Speech Emotion Recognition Using Sequential Capsule Networks

Xixin Wu, *Member, IEEE*, Yuewen Cao, *Member, IEEE*, Hui Lu, *Member, IEEE*, Songxiang Liu, *Member, IEEE*, Disong Wang, *Graduate Student Member, IEEE*, Zhiyong Wu, *Member, IEEE*, Xunying Liu, *Member, IEEE*, and Helen Meng, *Fellow, IEEE*

***Abstract*—Speech emotion recognition (SER) is an indispensable part of fluid human-machine interaction and attracts lots of research attentions. Recent work on SER has successfully applied convolutional neural networks (CNNs) to learn feature representations from speech spectrograms. However, the fundamental problem of CNNs is that the spatial information in spectrograms is lost, which includes positional and relationship information of low-level features, such as pitch and formant frequencies. We propose a novel architecture of sequential capsule networks (CapNets) by leveraging the advantange of CapNets that spatial information can be preserved in capsules and passed to upper capsule layers via dynamic routing. Also, the dynamic routing algorithm provides an effective alternative to pooling or storing recurrent hidden states for obtaining utterance-level features from the sequential capsule outputs. To further improve the model's ability to capture contextual information, we introduce a recurrent connection to the sequential structure. The experimental comparison of the proposed systems and previously published systems using CNNs and recurrent neural networks (RNNs) based on the IEMOCAP corpus demonstrates the effectiveness of the proposed sequential CapNets.**

***Index Terms*—Speech emotion recognition, capsule network, spatial information, sequential, recurrent.**

I. Introduction

EMOTION perception is an important step towards intelligent human-machine speech-based interactions, especially for informing the processes of machine inference and response generation during the interactions. The user's input speech contains rich emotive information, which needs to be captured in order to enable the machine to exhibit "Emotional Intelligence".

Manuscript received February 8, 2021; revised June 26, 2021; accepted September 11, 2021. Date of publication October 15, 2021; date of current version November 4, 2021. This work was supported in part by the National Natural Science Foundation of China-Research Grants Council of Hong Kong (NSFC-RGC) joint fund under Grants 61531166002 and N_CUHK404/15. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Lei Xie. *(Corresponding author: Zhiyong Wu.)*

Xixin Wu is with the Stanley Ho Big Data Decision Analytics Research Centre, The Chinese University of Hong Kong, Hong Kong, China (e-mail: wuxx@se.cuhk.edu.hk).

Yuewen Cao, Hui Lu, Songxiang Liu, Disong Wang, Xunying Liu, and Helen Meng are with the Department of Systems Engineering and Engineering Mangement, The Chinese University of Hong Kong, Hong Kong, China (e-mail: ywcao@se.cuhk.edu.hk; lu-h17@mails.tsinghua.edu.cn; sxliu@se.cuhk.edu.hk; dswang@se.cuhk.edu.hk; xyliu@se.cuhk.edu.hk; hmmeng@se.cuhk.edu.hk).

Zhiyong Wu is with the Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen 518055, China (e-mail: zywu@se.cuhk.edu.hk).

Digital Object Identifier 10.1109/TASLP.2021.3120586

Speech emotion recognition (SER) aims to identify the affective status of a speech utterance from the features such as MFCC features, energy-related features and pitch-related features [1]–[6]. The starting question is which features are effective for emotion recognition. This is still an open question, although there are some features that are commonly considered to be highly related to emotions, such as F0 and energy [7]–[9]. The second question is how to predict the affective status based on the input features. One possible way is to calculate the statistics of some hand-crafted features and apply classifiers such as support vector machine (SVM) or Gaussian mixture model (GMM) to the statistics [3]. Since global statistics tend to ignore detailed temporal variations in the features, researchers have also investigated models to leverage temporal information. Schuller *et al.* [5] applied hidden Markov model (HMM) to SER. The HMM can handle temporal complexity with several hidden states to respectively model various parts of the sequence of low-level features (e.g. frame-level F0 and energy, etc.). Experimental results show that the performance increases as more states are used, which reflects the potential of leveraging detailed temporal variations.

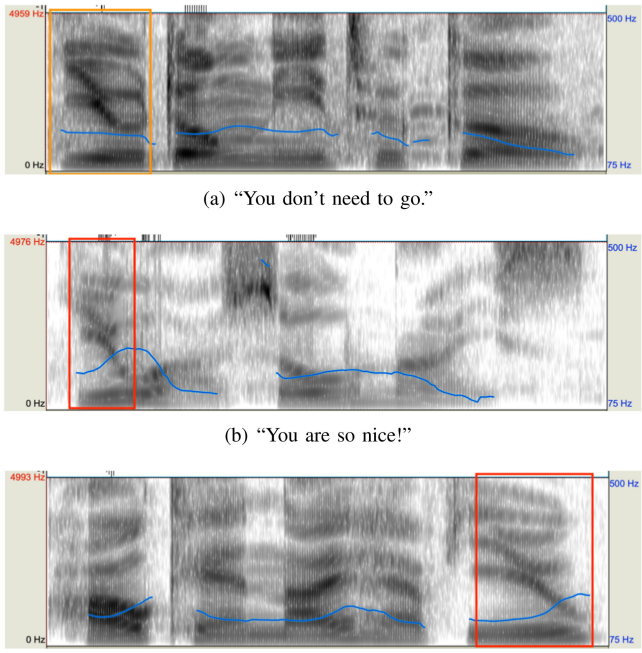
Recently, deep learning techniques have been applied to SER and demonstrated significant performance improvements [6], [10]–[13]. Much research effort has been devoted to the two questions above, relating to feature selection and model prediction under the deep learning framework. The application of neural networks enables the automatic selection of effective features and automatic learning of hidden representations. The work by Han, Yu and Tashev [10] is representative in utilizing deep neural network (DNN)-based models to learn neural features from input features of MFCCs, pitch period and harmonics-to-noise ratio. In addition to the above acoustic features, Li *et al.* [14] also applied neural networks to lexical features (e.g. word embedding – a neural representation of words) to infer users' emotion states in conversational dialogues. The weights of the input features are determined automatically via network training. However, it is difficult to determine which features should be included in the input feature set. The increase of feature set size also incurs training complexity. Satt *et al.* [15] present a novel convolutional neural network (CNN)-based framework which directly uses speech spectrograms as input. The spectrogram is a 3-dimensional (time, frequency and magnitude) representation of a signal, depicting how the spectrum of frequencies varies with time. Compared to the hand-crafted feature sets (e.g.

consisting of MFCCs, pitch and energy features), the spectrogram is an raw representation without much specific feature expression. The powerful neural networks enable the automatic extraction of features from the raw representation, and hence reduce the burden of feature engineering. Various subsequent efforts apply convolutional layers and pooling layers directly on spectrograms and achieve successful results, demonstrating the advantages of using spectrograms as input features [16]–[22].

Given the input feature of spectrograms, various network architectures are proposed to capture the temporal information (variation across time axis) and spatial information (variation across time and frequency axes). The CNN structure is utilized to decide which information are essential for emotion classification and learn neural hidden representations that encode the information from the spectrograms. However, CNN tends to ignore the spatial information, e.g. relative positions of pitch and formant features, but these actually provide important clues for SER [23]. Sabour, Frosst and Hinton [23] proposed the capsule network structure with dynamic routing algorithm to consider the spatial information and achieve successful results in digit recognition from images. In this paper, we investigate the application of capsule networks to SER, based on our previous work [24]. The recognition performance is expected to be improved via accounting for spatial information in the spectrogram. The approach is as follows: First, the capsule-based SER systems use spectrograms as inputs, which preserve much of the emotive information in the inputs. We then apply a sequential structure composed of window-level capsules to the spectrograms to avoid local-to-global information loss. The sequential structure is able to handle the input sequence of feature frames with variable lengths. Recurrent connections are further introduced to the sequential structure to capture the temporal information in the input sequence. An utterance-level dynamic routing upon the sequential capsule outputs is utilized to obtain the utterance representation from each of the window-level outputs.

In summary, this paper presents a capsule network structure uniquely suited for speech emotion recognition (SER), that fully leverages the spatiotemporal information encoded in input spectrograms. This aims to mitigate the problems of spatial (i.e. frequency) information loss in the conventional CNN approach, where the loss of local information will propagate to a higher level, i.e. global information loss. Our contributions in devising the capsule network approach present the advantages of: (i) being able to handle input sequences of variable lengths; (ii) applying dynamic routing to capture the spatial (i.e. frequency) information in the spectrograms; (iii) incorporation of recurrent connections to capture the temporal information in the spectrograms; (iv) introducing the curriculum learning scheme to expedite capsule structure training; and (iv) demonstrating that the capsule network approach can apply the learned sequential structure to achieve superior performance in SER.

The rest of this paper is organized as follows: Section II reviews the previous work on SER, summarizing the two challenging problems of (i) spatial information loss, and (ii) local-to-global information lost. The baseline systems based on CNNs and recurrent neural networks (RNNs) are introduced in Section III. Section IV describes the capsule structures and the dynamic routing algorithm. The details of systems implementation and the evaluation results are given in Sections V and VI, respectively. Conclusions are drawn in Section VII.



(a) "You don't need to go."

(b) "You are so nice!"

(c) "How can I help you?"

Fig. 1. Utterance (a) is a neutral statement without salient intonation rise. In contrast, utterance (b) and (c) contain intonation rise at the beginning and the end, respectively. Utterance (b) is uttered in an emotive way and utterance (c) is in a question style. The emotive information provided by the salient intonation rises at different positions is expected to be captured for emotion prediction.

II. Related Work

Convolutional layers can effectively learn hidden representations that contain feature-level and time information for the subsequent decision models (e.g. HMM [11], ELM [10]) or classification layers (e.g. softmax layer [15], attention layer [19]). Convolutional layers are able to detect patterns described by the kernels in an efficient way that different parts of the input share the same kernels [21], [25]. Satt *et al.* [15] applied convolutional layers to learn spectrogram patterns that represent emotive information, e.g. the silence or low-energy zones and the harmonic structures. However, the detailed instantiation information is ignored, because the shared kernels are applied to every part of the input feature map, and the pooling operation just selects those frames with salient values (e.g. maximum and minimum) and ignores the other frames in the pooling window. Chen *et al.* [26] proposed the dynamic multi-pooling CNN to maintain three pooling operations on three disjoint parts of the input feature map respectively, in order to capture simultaneous feature events in the feature map. However, for each of pooling operations, the problem of information loss still exists. As shown in the three examples in Fig. 1, the salient feature pattern of intonation rise exists in both the emotive utterance in Fig. 1(b) and the neutral utterance (which is a question) in Fig. 1(c), but not in the neutral utterance in Fig. 1(a). The different spatial

information, i.e. the relative position, needs to be considered, such that accurate emotion prediction can be generated based on the salient patterns. The recent proposed capsule structure can preserve the positional information in the form of vector direction and pass the information to the upper layers (as will be shown in Sec. IV-A2) [23]. Hence, the capsule structure is expected to improve the performance with such positional information.

The outputs of the convolutional layers are a sequence of frames of variable lengths, which needs to be compressed to a fixed-size global utterance-level representation. How to obtain a global utterance-level representation from the local frame-level features is another challenging problem, because utterance-level statistics may obfuscate emotional information [27]. Recurrent layers are integrated to capture the temporal information in the sequence [12], [25], [28]–[32]. One possible method is to use the last (and/or first)-timestep hidden states or outputs of the top recurrent layer as utterance-level representation [15], [20]. However, the last (or first) timesteps correspond to silence frames in most cases. Also, contextual information propagation vanishes across long distances. To address this problem, the attention mechanism is utilized to summarize the time segments that are relevant to emotion recognition from the feature sequence [33]–[35]. An attention weight α_t is assigned to each step t of the input sequence $\boldsymbol{y}$ of length T. These weights determine the contribution of each step to the final summarized representation $\boldsymbol{z}$ as

$$\alpha_t = \frac{\exp\left(\boldsymbol{w}^\top \boldsymbol{y}_t\right)}{\sum_{\tau=1}^{T} \exp\left(\boldsymbol{w}^\top \boldsymbol{y}_\tau\right)} \tag{1}$$

and

$$\boldsymbol{z} = \sum_{\tau=1}^{T} \alpha_\tau \boldsymbol{y}_\tau, \tag{2}$$

where $\boldsymbol{w}$ is trainable parameters. Since the summation is operated on the whole sequence, the long-distance contextual information can be directly accessed, rather than propagated through recurrent connections in recurrent neural networks (RNNs). The steps that contain less emotive information are expected to be assigned lower attention weights, hence the output decisions can be made with focuses more on the salient parts [12]. In addition to the attention across timesteps, Xie *et al.* [36] introduce the attention across feature dimensions, i.e. the attention $\boldsymbol{\alpha}$ are calculated to determine the weights for each feature dimension. However, the two attentions across time and across feature dimensions are calculated separately. A unified attention mechanism across both dimensions is desirable. Another similar mechanism applied to CNN for capturing salient parts is the gating mechanism [37], which generates gate values (typically via the sigmoid function, ranged from 0 to 1) based on the fixed-size input windows over the feature map. The gate values are multiplied to the outputs of the windows to obtain the gated outputs. Salient parts are expected to have close-to-one gate values such that the salient information can pass through the gates. However, the gate value calculation is limited to a fixed-size window, and so contrastive contextual information cannot be considered.

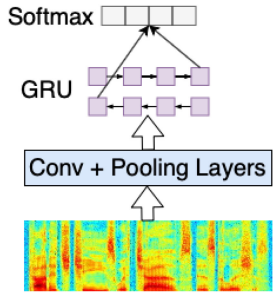


Fig. 2. Architecture of the baseline system composed of a CNN component, GRU layers and a final softmax layer.

The dynamic routing algorithm introduced in [23] generates the utterance-level representation by jointly considering the weights assigned to timesteps and feature dimensions. Also, the whole sequence can be accessed in the routing process, which ensures the sufficient capturing of long-distance context.

The capsule structure is proposed to improve CNN to capture detailed spatial information by Hinton *et al.* [23], [38]. The dynamic routing algorithm connecting the capsule layers is designed to pass the information to the upper layer. In this work, we try to address the two problems described above, i.e. spatial information loss and local-to-global information loss, by using capsule structures with the dynamic routing algorithm. We propose to apply a sequential structure of capsule networks to SER to enhance the capturing of spatial information across time and frequency axes in speech spectrograms. The capsule structures have been successfully applied to various tasks, e.g. image processing [23], natural language understanding [39], [40] and speech processing [41], [42]. Jalal *et al.* [43] also investigate the application of capsules in SER. Our approach is novel in the sequential structure, the recurrent connection and the utterance-level dynamic routing.

III. Baseline System Based on CNN and RNN

The overall structure of the baseline system consists of a CNN component and multiple recurrent layers [20], as shown in Fig. 2. From the input spectrogram, a neural representation is learned by the CNN component, where the convolutional layers are expected to recognize feature patterns from the spectrogram and the pooling layers are utlized to reduce the input feature size. Upon the CNN component, gated recurrent units (GRUs) are integrated to capture the temporal information. A final softmax layer outputs the probabilistic predictions.

A. CNN Component

The CNN component is composed of convolutional layers and pooling layers. For each convolutional layer, the input feature map of the three dimensions (width, height and channel number) is transformed to another 3-dimension feature map, as shown in Fig. 3(a). A matrix of the three dimensions of width, height and channel number, referred to as *kernel*, is applied to each portion of the input feature map, by performing a element-wise multiplication operation between the kernel and the feature map

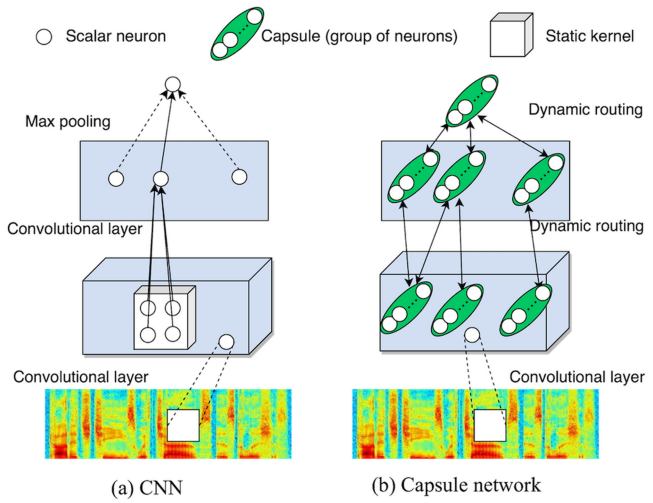


Fig. 3. Architecture comparison between CNN and capsule networks. There are mainly two differences: (i) replacing a neuron that outputs a scalar with a capsule (a group of neurons) that outputs a vector – the vector length is used to represent the probability that certain feature pattern exists, and the vector direction is used to represent instantiation parameters; (ii) replacing max-pooling layer with dynamic-routing algorithm – the routing couples capsules in various positions of the lower layer to the upper-layer capsules, enabling upper-layer capsules to consider spatial relationships.

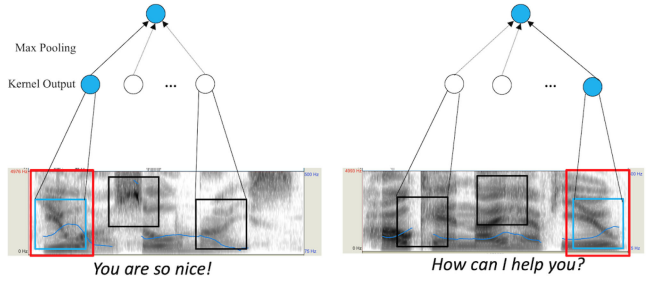


Fig. 4. CNN perception of intonation rises at different positions. The shared kernel is applied to various parts of the input. The activated outputs are highlighted as blue. Although the activated outputs come from different parts of the input spectrogram, the max-pooling operation produces the same result (both highlighted as blue), which hinders the accurate classification of the emotional (left) and neutral (right) utterances.

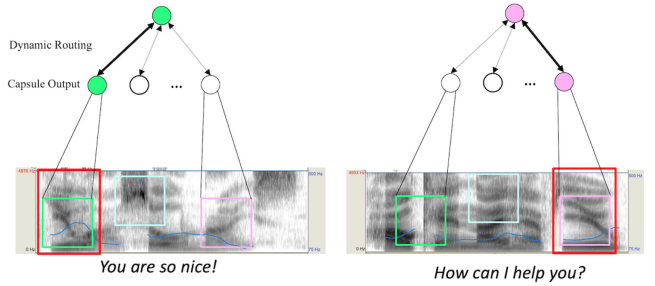


Fig. 5. Capsule perception of intonation rises at different positions. The intonation rises at different positions produce different capsule outputs, which are highlighted as green and purple. The dynamic routing then passes the capsule outputs to the upper layer. The final distinguished outputs (highlighted as green and purple) support the accurate emotion classification.

portion. The kernel is shared across different portions. The multiplication outputs are fed to a non-linear activation (e.g. ReLU) to generate the scale-valued outputs. One kernel and the corresponding non-linear activation make up one neuron. In practical use, multiple kernels are applied to obtain multiple output values, referred to as multiple channels. The pooling layer is responsible for reducing the spatial size of the input feature map. A typical pooling operation is *max-pooling*, which takes the maximum over values of a certain window (e.g. size of 2×2) determined by the layer's hyperparameters.

B. Recurrent Layers

Upon the CNN component, bi-directional GRU layers are applied to capture temporal information. The outputs of the CNN component are fed to the forward and backward GRU layers respectively. The final state of forward GRU and the first state of backward GRU are concatenated and fed to the subsequent softmax layer for final predictions.

C. Classification Layer

This paper focuses on discrete emotion classes, e.g. the classes of *Happy*, *Angry*, *Sad* and *Neutral*. The softmax layer is leveraged for calculating the probabilities of the emotions expressed in the given speech. Generally only the emotion with the top probability is considered.

IV. Capsule-Based Systems

The whole architecture of the SER system based on capsule networks is composed of one CNN component, multiple capsule layers and one classification layer, as shown in Fig. 6. The CNN component structure is the same as the baseline system. Although the capsule structure is able to capture spatial information, the computational cost is relatively higher than the CNN structure. Also, convolutional layer kernels are shared across the input feature map. This allows the CNN structure to transfer knowledge about good weight values learned at one position in an image to other positions. With the pooling operations, the higher layers cover larger regions of the input. We therefore aim to leverage the convolutional layers and pooling layers in the CNN component to learn preliminary features for the upper capsule layers. The following sections describe the capsule layers we use.

A. Capsule Networks

The capsule network (CapNet) is proposed by Hinton *et al.* [23], [38], [44] to improve the CNN structure's sensitivity to instantiation parameters of the recognized patterns, e.g. spatial information. The idea is to maintain a group of neurons, instead of a unique neuron, to capture both the existence probability and the instantiation information. As shown in Fig. 3(b), convolutional layers are utilized to create the first layer of capsules, called primary capsules. The neurons of different channels at the same position along the width- and height-axes of output feature map of the convolutional layers are grouped together to form a capsule. For the connection between capsule layers,

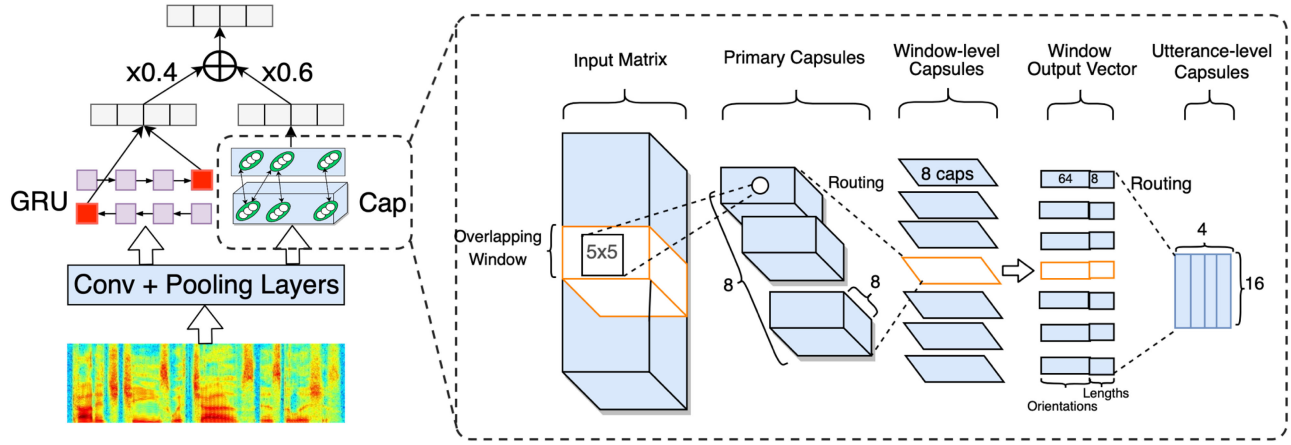


Fig. 6. Architecture of the CNN_GRU_Cap system. For the GRU branch, the last hidden states (highlighted as red) are used as the utterance-level representation for classification. For the capsule branch, the output feature map from the lower convolutional and pooling layers is first sliced into windows, and then a shared-weight capsule layer is applied to the windows. Finally the window outputs are aggregated and routed to the subsequent layer to obtain utterance-level capsules for classification.

the dynamic routing algorithm is applied to learn the hierarchical relationships between the learned features in neighboring layers.

1) Dynamic Routing: Assume that the i-th capsule in layer l is $\boldsymbol{u}_i$, and the j-th capsule in layer l+1 is $\boldsymbol{v}_j$. The $\boldsymbol{u}_i$ is first projected to the space of $\boldsymbol{v}_j$ by

$$\hat{\boldsymbol{u}}_{j|i} = \boldsymbol{W}_{ij}\boldsymbol{u}_i + \boldsymbol{b}_{ij}, \tag{3}$$

where $\boldsymbol{W}_{ij}$ and $\boldsymbol{b}_{ij}$ are weight matrix and bias vector, and they are both position-aware and trainable. To obtain the capsule $\boldsymbol{v}_j$ in the upper layer, the procedure described by (4)–7 is iterated for a predefined number of times n, with the initial value of $d_{ij} = 0$:

$$c_{ij} = \frac{\exp(d_{ij})}{\sum_k \exp(d_{ik})}, \tag{4}$$

$$\boldsymbol{s}_j = \sum_i c_{ij}\hat{\boldsymbol{u}}_{j|i}, \tag{5}$$

$$\boldsymbol{v}_j = \frac{||\boldsymbol{s}_j||^2}{1 + ||\boldsymbol{s}_j||^2}\frac{\boldsymbol{s}_j}{||\boldsymbol{s}_j||} \tag{6}$$

and

$$d_{ij} \leftarrow d_{ij} + \hat{\boldsymbol{u}}_{j|i} \cdot \boldsymbol{v}_j, \tag{7}$$

where $\cdot$ represents dot production. The c_{ij} is the coupling coefficient measuring the agreement between $\boldsymbol{v}_j$ in the upper layer and $\hat{\boldsymbol{u}}_{j|i}$ projected from $\boldsymbol{u}_i$. Hence, this algorithm is also called routing-by-agreement.

2) Comparison of CNNs and CapNets: The limitation of CNNs is the insensitivity to detailed spatial information. The neurons in the convolutional layers output a scalar, which only provides the probability that the feature pattern (e.g. formants) matches the kernel. However, the more detailed instantiation parameters (e.g. position) are ignored. As shown in Fig. 4, the shared kernel is applied to various parts of the spectrogram. When the feature pattern (e.g. intonation rise) matches the kernel, the output is activated (highlighted as blue), regardless of which position (the beginning or ending part) the activation occurs. Also, the max-pooling layer discards all but the most activated neuron, which hinders the spatial relationship information to be passed to the upper layers. As shown in Fig. 4, the activated outputs at different positions are selected by the max-pooling operation. The final CNN outputs of the neutral and emotional utterances are the same. Hence, the subsequent layers lack the essential information for distinguishing the two utterances.

Compared to the CNN architecture, CapNets have two improvements, which enable CapNets to consider the detailed instantiation parameters of the recognized feature patterns, as shown in Fig. 3:

- Neuron vs. capsule: The neuron that outputs a scalar in the convolutional layer is replaced with a capsule, i.e. a group of neurons, that outputs a vector, which contains the instantiation information. The information includes the pose and position information of the recognized pattern.
- Max-pooling vs. dynamic-routing: The max-pooling layer is replaced with the dynamic-routing algorithm, which couples capsules in various positions of lower layer to upper-layer capsules, enabling upper-layer capsules to consider spatial relationship.

As shown in Fig. 5, the positional information distinguishes the capsule outputs. The intonation rises at different positions produce different capsule outputs, which are highlighted as green and purple. The position-aware capsule outputs are then passed to the upper layer via dynamic routing. The final distinguished outputs support the accurate emotion classification.

B. Sequential Capsules

In the task of SER, the input data is a sequence of feature frames with variable lengths (up to 1000 frames). The capsules corresponding to the ending frames will be trained with less data. The parameter size of the capsule model is huge when the input matrix is large, because the weight matrix for projecting the input capsule values into hidden representations is position-aware, as

TABLE I
CONFIGURATION OF THE CNN COMPONENT APPLIED TO THE INPUT SPECTROGRAM. C, K, AND W STAND FOR CHANNEL NUMBER, KERNEL SIZE AND POOLING WINDOW

Layer	Structure
Conv2d_1	C=8, K=2×8
Conv2d_2	C=8, K=8×2
Concat	Conv2d_1 + Conv2d_2
Max-pooling	W=2×1
Conv2d_3	C=16, K=5×5
Max-pooling	W=2×2
Conv2d_4	C=16, K=5×5
Max-pooling	W=2×2
Max-pooling	W=4×1

TABLE II
NUMBER OF PARAMETERS IN VARIOUS SYSTEMS

Systems	Parameter Number
CNN_GRU	833,012
CNN_Cap	131,110,832
LSTM_Cap	17,607,492
CNN_SeqCap	703,540
CNN_RecCap	708,148
CNN_GRU-SeqCap	1,523,448
CNN_GRU-RecCap	1,528,056

shown in (3). Feeding the whole sequence with large length to capsule layers is impractical, since the size of the trainable weights is huge, proportional to the frame number, and the weights are difficult to train (discussed in Sec. V-D).

In order to optimize the model upon the whole sequence simultaneously, we propose the structure of sequential capsules (SeqCaps), as shown in Fig. 6. The input frame sequence (e.g. spectrogram) is first sliced into overlapping windows, and the shared capsule layers are applied to each of these windows. In each window, several separated convolutional layers shared across windows are applied to the input to obtain primary capsules $\boldsymbol{u}$. The primary capsules are routed to generate window-level capsules $\boldsymbol{v}$. The utterance-level capsules are then obtained with utterance-level routing based on the output vectors $\boldsymbol{o}$ of these windows, which are defined as:

$$\boldsymbol{o} = \left[\boldsymbol{v}_1^\top, \ldots, \boldsymbol{v}_m^\top, ||\boldsymbol{v}_1||, \ldots, ||\boldsymbol{v}_m||\right]. \tag{8}$$

The window output vector consists of the orientations and lengths of all the m capsules in one window, since both the orientations and lengths contain useful information for the utterance-level emotion classification. Though the length information $||\boldsymbol{v}||$ is redundant given the vectors $\boldsymbol{v}$, we intend to provide this information explicitly to save the learning effort of the network. Since the convolutional layers and the capsule layers are shared across windows, the parameter number is reduced significantly (as will be shown in Table II).

C. Recurrent Capsules

The temporal information of speech contains important cues for emotion recognition. In order to enable the SeqCap model to capture temporal information, we propose the structure of recurrent capsule (RecCap), by introducing recurrent connections to the routing algorithm.

Denote the j-th capsule in layer $l+1$ in window $t-1$ as $\boldsymbol{v}_{t-1,j}$ and the i-th capsule in layer l in window t as $\boldsymbol{u}_{t,i}$. The projected vector $\hat{\boldsymbol{u}}_{t,j|i}$ in window t is produced as Eq. 9:

$$\hat{\boldsymbol{u}}_{t,j|i} = \boldsymbol{W}_{ij}^u \boldsymbol{u}_{t,i} + \boldsymbol{W}_{ij}^o \boldsymbol{o}_{t-1} + \boldsymbol{b}_{ij}, \tag{9}$$

where the window-level output vector $\boldsymbol{o}_{t-1}$ contains both the length and orientation information of all the m capsules in layer $l+1$ in window $t-1$:

$$\boldsymbol{o}_{t-1} = \left[\boldsymbol{v}_{t-1,1}^\top, \ldots, \boldsymbol{v}_{t-1,m}^\top, ||\boldsymbol{v}_{t-1,1}||, \ldots, ||\boldsymbol{v}_{t-1,m}||\right]. \tag{10}$$

Via this connection, the spatial information in the previous window can assist in determining the coupling coefficients and activating the activity which is salient in terms of window steps.

D. Curriculum Learning

The training of capsule structure is challenging, due to the introduction of utterance-level routing. The routing process groups all the outputs of the windows in the whole utterance. At the beginning of training, the projections ((3) and 9) are under-trained and not yet convergent. The routing of long utterances may propagate sub-optimal capsule grouping results to the subsequent layers. Hence, we introduce a simple, yet effective curriculum learning (CL)-based training scheme to the capsule training [45]. The CL research argues for presenting examples for learning in a meaningful order, e.g. from simple to complex concepts, rather than the random order. The well-organized order, similar to a curriculum, is believed to enhance the neural structure learning [46]. A scoring function that indicates the difficulty of learning each sample is defined for organizing the samples.

In our task, recognizing emotions from shorter utterances is defined as an easier task, and from longer utterances as a more difficult task. The model is trained using shorter utterances first, and gradually exposed to longer utterances. Hence, at the beginning, the capsule sequence is short and the projection parameters can be updated quickly towards the correct direction, without propagation of sub-optimal routing results. In practice, for the first epoch of training, we sort the utterances in the training set in an order of increasing length, i.e. number of frames. After the first epoch, the utterances are shuffled for each of the following epochs (i.e. backed off to normal random-order training).

V. SYSTEM IMPLEMENTATION

To evaluate the performance of the capsule structures, we compare the various systems consisting of convolutional layers, GRU layers and attention layers with the capsule-based systems.

A. CNN Component

The common component for these systems is the CNN component applied to the input spectrograms to extract neural representations for the subsequent layers. The configuration of the CNN component is shown in Table I. We apply two separated convolutional layers with kernel of 2×8 and 8×2 to capture the relationship information across frequencies and timesteps. The

outputs of these two separated convolutional layers are concatenated together and passed through another two convolutional layers and three max-pooling layers.

B. Baseline System Configuration

We refer to the CNN system described in [15] as one of the baseline systems. The CNN system consists of five convolutional layers and represents the state-of-the-art performance using CNN structure.

We explore applying a GRU layer upon the CNN component mentioned above via the system of CNN_GRU. The GRU layer is bidirectional with 64 cells per direction. The final state of forward GRU and the first state of backward GRU are concatenated and fed to a dense layer with 64 units activated by ReLU and dropped out with rate of 0.5. The dense layer outputs are then fed to a linear dense layer with 4 units. A softmax function is applied to the final outputs to obtain the emotion probabilities. Cross entropy criterion is used as the training objective function.

As demonstrated in [19], an attention layer following the GRU layer can further improve the system performance. We also implement a baseline system CNN_GRU_Att that replaces the last dense layer of the CNN_GRU with an attention layer that is composed of class-agnostic bottom-up, and class-specific top-down attention maps [47].

We also compare our proposed model with the capsule-based structure LSTM_Cap proposed by Jalal *et al.* [43]. The LSTM_Cap consists of two bi-directional long short-term memory (BLSTM) layers and two capsule layers. Each of the BLSTM layers contains 256 cells (i.e. 128 per direction). The first capsule layer is a primary capsule layer composed of four 1D convolutional layers with 32 filters and kernel size of 5. The outputs of the convolutional layers at each position are concatenated to form a capsule. The second capsule layer contains four capsules with 32 dimensions. The two capsule layers are connected via dynamic routing.

C. Capsule-Based System Configuration

We develop two capsule-based systems CNN_SeqCap and CNN_RecCap, by stacking SeqCaps and RecCaps on the CNN component, respectively. The detailed structure of the SeqCap is shown in Fig. 6, where the input matrix is the CNN outputs. For each window sliced from the input matrix, 8 convolutional layers with kernel size of 5×5 and channel number of 8 are applied to the input matrix. Then for each position in the outputs of the 8 convoutional layers, the units along the channel direction are concatenated together to obtain capsules with size of 8 (i.e. the channel number). These capsules are then routed to the subsequent window-level capsule layer with 8 capsules of size 8 in each window, as described by Eq. 3–7. The window output vectors are then obtained as Eq. 8. An utterance-level routing is conducted upon the window output vectors to produce 4 utterance-level capsules of size 16. The utterance-level capsules are then fed to two dense layers and softmax function, with the same configuration as the last two dense layers and softmax function in CNN_GRU. The window used to slice the input matrix is set to size of 40 input steps with shift of 20 steps. The iteration number of the routing algorithm is set to 3.

The system of CNN_RecCap has the same architecture as CNN_SeqCap, except the recurrent connection in the routing from the primary capsules to the window-level capsules (Eq. 9). The RecCap structure is expected to capture the temporal information better than SeqCaps.

To further improve the system's long-term view, we add another branch of GRU layer upon the CNN component, parallel to the capsule branch, denoted as CNN_GRU-SeqCap. The outputs of the GRU layers and those of the capsule components are fed to separate sets of dense layers and softmax function. The softmax outputs of the two branches are merged together by heuristically determined weights (e.g. 0.4 and 0.6), as shown in Fig. 6. At the training stage, the total loss of CNN_GRU-SeqCap is the unweighted sum of losses of the two branches. At the testing stage, the output probabilities of the capsule branch and the GRU branch are combined with the weights of λ and $1-\lambda$ respectively. We set λ as 0.6 in our experiments. Similar architecture of CNN_GRU-RecCap replacing the SeqCaps structure with RecCaps is also evaluated.

D. Parameter Sizes of Systems

The parameter numbers of the systems are shown in Table II. The system CNN_Cap uses non-sequential capsule layers, i.e. the whole input sequence is fed to the capsule layers. The parameter number is proportional to the input sequence length (e.g. 200 frames). As can be seen, the parameter number of CNN_Cap is much greater than that of CNN_GRU, leading to difficulty in training. With the sequential structure, the parameter numbers of the systems CNN_SeqCap and CNN_RecCap decrease significantly (even smaller than that of CNN_GRU). The parameter size of LSTM_Cap is also bigger than those of CNN_SeqCap and CNN_RecCap, since no sequential structure is used in the LSTM_Cap structure.

E. Network Training

We found that good weight initialization is quite important to the convergence of CapNets [41]. In our experiments, we use the Xavier initializer for both the CNN component and the capsule layer initialization. The batch size is set to 16 and the Adam algorithm is configured with parameters of $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 1e$-8 [48]. The learning rate is set to 0.001 in the first 3 epochs, and decayed dynamically determined by the average of training losses of the lastest 100 training steps. The learning rate is reduced to 0.0005, 0.0002 and 0.0001 gradually, when the average training loss is reduced by a factor of 10. The models are all trained for 20 epochs and then optimized on the validation set with respect to the weighted accuracy.

VI. System Evaluation

We conduct experiments to evaluate the effectiveness of the sequential capsule and the recurrent capsule structures based on a public corpus, using the common metrics of weighted accuracy and unweighted accuracy.

TABLE III
NUMBER OF UTTERANCES TO THE FOUR EMOTIONS IN THE EXPERIMENTAL DATASET

	Neutral	*Angry*	*Happy*	*Sad*
Utt #	1099	289	284	608

A. Emotion Recognition Corpus

We evaluate the capsule structures on the common evaluation dataset interactive emotional dyadic motion capture (IEMOCAP) database [49], which consists of five sessions, with two speakers in each session. We adopt five-fold cross validation as [15]: 8 speakers from four sessions in the corpus are used as training data. One speaker from the remaining session is used as validation data, and the other one as test data. To further validate the effectiveness of the systems, we run the experiments five times with different random seeds for *t*-test analysis. We evaluate our systems on four emotions in the corpus, i.e. *Neutral*, *Angry*, *Happy* and *Sad*, following previous work [15], [20]. The improvised data is used, and the utterance numbers of the emotions are shown in Table III. It is already reported in previous works [20], [50] that the category of *Happy* is difficult to be recognized because of limited training data and its special emotion characteristics, i.e. the *Happy* emotion relies on context contrastive information more than the other categories.

Spectrograms are extracted from the speech signal in IEMOCAP and split into 2-second segments. The segments split from one sentence share the same emotion label. The training is conducted based on the 2-second segments. It is only during the testing stage that the whole original spectrogram is used for evaluation. The spectrograms are extracted with 40-ms Hanning window, 10-ms shift and DFT of length 1600 (for 10 Hz grid resolution). The frequency range of 0-5.12KHz is used, ignoring the rest. The spectrograms are finally represented by a $N \times M$ matrix, where $N \leq 200$ corresponds to the segment length and $M = 512$ according to the selected frequency grid resolution. We normalize the whole dataset to have zero mean and unit variance.

B. Evaluation Metrics

We use two common evaluation metrics to evaluate the systems' performance:

- Weighted Accuracy (WA) – the accuracy of all samples in the test data.
- Unweighted Accuracy (UA) – the average of class accuracies in the test set.

$$\texttt{WA} = \frac{\sum_{i=1}^{K} P_i}{\sum_{i=1}^{K} U_i} \tag{11}$$

$$\texttt{UA} = \frac{\sum_{i=1}^{K} P_i / U_i}{K} \tag{12}$$

where P_i is the number of utterances with correct prediction of emotion i, U_i is the number of utterances with actual emotion i, and K is the number of emotions tested.

TABLE IV
PERFORMANCE IMPROVEMENT WITH CURRICULUM LEARNING

Systems	CL	WA(%)	UA(%)
CNN_GRU	×	67.31	51.53
	✓	67.02	51.84
CNN_GRU_Att	×	68.26	53.07
	✓	68.20	54.89
LSTM_Cap	×	66.48	52.98
	✓	67.31	53.70
CNN_SeqCap	×	68.40	52.43
	✓	69.86	56.71
CNN_RecCap	×	70.06	56.24
	✓	70.62	58.17

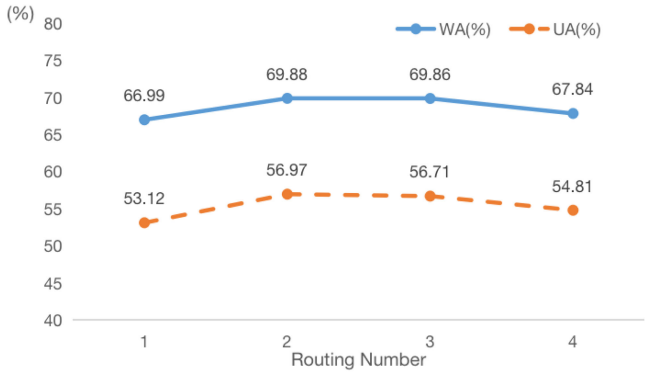


Fig. 7. Performances of CNN_SeqCap with various routing number.

C. Effectiveness of Curriculum Learning

We examine the CL training scheme on the models. The comparison of performances with and without CL training is shown in Table IV. It can be found that the CL training scheme effectively improves the performance of all the three capsule-based structures, LSTM_Cap, CNN_SeqCap and CNN_RecCap, at both WA and UA. This conforms with the nature of the capsule structure that the dynamic routing applied to long sequences at the beginning of training is not optimal and the resulting network connections can be very different from the final convergent connections. For the two conventional structures based on CNN and GRU, of which the connections are static, the CL scheme has little impact and only slightly improves the UA performance. For fair comparison, we apply this training scheme in all the following experiments.

D. Impact of Routing Number

Dynamic routing is the critical process for ensuring the grouping of similar lower-layer capsules to upper layers. The routing iteration can be conducted as many times as desired. A typical number chosen in previous works [23], [39]–[41] is 3. Small number of iteration may lead to insufficient grouping of the capsules. Whereas large iteration number can bring huge computational burden. It is desirable to check that whether the previous agreed iteration number of 3 still applies to the SER task with spectrograms as inputs. Fig. 7 shows the performance comparison of the system CNN_SeqCap with different routing numbers, from 1 to 4. As can be found that, the iteration number of 2 achieves the best performance (69.88% at WA, 56.97%

TABLE V
WA AND UA OF PROPOSED AND BASELINE SYSTEMS. "+" DENOTES SYSTEM COMBINATION BASED ON AVERAGE OF THE PREDICTED PROBABILITY VALUES

Systems	WA(%)	UA(%)
CNN [15]	66.1	56.6
CNN_LSTM [15]	68.80	59.40
CNN_GRU	67.02	51.84
CNN_GRU_Att	68.20	54.89
LSTM_Cap	67.31	53.70
CNN_SeqCap	69.86	56.71
CNN_RecCap	70.62	58.17
CNN_GRU-SeqCap	**72.73**	**59.71**
CNN_GRU-RecCap	70.62	57.58
CNN_GRU + CNN_SeqCap	72.64	57.15
CNN_GRU + CNN_RecCap	**72.73**	57.61

at UA). No improvements are observed with further iterations of routing. Also, there is no significant difference between the iteration numbers of 2 and 3. To ensure sufficient routing and follow previous works, we select 3 as the number of iterations for all the following experiments.

E. Effectiveness of Capsule Structures

Experimental results that compare the capsule-based systems with the baseline systems based on the WA and UA metrics provide evidences of superiority of the capsule structures.

Capsule vs. CNN: As shown in Table V, both the CNN_SeqCap and the CNN_RecCap systems outperform the baseline system CNN. The CNN_SeqCap has improvement of 3.76% at WA and 0.11% at UA, and the CNN_RecCap has improvement of 4.52% at WA and 1.57% at UA over the CNN system. The LSTM_Cap also performs better than the baseline CNN system. This shows the advantage of applying capsule structures to capture the spatial information. Our proposed systems, CNN_SeqCap and CNN_RecCap, achieve better results at both WA and UA than LSTM_Cap, which validates the effectiveness of the proposed sequential structure.

SeqCap vs. GRU: The CNN_SeqCap outperforms both the CNN_GRU and CNN_GRU_Att in Table V. We do the statistical analysis and find that the CNN_SeqCap outperforms CNN_GRU_Att at WA and UA both significantly with $p < 0.05$. To analyze the intermediate mechanisms for better understanding of the CNN_GRU_Att and the capsule structure CNN_SeqCap, we visualize the gradient values with respect to the input spectrograms in Fig. 8. As highlithed with red rectangles, the gradient values from the trained CNN_SeqCap are sensitive to the salient features in the spectrograms, but the values from the trained CNN_GRU_Att are insensitive. This provides evidence of the capsule structure advantage in capturing spectrogram spatial information.

RecCap vs. SeqCap: From Table V, it can be found that CNN_RecCap outperforms CNN_SeqCap (0.76% of WA marginally and 1.46% at UA significantly with $p < 0.05$), which demonstrates the effectiveness of the recurrent connections. In our experiments, we find that the CapNets have better performance in the emotions *Neutral*, *Angry* and *Sad*, but worse results for *Happy*, as shown in Table VI. This coincides with the results in previous works [20], [50] that the category of *Happy* is

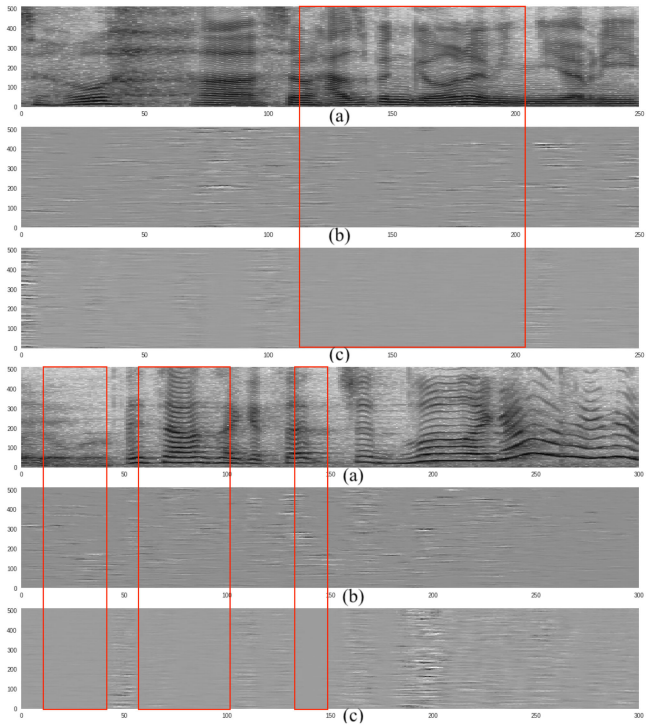


Fig. 8. Comparison of gradients with respect to the input spectrograms in (a), obtained from the trained CNN_SeqCap in (b) and CNN_GRU_Att in (c). The red rectangles highlight the comparison that the gradient values from CNN_SeqCap are sensitive to the salient features in the spectrograms, but the values from CNN_GRU_Att are insensitive.

TABLE VI
CONFUSION MATRIX OF CNN_SEQCAP

Actual/Predict	Neutral	Angry	Happy	Sad
Neutral	**84.22%**	5.26%	0.58%	9.94%
Angry	29.03%	**68.39%**	0%	2.58%
Happy	73.75%	18.64%	**1.69%**	5.92%
Sad	26.90%	0.29%	0.29%	**72.52%**

TABLE VII
CONFUSION MATRIX OF CNN_RECCAP

Actual/Predict	Neutral	Angry	Happy	Sad
Neutral	**83.85%**	2.92%	3.11%	10.12%
Angry	34.06%	**59.42%**	2.17%	4.35%
Happy	76.19%	5.56%	**11.90%**	6.35%
Sad	19.29%	1.93%	1.29%	**77.49%**

difficult to be recognized because of limited training data and its special emotive characteristics of relying on context contrastive information more than the other categories. From the results shown in Table VI and Table VII, RecCaps improve accuracies for *Happy* (from 1.69% to 11.9%) and *Sad* (from 72.52% to 77.49%) categories over SeqCaps, but accuracy for *Angry* declines. This also coincides with the previous observation in [50] that the two categories of *Happy* and *Angry* are similar in this corpus.

F. Combination of Capsule and GRU

The GRU and the SeqCap show superiority in capturing temporal and detailed spatial information, respectively. It is natural to ask the question whether the combination of SeqCap and GRU brings further improvement. In the evaluation of combining GRU and SeqCap, as shown in Table V, the CNN_GRU-SeqCap system outperforms all the other four systems of CNN, CNN_GRU, CNN_GRU_Att and CNN_SeqCap. This shows that the combination is beneficial. The WA performance of the combined system CNN_GRU-SeqCap is comparable with state-of-the-art performance 72.7% as reported in [32], but the UA performance still needs to be improved. It should be noted that in [32] both improvised and scripted data are used, but in this work we only use the improvised part following [15], [20]. We also investigate the combination of GRU and RecCap. The combined system CNN_GRU-RecCap outperforms the baseline systems of CNN and CNN_GRU, but is inferior to the system of CNN_RecCap. However, both the CNN_GRU-RecCap and the CNN_RecCap systems outperform the baseline of CNN and CNN_GRU. The other combination option is to use the average of generated probability values as predictions. The results in Table V show that simple system combination based on averaged probability also improves the WA performance.

It can be concluded that replacing GRUs in the CNN_GRU with capsules brings improvement overall, with RecCaps giving more gains than SeqCaps (3.42% vs. 2.66% at WA and 6.33% vs. 4.87% at UA significantly with $p < 0.005$). Augmenting GRUs with capsules brings improvement overall, with SeqCaps giving more gains than RecCaps (5.53% vs. 3.42% at WA and 7.87% vs. 5.74% at UA significantly with $p < 0.005$).

VII. Conclusion

In this paper, we devise the approach of applying capsule networks to the speech emotion recognition (SER) task. In order to capture the spatial information from input speech spectrograms, we propose a novel sequential capsule structure to obtain neural representations, and introduce recurrent connections to the sequential structure to capture the temporal information. The utterance-level dynamic routing is designed to obtain utterance representations for the final emotion prediction. Objective evaluations using the publicly available dataset IEMOCAP demonstrate the effectiveness of the proposed sequential capsule structure, CNN_SeqCap, and the recurrent capsule structure, CNN_RecCap. The CNN_SeqCap gives improvement of 3.76% at WA and 0.11% at UA, and the CNN_RecCap has improvement of 4.52% at WA and 1.57% at UA over the CNN system. In the future, we plan to enhance the recurrent connection to capture longer distance context to improve the recognition performance.

References

[1] M. E. Ayadi, M. Kamel, and F. Karray, "Survey on speech emotion recognition: Features, classification schemes, and databases," *Pattern Recognit.*, vol. 44, no. 3, pp. 572–587, 2011.

[2] M. M. E. Ayadi, M. S. Kamel, and F. Karray, "Speech emotion recognition using Gaussian mixture vector autoregressive models," *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, vol. 4, pp. 957–960, 2007.

[3] O.-W. Kwon, K. Chan, J. Hao, and T.-W. Lee, "Emotion recognition by speech signals," in *Proc. 8th Eur. Conf. Speech Commun. Technol.*, 2003, pp. 125–128.

[4] B. Schuller, A. Batliner, S. Steidl, and D. Seppi, "Recognising realistic emotions and affect in speech: State of the art and lessons learnt from the first challenge," *Speech Commun.*, vol. 53, no. 9–10, pp. 1062–1087, 2011.

[5] B. Schuller, G. Rigoll, and M. Lang, "Hidden Markov model-based speech emotion recognition," *Proc. Int. Conf. Multimedia Expo.*, vol. 2, pp. 401–404, 2003.

[6] S. Parthasarathy and C. Busso, "Semi-supervised speech emotion recognition with ladder networks," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 2697–2709, 2020, doi: 10.1109/TASLP.2020.3023632.

[7] M. Tahon and L. Devillers, "Towards a small set of robust acoustic features for emotion recognition: Challenges," *IEEE/ACM Trans. Audio, Speech Lang. Process.*, vol. 24, no. 1, pp. 16–28, Jan. 2016.

[8] J. Deng, X. Xu, Z. Zhang, S. Frühholz, and B. Schuller, "Semi-supervised autoencoders for speech emotion recognition," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 1, pp. 31–43, Jan. 2018.

[9] W. A. Jassim, R. Paramesran, and N. Harte, "Speech emotion classification using combined neurogram and INTERSPEECH 2010 paralinguistic challenge features," *IET Signal Process.*, vol. 11, no. 5, pp. 587–595, 2017.

[10] K. Han, D. Yu, and I. Tashev, "Speech emotion recognition using deep neural network and extreme learning machine," in *Proc. 15th Annu. Conf. Int. Speech Commun. Assoc.*, 2014, pp. 223–227.

[11] J. Lee and I. Tashev, "High-level feature representation using recurrent neural network for speech emotion recognition," in *Proc. 16th Annu. Conf. Int. Speech Commun. Assoc.*, 2015.

[12] S. Mirsamadi, E. Barsoum, and C. Zhang, "Automatic speech emotion recognition using recurrent neural networks with local attention," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2017, pp. 2227–2231.

[13] S. Latif, R. Rana, S. Khalifa, R. Jurdak, J. Epps, and B. W. Schuller, "Multi-task semi-supervised adversarial autoencoding for speech emotion recognition," *IEEE Trans. Affect. Comput.*, to be published, doi: 10.1109/TAFFC.2020.2983669.

[14] R. Li, Z. Wu, J. Jia, J. Li, W. Chen, and H. Meng, "Inferring user emotive state changes in realistic human-computer conversational dialogs," in *Proc. ACM Multimedia Conf. Multimedia Conf.*, ACM, 2018, pp. 136–144.

[15] A. Satt, S. Rozenberg, and R. Hoory, "Efficient emotion recognition from speech using deep learning on spectrograms," in *Proc. Conf. Int. Speech Commun. Assoc.*, 2017, pp. 1089–1093.

[16] D. Dai, Z. Wu, R. Li, X. Wu, J. Jia, and H. Meng, "Learning discriminative features from spectrograms using center loss for speech emotion recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 7405–7409.

[17] L. Guo, L. Wang, J. Dang, Z. Liu, and H. Guan, "Exploration of complementary features for speech emotion recognition based on kernel extreme learning machine," *IEEE Access*, vol. 7, pp. 75 798–75 809, 2019, doi: 10.1109/ACCESS.2019.2921390.

[18] L. Guo, L. Wang, J. Dang, L. Zhang, and H. Guan, "A feature fusion method based on extreme learning machine for speech emotion recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2018, pp. 2666–2670.

[19] P. Li, Y. Song, I. McLoughlin, W. Guo, and L. Dai, "An attention pooling based representation learning method for speech emotion recognition," in *Proc. Conf. Int. Speech Commun. Assoc.*, 2018, pp. 3087–3091.

[20] X. Ma, Z. Wu, J. Jia, M. Xu, H. Meng, and L. Cai, "Emotion recognition from variable-length speech segments using deep learning on spectrograms," in *Proc. Conf. Int. Speech Commun. Assoc.*, 2018, pp. 3683–3687.

[21] L. Zhang, L. Wang, J. Dang, L. Guo, and H. Guan, "Convolutional neural network with spectrogram and perceptual features for speech emotion recognition," in *Proc. Int. Conf. Neural Inf. Process.*, 2018, pp. 62–71.

[22] Z. Yang and J. Hirschberg, "Predicting arousal and valence from waveforms and spectrograms using deep neural networks," in *Proc. Conf. Int. Speech Commun. Assoc.*, 2018, pp. 3092–3096.

[23] S. Sabour, N. Frosst, and G. Hinton, "Dynamic routing between capsules," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 3856–3866.

[24] X. Wu *et al.*, "Speech emotion recognition using capsule networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 6695–6699.

[25] L. Guo, L. Wang, J. Dang, L. Zhang, H. Guan, and X. Li, "Speech emotion recognition by combining amplitude and phase information using convolutional neural network," in *Proc. Conf. Int. Speech Commun. Assoc.*, 2018, pp. 1611–1615.

[26] Y. Chen, L. Xu, K. Liu, D. Zeng, and J. Zhao, "Event extraction via dynamic multi-pooling convolutional neural networks," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics 7th Int. Joint Conf. Natural Lang. Process.*, 2015, pp. 167–176.
[27] Z. Aldeneh and E. M. Provost, "Using regional saliency for speech emotion recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2017, pp. 2741–2745.
[28] M. Wöllmer, A. Metallinou, N. Katsamanis, B. Schuller, and S. Narayanan, "Analyzing the memory of BLSTM neural networks for enhanced emotion classification in dyadic spoken interactions," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2012, pp. 4157–4160.
[29] W. Lim, D. Jang, and T. Lee, "Speech emotion recognition using convolutional and recurrent neural networks," in *Proc. Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, 2016, pp. 1–4.
[30] G. Keren and B. Schuller, "Convolutional RNN: An enhanced model for extracting features from sequential data," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, 2016, pp. 3412–3419.
[31] X. Zhu *et al.*, "Dependency exploitation: A unified CNN-RNN approach for visual emotion recognition," in *Proc. Int. Joint Conf. Artif. Intell.*, 2017, pp. 3595–3601.
[32] J. Wang, M. Xue, R. Culhane, E. Diao, J. Ding, and V. Tarokh, "Speech emotion recognition with dual-sequence LSTM architecture," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2020, pp. 6474–6478.
[33] Z. Zhao, Y. Zheng, Z. Zhang, H. Wang, Y. Zhao, and C. Li, "Exploring spatio-temporal representations by integrating attention-based bidirectional-LSTM-RNNs and FCNs for speech emotion recognition," in *Proc. of Conf. Int. Speech Commun. Assoc.*, 2018, pp. 272–276.
[34] S. Yoon, S. Dey, H. Lee, and K. Jung, "Attentive modality hopping mechanism for speech emotion recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2020, pp. 3362–3366.
[35] A. Nediyanchath, P. Paramasivam, and P. Yenigalla, "Multi-head attention for speech emotion recognition with auxiliary learning of gender recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2020, pp. 7179–7183.
[36] Y. Xie, R. Liang, Z. Liang, C. Huang, C. Zou, and B. Schuller, "Speech emotion classification using attention-based LSTM," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 11, pp. 1675–1685, Nov. 2019.
[37] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, pp. 933–941, 2017.
[38] G. E. Hinton, A. Krizhevsky, and S. D. Wang, "Transforming auto-encoders," in *Proc. Int. Conf. Artif. Neural Netw.*, Springer, 2011, pp. 44–51.
[39] W. Zhao, J. Ye, M. Yang, Z. Lei, S. Zhang, and Z. Zhao, "Investigating capsule networks with dynamic routing for text classification," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2018, pp. 3110–3119.
[40] X. Zhang, P. Li, W. Jia, and H. Zhao, "Multi-labeled relation extraction with attentive capsule network," *Proc. AAAI Conf. Artif. Intell.*, vol. 33, pp. 7484–7491, 2019.
[41] J. Bae and D. Kim, "End-to-end speech command recognition with capsule network," in *Proc. Conf. Int. Speech Commun. Assoc.*, 2018, pp. 776–780.
[42] M. Turan and E. Erzin, "Monitoring infant's emotional cry in domestic environments using the capsule network architecture," in *Proc. Conf. Int. Speech Commun. Assoc.*, 2018, pp. 132–136.
[43] M. A. Jalal, E. Loweimi, R. K. Moore, and T. Hain, "Learning temporal clusters using capsule routing for speech emotion recognition," in *Proc. Conf. Int. Speech Commun. Assoc.*, 2019, pp. 1701–1705.
[44] G. E. Hinton, S. Sabour, and N. Frosst, "Matrix capsules with EM routing," in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–15.
[45] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proc. Int. Conf. Mach. Learn.*, 2009, pp. 41–48.
[46] G. Hacohen and D. Weinshall, "On the power of curriculum learning in training deep networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 2535–2544.
[47] R. Girdhar and D. Ramanan, "Attentional pooling for action recognition," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 33–44.
[48] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 1–15.
[49] C. Busso *et al.*, "IEMOCAP: Interactive emotional dyadic motion capture database," *Lang. Resour. Eval.*, vol. 42, no. 4, pp. 335–359, 2008.
[50] X. Ma, Z. Wu, J. Jia, M. Xu, H. Meng, and L. Cai, "Speech emotion recognition with emotion-pair based framework considering emotion distribution information in dimensional emotion space," in *Proc. Conf. Int. Speech Commun. Assoc.*, 2017, pp. 1238–1242.

Xixin Wu (Member, IEEE) received the B.S. degree from Beihang University, Beijing, China, the M.S. degree from Tsinghua University, Beijing, China, and the Ph.D. degree from The Chinese University of Hong Kong, Hong Kong. He had been a Research Associate with the Machine Intelligence Laboratory, Cambridge University Engineering Department, and from 2021 has been a Research Assistant Professor with the Stanley Ho Big Data Decision Analytics Research Centre, The Chinese University of Hong Kong. His research interests include speech synthesis and recognition, speaker verification, and neural network uncertainty. He is a Member of ISCA.

Yuewen Cao (Member, IEEE) received the B.S. degree in communication engineering from the Huazhong University of Science & Technology, Wuhan, China, in 2017. She is currently working toward the Ph.D. degree with the Human-Computer Communications Lab (HCCL), The Chinese University of Hong Kong, China. Her research interests include speech synthesis and voice conversion.

Hui Lu (Member, IEEE) received the B.S. degree in communication engineering from Tongji University, Shanghai, China, in 2017. He received the M.S. degree in computer technology from Tsinghua University, Beijing, China, in 2020. He is currently working toward the Ph.D. degree with the Human-Computer Communications Lab (HCCL), The Chinese University of Hong Kong, Hong Kong, China. His research interests include speech synthesis and voice conversion.

Songxiang Liu (Member, IEEE) received the B.Eng. degree in automation from Department of Control Science and Engineering, Zhejiang University, Hangzhou, China, in 2016, and the Ph.D. degree from the Human-Computer Communications Laboratory (HCCL), The Chinese University of Hong Kong, Hong Kong, China, in 2021. His research interests include the broad field of spoken language processing, including speech and singing synthesis (e.g., voice transformation and text-to-speech synthesis), audio adversarial attacks and defense, etc.

Disong Wang (Graduate Student, Member, IEEE) received the B.S. degree in mathematics and physics basic science from the University of Electronic Science and Technology of China (UESTC) in 2015, and the M.E. in computer applied technology from Peking University (PKU), in 2018. He is currently a Ph.D. candidate with the Human Computer Communications Lab (HCCL) in the Chinese University of Hong Kong (CUHK). His research interests include voice conversion, text-to-speech synthesis, automatic speech recognition and their applications to non-standard speech, such as dysarthric speech and accented voice.

Zhiyong Wu (Member, IEEE) received the B.S. and Ph.D. degrees in computer science and technology from Tsinghua University, Beijing, China, in 1999 and 2005, respectively. From 2005 to 2007, he was a Postdoctoral Fellow with the Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong (CUHK), Hong Kong. He then joined the Graduate School at Shenzhen (now Shenzhen International Graduate School), Tsinghua University, Shenzhen, China, and is currently an Associate Professor. He is also a Coordinator with Tsinghua-CUHK Joint Research Center for Media Sciences, Technologies and Systems. His research interests include intelligent speech interaction, more specially, speech processing, audiovisual bimodal modeling, text-to-audio-visual-speech synthesis, and natural language understanding and generation. He is a Member of International Speech Communication Association and China Computer Federation.

Xunying Liu (Member, IEEE) received the Ph.D. degree in speech recognition and the M.Phil. degree in computer speech and language processing from the University of Cambridge, Cambridge, U.K., prior to his undergraduate study with Shanghai Jiao Tong University, Shanghai, China. He was a Senior Research Associate with Machine Intelligence Laboratory, Cambridge University Engineering Department, University of Cambridge, and since 2016, he has been an Associate Professor with the Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Hong Kong. His current research interests include large vocabulary continuous speech recognition, statistical language modelling, audio-visual speech processing, machine learning, language learning, speech synthesis and assistive technology. He and his students were the recipients of a number of best paper awards and nominations, including the Best Paper Award at ISCA Interspeech2010 for the paper titled Language Model Cross Adaptation for LVCSR System Combination and the Best Paper Award at IEEE ICASSP2019 for their paper titled BLHUC: Bayesian Learning of Hidden Unit Contributions for Deep Neural Network Speaker Adaptation. He is a Member of ISCA.

Helen Meng (Fellow, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the Massachusetts Institute of Technology, Cambridge, MA, USA. In 1998, she joined the Chinese University of Hong Kong, Hong Kong, where she is currently the Chair Professor with the Department of Systems Engineering & Engineering Management. She was the former Department Chairman and the Associate Dean of Research with the faculty of Engineering. Her research interests include human–computer interaction via multimodal and multilingual spoken language systems, spoken dialog systems, computer-aided pronunciation training, speech processing in assistive technologies, health- related applications, and Big Data decision analytics. She was the Editor-in-Chief of the IEEE TRANSACTIONS ON AUDIO, SPEECH AND LANGUAGE PROCESSING between 2009 and 2011. She was the recipient of the IEEE Signal Processing Society Leo L. Beranek Meritorious Service Award in 2019. She was also on the Elected Board Member of the International Speech Communication Association (ISCA) and an International Advisory Board Member. She is a ISCA, HKCS, and HKIE.