Coughing-based Recognition of Covid-19 with Spatial Attentive ConvLSTM Recurrent Neural Networks

Tianhao Yan¹,², Hao Meng¹, Emilia Parada-Cabaleiro³, Shuo Liu², Meishu Song², Björn W. Schuller²,⁴

¹College of Intelligent Systems Science and Engineering, Harbin Engineering University, Harbin, 150001, China
²Chair of Embedded Intelligence for Health Care and Wellbeing, University of Augsburg, Germany
³Institute of Computational Perception, Johannes Kepler University Linz, Austria
⁴GLAM – Group on Language, Audio, & Music, Imperial College London, UK
menghao@hrbeu.edu.cn

Abstract
The rapid emergence of COVID-19 has become a major public health threat around the world. Although early detection is crucial to reduce its spread, the existing diagnostic methods are still insufficient in bringing the pandemic under control. Thus, more sophisticated systems, able to easily identify the infection from a larger variety of symptoms, such as cough, are urgently needed. Deep learning models can indeed convey numerous signal features relevant to fight against the disease; yet, the performance of state-of-the-art approaches is still severely restricted by the feature information loss typically due to the high number of layers. To mitigate this phenomenon, identifying the most relevant feature areas by drawing into attention mechanisms becomes essential. In this paper, we introduce Spatial Attentional ConvLSTM-RNN (SACRNN), a novel algorithm that is using Convolutional Long-Short Term Memory Recurrent Neural Networks with embedded attention that has the ability to identify the most valuable features. The promising results achieved by the fusion between the proposed model and a conventional Attentional Convolutional Recurrent Neural Network, on the automatic recognition of COVID-19 coughing (78.3% of Unweighted Average Recall) show the great potential of the presented approach in developing efficient solutions to defeat the pandemic.

Index Terms: SARS-CoV-2 detection, Computer Audition, Spatial Attention, CNN, RNN, LSTM, Spectrogram, ACRNN.

1. Introduction
Since 2019, human beings are threatened by the COVID-19 pandemic, declared by the World Health Organization as a global public health crisis [1] [2]. The dramatic death rate, devastated economy, and individuals’ freedom restrictions derived by the ongoing pandemic, continue to have an extreme negative impact on human life [3] [4]. To overcome this critical situation, one of the main difficulties that need to be addressed is the still insufficient detection of the disease, something essential to control its spread. To this end, developing mechanisms that identify infected individuals at an early stage is crucial [5] [6]. In order to cope with this medical challenge, the astonishing achievements of deep learning (DL) algorithms have expeditiously turned them into powerful tools for medical diagnosis [7], showing to be very promising in the COVID-19 detection, too [8].

State-of-the-art research aimed to identify COVID-19 through DL methods is typically based on convolutional neural networks (CNN), especially when considering medical images, such as chest X-ray, as input data [9]. Nevertheless, the performance of CNN is typically restricted by the feature information loss characteristic of very deep architectures. In this regard, recurrent neural networks (RNN), such as long-short term memory (LSTM) RNN— particularly suited for classifying continuous data due to their capability to remember characteristics of the signal displayed earlier— have been presented in order to alleviate the degradation problem [10]. Yet, to mitigate the information loss, identifying the most relevant feature areas by drawing into attention mechanisms is also essential, something especially promoted by the breakthrough of Transformers in natural language processing (NLP), whose attention mechanism enables to efficiently identify the most relevant feature area of the network. Indeed, this advance has already shown successful results for NLP-based COVID-19 applications [11].

Since early detection is still an open challenge in the fight against the pandemic, efficient methods able to easily detect the infection from a broader variety of symptoms are particularly necessary. In this regards, the development of coughing-based systems for automatic COVID-19 recognition are especially useful in supporting the ongoing DL-based diagnostic tools [8]. To this end, we propose Spatial Attentional ConvLSTM-RNN (in short, SACRNN), a novel deep learning architecture based on convolution recurrent neural networks with embedded attention for the automatic detection of COVID-19 coughing [12]. Our approach employs a CNN to extract the most relevant characteristics of the FFT-based Mel-spectrogram representation of the audio signal. Subsequently, after passing a series of LSTM layers, the feature maps are connected to an Attention structure and to a GlobalAveragePooling Layer. Finally, the model is enhanced by fusing it to a conventional Attentional Convolutional Recurrent Neural Network (ACRNN) [13].

2. Related Work
The acoustic features typically used for speech classification are mainly divided into two categories: 1) the ‘traditional’ features, which include frame-based Low Level Descriptors (LLDs) and utterance-based High-level Statistical Functionals (HSFs) [14] [15] [16]; 2) the spectrogram-based features [17] [18]. In the past, the traditional features, typically extracted with toolkits such as openSMILE [19], were feed into machine learning algorithms, e.g., Support Vector Machine (SVM), XGBoost, or Random Forests [20] [21] [22]. Nowadays, with the advent of DL, researchers began to successfully feed DL models with raw audio representations [23], such as spectrograms [24].

Inspired and benefiting from the achievements reached in Computer Vision and NLP, most of the speech classification research presented in the recent years is based on DL models,
Voice Activity Detection (VAD) on the audio signal to cut out the silent frames at the beginning and the end of the recordings. Subsequently, we generate three types of Mel-spectrograms: static, deltas, and delta-deltas; which will be used as input of the DL model. The proposed model incorporates a CNN structure on top of a RNN aimed to enhance the effect of the attention mechanism. In specific, the spatial block framework, followed by a global-average-pooling layer and a softmax layer, is fused with the attention mechanism. In order to focus on specific image target areas. When the spectrogram feature maps from audio signals (analogous to the channel in images) present certain time-frequency domain characteristics, from which attentive feature information can be extracted, this has not yet been taken into consideration by any of the already presented approaches. Considering this, and inspired by the outstanding results achieved in computer vision by the use of specific attentional ConvLSTM architectures [31], we add an attentive LSTM structure to the output of a CNN in order to specifically extract more centralised temporal-frequency feature maps. By extracting the features of the LSTM-Attention layer in the time dimension of the spectrograms, these can be connected with an AttentiveConvLSTM layer, so that the model has the ability to locate and re-extract the most relevant features of the audio signal.

3. Model Architecture

To build the proposed architecture (SACRNN), we first apply Voice Activity Detection (VAD) on the audio signal to cut out the silent frames at the beginning and the end of the recordings. Subsequently, we generate three types of Mel-spectrograms: static, deltas, and delta-deltas; which will be used as input of the DL model. The proposed model incorporates a CNN structure on top of a RNN aimed to enhance the effect of the attention mechanism. In specific, the spatial block framework, followed by a global-average-pooling layer and a softmax layer, is fused with a conventional ACRNN framework. Note that the two branch models share the same convolutional layer, as shown in Figure 1.

3.1. Mel-Spectrogram Generation

In order to focus on the coughing signal while disregarding static frames, VAD is applied through the Python data package `webRTCvad`. Following [32], who showed that 2-dimensional convolutions are more efficient than the 1-dimensional ones in small datasets, 2-dimensional Mel-spectrograms are generated through Fast Fourier Transform (FFT) from the audio signal. In order to better capture the dynamic information of the signal, deltas and delta-deltas are also extracted from the Mel-spectrograms. The samples are set to a sample rate of 16 kHz and the Mel-spectrograms are generated considering 40 filterbanks and a Hamming window of 25 ms with a shift of 10 ms. The energy spectrum is computed by passing the Short-Time Fourier Transform (STFT) of the raw signal through the Mel-filterbanks, by this obtaining the Mel-spectrogram $M$. Subsequently, the logarithm of the generated Mel-spectrogram is performed, as shown in Equation (1).

$$M \left( i \right) = \frac{\sum_{k=1}^{f(i+1)} \log \left( H_i \left( k \right) \right) * \left| X \left( k \right) \right|^2}{\sum_{n=1}^{N} n^2}, \quad (1)$$

where $\left| X \left( k \right) \right|^2$ represents the energy spectrum, $H_i \left( k \right)$ indicates the Mel-filterbanks, $k$ is the output of the FFTs, and $i$ is the number of filterbanks. Subsequently, from the 1-channel static Log-Mel Spectrum, deltas and double-deltas are calculated as shown in Equation (2) and Equation (3) respectively. Thus, leading to the 3-channel Spectrogram structure.

$$M \left( i \right)^d = \frac{\sum_{n=1}^{N} n \left( M \left( i \right)_{t+n} - M \left( i \right)_{t-n} \right)}{2 \sum_{n=1}^{N} n^2}, \quad (2)$$

$$M \left( i \right)^{dd} = \frac{\sum_{n=1}^{N} n \left( M \left( i \right)^d_{t+n} - M \left( i \right)^d_{t-n} \right)}{2 \sum_{n=1}^{N} n^2}, \quad (3)$$

where $t$ is the number of frames, $d$ indicate the number of delta, $n$ represents the number of differences between the current frame and the previous-next frames, and $N$ is set to 2 under normal situation. Hence, the output dimension of the 3-channel spectogram, which represents the values of static, deltas, and delta-deltas, respectively (input for the CNN), is $M \in \mathbb{R}^{t \times f \times e}$, where $t$ indicates the time, $f$ is the number of filters, and $e$ is the number of channels.

3.2. Model

From the image domain perspective, adding Attention layers has as a goal to focus on specific image target areas.
transferring this idea to the audio domain, finding the target area requires more effort. To do so, beyond adding the time dimension to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.1. CNN Model

The 3-channel 2-dimensional Mel-Spectrograms (cf. Section 3.1) were used as input for a CNN with 2 convolutional blocks. Each block contains two identical convolutional layers which include the same kernels and sizes (cf. Table 1). First, we connect a maxpooling layer with a pooling size of 2*2 after the first convolutional block. Different from the second maxpooling layer, the size is equals 1*2. In order to preserve the original features to the maximum, similar to the ResNet structure, we make a skip connection between the output of the first and second layers of convolution.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the CNN output with the time axis (iteration axis) by adding the convolution in the calculation, we can make use of the LSTM layer to better retain the characteristics. First, we combine the CNN output with the time axis (iteration axis) by adding the hidden state $H_{t-1}$ to the Tanh activation function, which is performed by a convolutional layer with a kernel of 1 named $R_t$, as shown in Equation (4).

$$Z_t = R_t \ast \text{tanh}(W_t \ast X + H_{t-1} + b),$$  \hspace{1cm} (4)$$

where, $W_t$ and $U_t$ represent the convolutional kernel of $X$ and $H_{t-1}$ respectively. Subsequently, the attention mechanism is carried out by applying Softmax (in the spatial dimension) to the 2-dimensional tensor, as shown in Equation (5).

Table 1: Parameters of the used SACRNN model

<table>
<thead>
<tr>
<th>Parameters</th>
<th>OutputDim</th>
<th>Kernel Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conv. Block</td>
<td>$T \ast N \ast 32$</td>
<td>$3 \ast 3$</td>
</tr>
<tr>
<td>MaxP. Layer</td>
<td>$T/2 \ast N/2 \ast 32$</td>
<td>$2 \ast 2$</td>
</tr>
<tr>
<td>Conv Block</td>
<td>$T/2 \ast N/2 \ast 64$</td>
<td>$3 \ast 3$</td>
</tr>
<tr>
<td>MaxP. Layer</td>
<td>$T/2 \ast N/4 \ast 64$</td>
<td>$1 \ast 2$</td>
</tr>
<tr>
<td>Reshape</td>
<td>$T \cdot S \ast T/2 \ast N/4 \ast 128$</td>
<td>-</td>
</tr>
<tr>
<td>• AttConvLSTM</td>
<td>$T/2 \ast N/4 \ast 128$</td>
<td>-</td>
</tr>
<tr>
<td>Global Avg</td>
<td>128</td>
<td>-</td>
</tr>
<tr>
<td>Dense (Dropout)</td>
<td>(rate = 0.2)</td>
<td>64</td>
</tr>
<tr>
<td>Softmax</td>
<td>$K$</td>
<td>-</td>
</tr>
</tbody>
</table>

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.
after employing the LSTM Layer, another round of attention mechanism is performed for the time step. This is preferred, instead of simply performing a global average pooling over time, in order to score the importance of a series of high-level representations. Finally, the output from the above structures is sent to a 64-unit fully connected layer, by this gaining a high-level representation.

4. Experiment

4.1. Dataset Description

For our experiments, a subset of the Cambridge COVID-19 Sound database [34], as used in the COVID-19 Cough Sub-Challenge (CCS) from the INTERSPEECH 2021 Computational Paralinguistics Challenge3 (ComParE) [12], is considered. The data, collected in a crowdsourcing manner through the COVID-19 Sounds App, has a length of 1.63 hours and encompasses a total of 929 samples produced by 397 participants (each producing at least 3 samples). To collect the data, participants with both, positive and negative COVID-19 tests, were requested to provide forced coughs.

4.2. Experimental Setup

For the experiments, the partitioning proposed in the COVID-19 CCS is followed as given in the Challenge for experiments, where the sets are 286 samples for training, 231 for validation, and 208 for testing. As evaluation metrics, since the distribution between the two classes in the test set is imbalanced, both the accuracy, i.e., the Weighted Average Recall, as well as the Unweighted Average Recall (UAR) are considered. In addition, as common practice in the medical-related research, the results on the test set will also be reported in terms of specificity, sensitivity, and confidence intervals. Due to the small size of the dataset, in order to optimise parallel acceleration, after extracting the spectrograms, the coughing samples were split into sub-segments of 300 frames. For sub-segments shorter than 300 frames and longer than 100, zero padding was applied at the end. Note that the segmentation was performed on all the sets and the training set was fixed for both validation and test phases.

The model is trained with the sub-segments, which in the testing phase are considered to assess whether the prediction coincides with the one of the real label. Then, the proportion of sub-segments successfully predicted (Voting Rule) is computed [13]. If the proportion is greater than 0.5 w.r.t. the whole sample, the prediction is considered as the real label, otherwise it is regarded as a failed prediction. Before training, the coughing data of all sets is normalised by the overall average and variance of the training set. For comparability, along to the outcomes achieved by the proposed architecture, results for the end-to-end learning End2You approach [12], as well as the results from the SACRNN and the ACRNN individually, i.e., without fusion, are also given. To carry out the experiments, the overall architecture was implemented with Keras [35], using the Adam optimiser with Nestov momentum [36]. Concerning the number of layers, different depth levels were considered to find the optimal configuration.

5. Results and Discussion

In Table 3, our results show that the UAR and sensitivity are lower for SACRNN: 58.4 % and 17.95 % respectively, which indicates that the SACRNN alone is not sufficiently sensitive to recognise positive coughing. However, it can be seen from specificity that the SACRNN is close to 100 % in the classification of negative coughing, which indicates that our proposed model is valuable for the negative data. We interpret that this is due to the fact that the SACRNN shows excessive strengths concerning the features based on the spatial spectrogram while it ignores the inherent time-domain information, which results in lower recognition for positive coughing. Indeed, an improvement is shown when employing the ACRNN alone (70.91 % and 51.28 % for UAR and sensitivity), which confirms the importance of the continuous temporal characteristics. In this case, as expected, the recognition rate of negative coughing drops, as shown by the lower specificity (90.53 %), hence the fusion model taking advantage from both, the SACRNN and the ACRNN, presents the best performance, exceeding the baseline deep learning model (End2You) algorithm by 8.5 % UAR (cf. 73.2 % vs 64.7 % for Fusion vs End2You in Table 3). The results for sensitivity and specificity in the fusion model (56.41 % and 89.94 %, respectively) show the robustness and generalisation ability of the proposed approach, which is particularly efficient even with noisy samples – as those typically collected in-the-wild.

6. Conclusion

In this paper, we proposed a novel approach based on spatial attentive convolutional recurrent neural networks (SACRNN) based on three-channel spectrogram-based features fused with ACRNN structure. The promising results of our approach for COVID-19 coughing detection, acquiring 73.2 % UAR and considerably outperforming the baseline results, shows that it can be successfully applied to detect infected users, by this supporting medical diagnosis at early stage of the disease, something urgently needed in the current global crisis. In addition, the fusion algorithm, which connect our proposed model and the conventional ACRNN, constitutes a novel idea with potential applications in speech classification tasks in general. In the future, one needs to investigate strategies to minimise the influence of the imbalanced sample size in the models’ performance, which has particularly affected the presented study. Besides, a larger amount of data should be explored for improving the generalisability and robustness of the proposed method.
7. References


