EmotionNAS: Two-stream Neural Architecture Search for Speech Emotion Recognition

Haiyang Sun\textsuperscript{1,2}\textsuperscript{*}, Zheng Lian\textsuperscript{2}\textsuperscript{*}, Bin Liu\textsuperscript{2}, Ying Li\textsuperscript{2}, Licai Sun\textsuperscript{1,2}, Cong Cai\textsuperscript{3,2}, Jianhua Tao\textsuperscript{3}, Meng Wang\textsuperscript{4}, Yuan Cheng\textsuperscript{4}

\textsuperscript{1}School of Artificial Intelligence, University of Chinese Academy of Sciences
\textsuperscript{2}State Key Laboratory of Multimodal Artificial Intelligence Systems
\textsuperscript{3}Department of Automation, Tsinghua University, \textsuperscript{4}Ant Group

\{sunhaiyang2021, lianzheng2016\}@ia.ac.cn

Abstract

Speech emotion recognition (SER) is an important research topic in human-computer interaction. Existing works mainly rely on human expertise to design models. Despite their success, different datasets often require distinct structures and hyperparameters. Searching for an optimal model for each dataset is time-consuming and labor-intensive. To address this problem, we propose a two-stream neural architecture search (NAS) based framework, called “EmotionNAS”. Specifically, we take two-stream features (i.e., handcrafted and deep features) as the inputs, followed by NAS to search for the optimal structure for each stream. Furthermore, we incorporate complementary information in different streams through an efficient information supplement module. Experimental results demonstrate that our method outperforms existing manually-designed and NAS-based models, setting the new state-of-the-art record.

Index Terms: speech emotion recognition, neural architecture search, two-stream model, information supplement module.

1. Introduction

Speech emotion recognition (SER) has received increasing attention due to its contribution to human-computer interaction [1, 2, 3]. SER aims to understand how humans express their emotions and then classify each utterance into its emotional state [4, 5]. Existing works are mainly manually-designed models [6, 7, 8, 9]. Despite their success, these works rely on historical experience to design model structures, which is often time-consuming and labor-intensive [10, 11]. Therefore, how to design networks more intelligently has been brought into focus.

To this end, we explore neural architecture search (NAS) for SER [12, 13]. By setting the search space, the search strategy, and the evaluation metric, we can optimize the model architecture automatically with little human intervention.

Previously, search methods based on reinforcement learning [14] or evolutionary algorithms [15] avoided the exhaustive search and provided deep learning-inspired design principles. However, these methods need the retraining process every time a new substructure is sampled, which consumes numerous computational resources. To speed up the search process, researchers propose differentiable architecture search (DARTS) [10, 16, 17, 18], which relaxes the search space to be continuous by applying a softmax operation. Additionally, some efforts [19, 20] build a model library by pre-training multiple structures. These methods successfully alleviate the time-consuming process in previous works. In this paper, we rely on more efficient search methods to design SER models.

\*Equal contribution \textsuperscript{\dagger}Corresponding Author

In addition to model design, another challenge in SER is how to represent audios [21]. Among existing handcrafted features, spectrograms are widely utilized in SER [22, 23, ? 24]. However, the spectrogram loses some phase information during calculation. Recently, the deep feature, wav2vec [25], has demonstrated its effectiveness in speech representation learning [26]. It is a self-supervised framework that can learn powerful acoustic representations with the help of large amounts of unlabeled data. However, wav2vec may lose some emotion-related information due to different training objectives. To obtain a more comprehensive speech representation, we integrate handcrafted and deep features via a two-stream framework.

Meanwhile, how to fuse multiple features also affects the classification performance. Unsuitable fusion approaches cannot effectively utilize the complementary information in different features, which may lead to performance degradation compared to the best-performing feature (denoted as the dominant feature in this paper). Therefore, we further design an effective fusion method, called “information supply module (ISM)”, to ensure that the performance of the dominant feature is not degraded and can be further improved.

In summary, to address the low efficiency of the manually-designed approach and inefficient fusion of different features, we propose a novel two-stream framework called “EmotionNAS”. Figure 1 shows the overall structure of our proposed method. Experimental results show that EmotionNAS outperforms existing manually-designed and NAS-based models. The contributions of this paper can be summarized as follows:

- We propose a novel framework called “EmotionNAS”. It incorporates complementary information in handcrafted and deep features, followed by NAS to efficiently search through numerous possible networks to find the optimal structure.
- We further design ISM to effectively fuse complementary information in different features.
- Experimental results on IEMOCAP show that our method successfully outperforms existing manually-designed and NAS-based models, setting the new state-of-the-art record.

2. Methodology

EmotionNAS is a two-stream framework that combines handcrafted and deep features for SER. To search for the optimal structure, we explore NAS in model design. Subsequently, we employ ISM to fuse two-stream features effectively.

2.1. Architecture Search for Spectrogram

To better utilize the time-frequency two-dimensional information in the spectrogram, our method automatically optimizes the CNN-based structure through an efficient architecture search al-
Figure 1: The overall structure of EmotionNAS. It is a two-stream architecture that takes spectrogram and wav2vec as the inputs, followed by NAS to design the model automatically. We further fuse the outputs of two branches through ISM to achieve better performance.

![Diagram](image)

Figure 2: DARTS architecture search in a cell: (a) the initial state of a cell with an ordered sequence of $N = 4$ nodes and $|\mathcal{O}| = 3$ operations. Different-colored arrows represent distinct operations; (b) during training, some operations’ weights are reduced, resulting in lighter-colored lines; (c) only the operations with the highest weight are retained.

Algorithm DARTS [16]. It divides the whole network into normal and reduction cells. All operations adjacent to the reduction cells have a stride of 2, while others have a stride of 1.

Figure 2 shows the overall search process in a cell. Suppose each cell consists of an ordered sequence of $N$ nodes. Let $x^{(i)}$ denotes the latent feature of the $i$-th node. Assume $\mathcal{O}$ is a set of candidate operations, and $|\mathcal{O}|$ represents the number of operations. To make the search space continuous, DARTS incorporates all possible operations through weights. We take the operations from $x^{(i)}$ to $x^{(j)}$ as an example:

$$\alpha_{o}^{(i,j)} = \text{softmax} \left( \theta_{o}^{(i,j)} \right),$$  \hspace{1cm} (1)

$$x^{(j)} = \sum_{o \in \mathcal{O}} \alpha_{o}^{(i,j)} o(x^{(i)}),$$  \hspace{1cm} (2)

where $\theta_{o}^{(i,j)}$ and $\alpha_{o}^{(i,j)}$ represent the raw and normalized weights of the operation $o \in \mathcal{O}$ from $x^{(i)}$ to $x^{(j)}$, respectively. The task of architecture search reduces to learning a set of continuous weights $\{\alpha_{o}^{(i,j)}\}$. At the end of the search, we only keep the operation with the highest weight.

2.2. Architecture Search for Wav2vec

Since wav2vec contains temporal information, we search for the optimal RNN-based structure for this branch [20]. The search space includes all regular operations in RNN, such as linear mapping, blending, activation function, and element-wise operation. The optimized cell structure is shared across different timesteps. For timestep $t$, the initial nodes consist of the input vector $x_t$ and two hidden states $h_{t-1}$ and $h_{t-1}$. We aim to optimize the cell architecture and generate new hidden states for the next timestep, i.e., $h_t^1$ and $h_t^2$ (see Section 4.6 for details).

At the end of the search, we feed wav2vec into the optimized recurrent neural architecture to generate frame-level representations. Since different frames play different parts in SER, we further exploit an attention mechanism to prioritize important frames and fuse them [27].

2.3. Information Supplement Module

After finding the optimal structure for each stream, we propose an ISM to fuse complementary information from two streams. To ensure that the performance of the dominant feature is not degraded after fusion, we abandon to compress this feature. Instead, we select complementary information from other inputs and fuse them with the dominant feature.

Specifically, suppose the outputs of two branches are denoted as $X_1 \in \mathbb{R}^{d_1}$ and $X_2 \in \mathbb{R}^{d_2}$, respectively. Here, $d_1$ and $d_2$ are the feature dimensions of $X_1$ and $X_2$. Suppose $X_2$ is the dominant feature. The calculation formula of our ISM can be summarized as follows:

$$H_1 = \text{Proj}(X_1),$$  \hspace{1cm} (3)

$$A_1 = \tanh(\text{Proj}(X_2) \odot H_1),$$  \hspace{1cm} (4)

$$F = X_2 + A_1 \odot H_1,$$  \hspace{1cm} (5)

where $\text{Proj}()$ and $\odot$ represent a linear transformation function and an element-wise multiplication, respectively. Here, $H_1 \in \mathbb{R}^{d_2}$ and $A_1 \in \mathbb{R}^{d_2}$. In the end, we leverage the fused feature $F \in \mathbb{R}^{d_2}$ for emotion recognition.

3. Experimental Database and Setup

We first describe the benchmark dataset in our experiments. Following that, we illustrate the feature extraction process, implementation details, and various current advanced baselines.

3.1. Database

IEMOCAP [28] is a benchmark dataset for SER. It consists of five sessions, each with two speakers. For a fair comparison, we adopt five-fold cross-validation with the leave-one-session-out strategy [9]. Eight speakers from four sessions are used as the training set. One speaker in the remaining session is used as the validation set and the other as the test set. We evaluate four emotions (i.e., neutral, angry, happy and sad) on improvised data, in line with previous works [9]. Due to the imbalanced class distribution, we use unweighted accuracy (UA) as the primary metric and also report weighted accuracy (WA).

3.2. Feature Extraction

In EmotionNAS, we take spectrogram and wav2vec as inputs. The extraction processes are described as follows:

**Spectrogram:** We utilize the librosa toolkit [29] to extract spectrograms from audio. We unify the duration of audio into 8 seconds by zero padding and truncation. Then, a spectrogram is extracted with 25ms Hamming windows and 14ms overlap. Finally, we downsample the spectrogram to $140 \times 140$ using an average pooling operation.

**Wav2vec:** We use the pretrained wav2vec-large [25] as the acoustic feature extractor. We unify the wav2vec feature to the maximum length by zero padding, followed by average pooling to downsample it to $727 \times 512$.  

3598
Table 1: Performance of different approaches. Bold font represents the best performance.

<table>
<thead>
<tr>
<th>Method</th>
<th>UA(%)</th>
<th>WA(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN-GRU [6]</td>
<td>64.2</td>
<td>71.5</td>
</tr>
<tr>
<td>CTC-RNN [7]</td>
<td>65.7</td>
<td>64.2</td>
</tr>
<tr>
<td>SeqCap [8]</td>
<td>59.7</td>
<td>72.7</td>
</tr>
<tr>
<td>PCNSE-SADRN-CTC [9]</td>
<td>66.3</td>
<td>73.1</td>
</tr>
<tr>
<td>UniformNAS [32]</td>
<td>56.9</td>
<td>70.5</td>
</tr>
<tr>
<td>EmotionNAS (Ours)</td>
<td>69.1</td>
<td>72.1</td>
</tr>
</tbody>
</table>

Table 2: Performance comparison between NAS-based methods and ResNet. * represents our fair comparison by adjusting the model structure with the similar number of parameters.

<table>
<thead>
<tr>
<th>Spectrogram Branch</th>
<th>Params</th>
<th>UA(%)</th>
<th>WA(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResNet18</td>
<td>11.18M</td>
<td>59.4</td>
<td>60.7</td>
</tr>
<tr>
<td>ResNet*</td>
<td>1.23M</td>
<td>55.2</td>
<td>58.4</td>
</tr>
<tr>
<td>NAS-C4L3</td>
<td>0.07M</td>
<td>52.1</td>
<td>54.4</td>
</tr>
<tr>
<td>NAS-Col3</td>
<td>0.13M</td>
<td>57.3</td>
<td>63.2</td>
</tr>
<tr>
<td>NAS-C8L4</td>
<td>0.35M</td>
<td>58.6</td>
<td>62.2</td>
</tr>
</tbody>
</table>

3.3. Implementation Details

Our method consists of three key modules: 1) NAS for spectrogram; 2) NAS for wav2vec; 3) ISM for fusion. Since wav2vec can achieve better performance in SER, we treat this feature as the dominant feature. For the search process, we set hyperparameters based on validation performance.

NAS for spectrogram: In each cell, we set the number of nodes \( N = 4 \) and the number of operations \( |O| = 8 \). The search space contains regular operations in CNNs, such as \( 3 \times 3 \) max pooling, \( 3 \times 3 \) average pooling, skip connection, \( 3 \times 3 \) separable convolution, \( 5 \times 5 \) separable convolution, \( 3 \times 3 \) dilated convolution, \( 5 \times 5 \) dilated convolution, and no connection. At the same time, we set the initial number of channels \( C = 6 \) and the number of layers \( L = 3 \).

NAS for wav2vec: The search space contains regular operations in RNNs, such as linear function, blending, element-wise product, element-wise sum, tanh function, sigmoid function, and Leaky ReLU function. Previous work [20] provided various RNN-based cell structures searched on the Penn Tree Bank dataset [30]. We choose the structure with the lowest validation loss in the emotion dataset. Meanwhile, we set the number of layers \( L = 1 \) and the hidden feature dimension \( H = 256 \).

3.4. Baselines


4. Results and Discussion

In this section, we first conduct comparative experiments with currently advanced systems to verify the effectiveness of our method. Then, we systematically investigate the importance of each module in EmotionNAS, including the NAS-based approach, two-stream framework, and ISM fusion strategy. Finally, we visualize the confusion matrices and search results.

4.1. Comparison with Existing Works

To verify the effectiveness of our method, we treat current advanced approaches as our baselines. These methods include existing manually-designed and NAS-based models with consistent data division. We report baseline results according to their original papers. Experimental results in Table 1 show that EmotionNAS outperforms most existing methods and achieves a performance improvement of 2.8% over manually-designed models [9]. Compared with these approaches, our NAS-based framework can optimize structures and find more effective models than manually-designed approaches, automatically.

Meanwhile, we observe that EmotionNAS outperforms UniformNAS [32], a NAS-based approach, by 12.2% on UA and 1.2% on WA. But limited by handcrafted features, this baseline cannot leverage the useful information from deep features. Our framework, by contrast, not only enables efficient network design but also effectively fuses handcrafted and deep features, bringing better performance on SER.

4.2. Advantage of NAS

In this section, we compare the performance of several NAS-based and manually-designed models. Experimental results are shown in Table 2–3. For the spectrogram branch, NAS-based methods can achieve better performance with fewer parameters (see Table 2). For the wav2vec branch, our method also outperforms RNN and LSTM in emotion recognition (see Table 3). These results reveal the advantage of NAS in model design.

4.3. Necessity of Two Branches

In this section, we further reveal the necessity of two branches. As shown in Table 4, spectrogram and wav2vec achieve 57.3% and 66.2% on UA, 63.2% and 70.3% on WA, respectively. After multi-branch fusion, we further improve UA to 69.1% and
Table 5: Importance of ISM.

<table>
<thead>
<tr>
<th>Method</th>
<th>UA(%)</th>
<th>WA(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dominant branch (wav2vec)</td>
<td>66.2</td>
<td>70.3</td>
</tr>
<tr>
<td>Concat</td>
<td>67.1</td>
<td>68.5</td>
</tr>
<tr>
<td>Sum</td>
<td>63.8</td>
<td>66.0</td>
</tr>
<tr>
<td>Max</td>
<td>65.0</td>
<td>67.5</td>
</tr>
<tr>
<td>Min</td>
<td>68.1</td>
<td>65.2</td>
</tr>
<tr>
<td>EmotionNAS (Ours)</td>
<td>69.1</td>
<td>72.1</td>
</tr>
</tbody>
</table>

Figure 3: Visualization of confusion matrices. (a) Results of Min. (b) Results of EmotionNAS.

WA to 72.1%. These results confirm that handcrafted and deep features contain complementary information in emotion recognition. Through the fusion process, we can achieve better classification performance on SER.

4.4. Effectiveness of ISM

To verify the effectiveness of our method, we compare ISM with other fusion strategies. Followed with the symbols in Section 2.3, we denote the outputs of two branches as $X_1$ and $X_2$. Here, $X_1$ is the dominant feature and $H_1$ is the linear transformation of $X_1$. Concat concatenates $X_1$ and $X_2$. Sum, Max and Min denote the corresponding operations on $H_1$ and $X_2$.

Experimental results are listed in Table 5. Among the four comparison strategies, only Min and Concat surpass the performance of the dominant branch. It indicates that most strategies cannot effectively fuse complementary information in different features. But through ISM, we can significantly improve the performance of UA and WA. These results demonstrate the effectiveness of our fusion strategy.

4.5. Visualization of Confusion Matrices

Figure 3 shows the confusion matrices for Min and EmotionNAS. We observe that EmotionNAS achieves better performance in most emotion categories, especially happy and angry. Our findings also highlight the limitations of traditional fusion strategies. Compared with Min, our fusion method ISM can effectively fuse multiple features and achieve better performance.

4.6. Visualization of Search Results

In this section, we further visualize the search results for each branch. Figure 4(a)~4(b) show the search results for the spectrogram branch. There are two types of cells: the normal cell and the reduction cell. For cell $k$, the initial nodes consist of the outputs in previous cells, i.e., $c_{k-2}$ and $c_{k-1}$. Figure 4(c) shows the search result for the wav2vec branch. For timestep $t$, the initial nodes consist of the input vector $x_t$ and two hidden states $h^1_{t-1}$ and $h^2_{t-1}$. We aim to generate new hidden states for the next timestep, i.e., $h^1_t$ and $h^2_t$.

5. Conclusions

In this paper, we propose a novel framework for SER, called “EmotionNAS”. It takes spectrogram and wav2vec as the inputs, followed by NAS to optimize the network structure automatically. We further design ISM to integrate complementary information in different branches. Experimental results demonstrate that our method outperforms existing manually-designed and NAS-based methods. Meanwhile, we also systemically prove the importance of each component in EmotionNAS, including the NAS-based method, the two-stream framework and the ISM fusion strategy.

6. Acknowledgments

This work is supported by the National Natural Science Foundation of China (NSFC) (No.61831022, No.62276259, No.62201572, No.U21B2010), Beijing Municipal Science&Technology Commission, Administrative Commission of Zhongguancun Science Park No.Z211100004821013, Open Research Projects of Zhongqian Lab (NO.2021KH0AB06), CCF-Baidu Open Fund (No.OF2022025).
7. References


