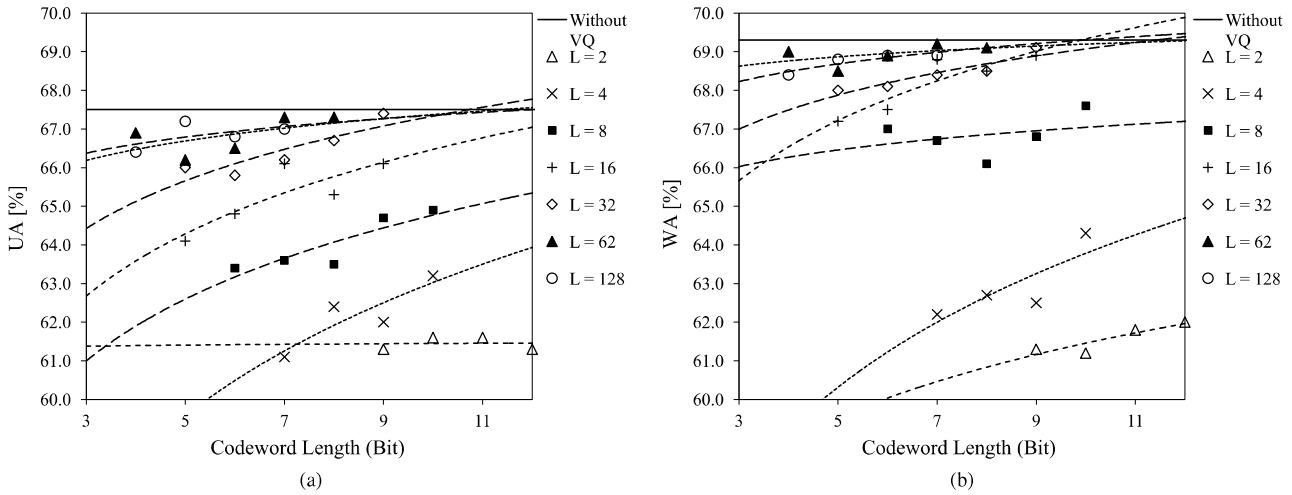

Fig. 3. Un-/weighted accuracies (UA (a) / WA (b)) for distributed speech emotion recognition with different numbers of feature subvectors and codeword lengths. L denotes the number of subvectors

Table 2. Number of instances for two classes: **NEG**ative and **IDL**e.

#	**NEG**	**IDL**	$\sum$
train	3 358	6 601	9 959
test	2 465	5 792	8 257
$\sum$	5 823	12 393	**18 216**

3. DATABASE

The FAU Aibo Emotion Corpus [7] used for the INTERSPEECH 2009 Emotion Challenge is chosen for our experiments. It is a corpus that containts recordings of children interacting with Sony's pet robot Aibo. The corpus consists of spontaneous, German speech that is emotionally coloured. The children were led to believe that the Aibo was responding to their commands, while the robot was actually controlled by a human operator. The wizard caused the Aibo to perform a fixed, predetermined sequence of actions; sometimes the Aibo behaved disobediently, thereby provoking emotional reactions. The data was collected at two different schools ('MONT' and 'OHM') from 51 children (age 10 – 13, 21 male, 30 female; about 9.2 hours of speech without pauses). Speech was transmitted with a high quality wireless head set and recorded with a DAT-recorder (16 bit, 48 kHz down-sampled to 16 kHz).

The recordings were segmented into chunks which are manually defined based on syntactic-prosodic criteria. The whole corpus consisting of 18,216 chunks and labelled into two classes is used in this paper. The cover classes **NEG**ative (subsuming *angry*, *touchy*, *reprimanding*, and *emphatic*) and **IDL**e (consisting of all non-negative states) are to be discriminated. The number of instances of the two classes are given in Table 2. Speaker independence is guaranteed by using the data of one school (OHM, 13 male, 13 female) for training and the data of the other school (MONT, 8 male, 17 female) for testing.

4. EXPERIMENTS AND RESULTS

To evaluate the effect of using vector quantized features for speech emotion recognition, we employ unweighted accuracy (UA) which was also the official 2009 Emotion Challenge performance measure [5] and reflects the imbalance among the classes. It is equivalent to the average recall of the IDL and NEG classes. Furthermore, we consider the conventional weighted accuracy (WA) (weighted with the prior class probabilities).

As classifier, we employ Support Vector Machines (SVMs) which are presently one of the most used classifier in emotion recognition. Thus, for representative results in our experiments, we chose SVM with linear kernel, complexity 0.05, and pairwise multi-class discrimination based on Sequential Minimal Optimization. Implementations in the Weka toolkit [8] were used for further reproducibility and by that keeping in-line with the challenge baseline.

Compared to most distributed speech recognition systems which have to transfer a set of only 39 features (13 Mel-Frequency Cepstral Coefficients including deltas and double deltas), the feature space in distributed emotion recognition is rather large (e. g., 384 features are considered in this study, 6 554 features are used in [9]). Thus, we experiment with a rather large number of encoded subvectors (2 to 128).

In order to investigate the influence of different numbers of subvectors as well as different codebook sizes for DSER, we consider the transmission of L = 128, 64, 32, 16, 8, 4, and 2 feature subvectors, corresponding to K = 3, 6, 12, 24, 48, 96, and 192 features per group. Each group is quantized using the same codebook size.

The SVQ codebook is constructed using the features from the training set, ignoring the class labels. We applied the K-means algorithm, which can be seen as a popular unsupervised clustering method. As distance measure we use the Euclidean distance as stated earlier. Various codebook sizes from 16 to 4 096 were used to quantize the different subvector sets.

Figures 3 (a) and (b) show the unweighted accuracies (UA) and weighted accuracies (WA) obtained for the two-class problem (discriminating the affective states NEG and IDL) when evaluating various SVQ strategies with different numbers of subvectors and dif-

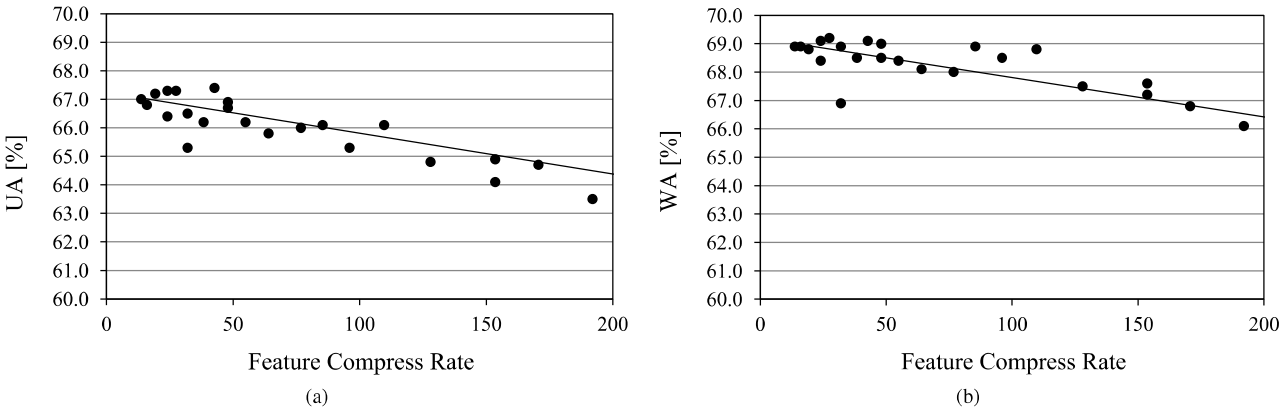

Fig. 4. Relationship between un-/weighted accuracies (UA (a) / WA (b)) and feature compression rate for distributed speech emotion recognition with several sets of permutations of codeword lengths and numbers of subvectors.

ferent codeword lengths. The seven trend lines (shown as different line types) correspond to the considered numbers of subvectors (2 to 128). For a given number of subvectors (e. g., L = 16, shown as 'plus' signs), we obtain better accuracies as the codeword length increases. Similarly, for a given codeword length (e. g., 7 bit), a larger number of subvectors leads to higher accuracies. Of course both, a large number of subvectors and a large codebook size increase the bandwidth needed for transmission. In our experiments, the best trade-off is obtained by using 32 subvetors and 9 bit codeword length (codebook size 512). For this setting we get an unweighted accuracy of 67.4 % and a weighted accuracy of 69.1 % (shown as diamonds), which is almost equal to the baseline performance (67.5 % UA; 69.3 %, WA) without SVQ processing (shown as solid line), but reaches a factor 30 data compression for feature transmission.

Figure 4 depicts the general relationship between UA (a) / WA (b) and feature compression rate employing K-means clustering and the proposed SVQ strategy. As expected, it can be seen that the accuracies decreases as the feature compression rate increases. For a feature compression rate between 25 and 50 we still obtain an acceptable recognition performance even though the transmission bandwidth is dramatically reduced. Assuming a transmission rate of one chunk-level feature set per second, only 288 bps bandwidth are needed when using a codebook size of 512 and 32 subvectors, compared to 12.35 kbps when transmitting the uncompressed feature space. Thus, by employing SVQ we are able to save a significant amount of bandwidth without observing a statistically significant reduction of recognition accuracy.

5. CONCLUSION

In this paper, we proposed and investigated a distributed speech emotion recognition (DSER) approach. Distributed affective computing is of great importance for internet-based large-scale real-life emotion recognition applications as well as for mobile and wearable applications of emotion recognition. We performed extensive evaluations on the FAU Aibo Emotion Corpus, strictly following the 2009 Emotion Challenge protocol. We investigated split vector quantification (SVQ) combined with unsupervised K-means clustering to reduce the required bandwidth in our distributed emotion recognition setting. Results show that the accuracy of DSER employing vector quantization (67.4 %, (UA); 69.1 %, (WA)) is nearly same as the baseline performance (67.5 %, UA; 69.3%, WA) obtained without employing the SVQ strategy. At the same time, we obtain a factor 30 feature compression (288 bps in our scenario). In general, a feature compression rate between 25 and 50 seems to be a good choice, as a higher compression degrades the performance of automatic emotion recognition.

Future experiments will focus on using different codeword lengths for different subvector groups, depending on their importance in discriminating emotional states. Further we plan to implement new SVQ methods to achieve higher compression rates without decreasing accuracy. Finally, DSER under noisy and reverberated environments [10] will be taken into account.

6. REFERENCES

[1] W. Zhang, L. He, Y. L. Chow, R. Yang, and Y. Su, "The study on distributed speech recognition system," in *Proc. of ICASSP*, Istanbul, Turkey, 2000, pp. 1431–1434.

[2] S. Tsakalidis, V. Digalakis, and L. Neumeyer, "Efficient speech recognition using subvector quantization and discrete-mixture hmms," in *Proc. of ICASSP*, Phoenix, AZ, USA, 1999, pp. 569–572.

[3] A. K. Jain, P. J. Flynn, and A. A. Ross, *Handbook of Biometrics*, Springer, 2008.

[4] ETSI ES 202 050 V1.1.5, "Speech processing, transmission and quality aspects (STQ), distributed speech recognition, advanced front-end feature extraction algorithm, compression algorithms," 2007.

[5] B. Schuller, S. Steidl, and A. Batliner, "The INTERSPEECH 2009 Emotion Challenge," in *Proc. of Interspeech*, Brighton, UK, 2009, pp. 312–315.

[6] F. Eyben, M. Wöllmer, and B. Schuller, "openSMILE – The Munich Versatile and Fast Open-Source Audio Feature Extractor," in *Proc. of ACM Multimedia (MM)*, Florence, Italy, 2010, pp. 1459–1462.

[7] S. Steidl, *Automatic Classification of Emotion-Related User States in Spontaneous Speech*, Logos, Berlin, Germany, 2009.

[8] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: an update," *ACM SIGKDD Explorations Newsletter*, vol. 11, pp. 10–18, 2009.

[9] Z. Zhang, F. Weninger, M. Wöllmer, and B. Schuller, "Unsupervised Learning in Cross-Corpus Acoustic Emotion Recognition," in *Proc. of ASRU*, Big Island, Hawaii, USA, 2011.

[10] M. Wöllmer, F. Weninger, S. Steidl, A. Batliner, and B. Schuller, "Speech-based non-prototypical affect recognition for child-robot interaction in reverberated environments," in *Proc. of Interspeech*, Florence, Italy, 2011, pp. 3113–3116.