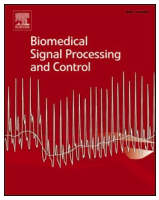
Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Biomedical Signal Processing and Control 72 (2022) 103304

Contents lists available at ScienceDirect

Biomedical Signal Processing and Control

journal homepage: www.elsevier.com/locate/bspc



Automatic cough detection from realistic audio recordings using C-BiLSTM with boundary regression

Mingyu You [a,b,*], Weihao Wang [a], You Li [a], Jiaming Liu [c], Xianghuai Xu [d], Zhongmin Qiu [d]

[a] Department of Control Science and Engineering, Tongji University, Shanghai, China
[b] Frontiers Science Center for Intelligent Autonomous Systems, Shanghai, China
[c] Department of Computer Vision Technology (VIS), Baidu Inc, Beijing, China
[d] Tongji Hospital of Tongji University, Shanghai, China

ARTICLE INFO

Keywords:
Cough detection
Deep learning
BiLSTM
C-BiLSTM
Boundary regression

ABSTRACT

Automatic cough detection in the patients' realistic audio recordings is of great significance to diagnose and monitor respiratory diseases, such as COVID-19. Many detection methods have been developed so far, but they are still unable to meet the practical requirements. In this paper, we present a deep convolutional bidirectional long short-term memory (C-BiLSTM) model with boundary regression for cough detection, where cough and non-cough parts need to be classified and located. We added convolutional layers before the LSTM to enhance the cough features and preserve the temporal information of the audio data. Considering the importance of the cough event integrity for subsequent analysis, the novel model includes an embedded boundary regression on the last feature map for both higher detection accuracy and more accurate boundaries. We delicately designed, collected and labelled a realistic audio dataset containing recordings of patients with respiratory diseases, named the Corp Dataset. 168 h of recordings with 9969 coughs from 42 different patients are included. The dataset is published online on the MARI Lab website (https://mari.tongji.edu.cn/info/1012/1030.htm). The results show that the system achieves a sensitivity of 84.13%, a specificity of 99.82% and an intersection-over-union (IoU) of 0.89, which is significantly superior to other related models. With the proposed method, all the criteria on cough detection significantly increased. The open source Corp Dataset provides useful material and a benchmark for researchers investigating cough detection. We propose the state-of-the-art system with boundary regression, laying the foundation for identifying cough sounds in real-world audio data.

1. Introduction

Cough is an important defensive mechanism of the respiratory system. It has a protective role in clearing the respiratory tract from foreign bodies and secretions and is accompanied by sound. It is regarded as an important symptom of certain diseases, such as croup [1] and COVID-19 [2], and attracts extensive attention from researchers. Cough diagnosis usually depends on the patient's subjective complaint or cough questionnaire assessments, such as the Leicester Cough Questionnaire [3] and the Baseline/Transitional Dyspnea Index [4]. In order to achieve an objective, accurate and reliable cough monitoring, approaches for automatic cough detection from audio data are necessary [5–7].

The most prevalent cough detection system is the Gaussian Mixture Model and Hidden Markov Model (GMM-HMM) based semi-automatic Leicester Cough [8,9]. A more powerful deep neural network (DNN) model has been adopted to replace the GMM part, forming the DNN–HMM model [10–12]. In order to handle sequential information in time series, recurrent neural network (RNN) has enabled cough detectors to achieve a higher accuracy [13–15]. Recent studies have focused on designing multiple acoustic features and achieved a high detection accuracy [16]. However, most of the current detection models are based on data that are recorded in a controlled environment or artificially composed of sound clips, while the real environment carries different challenges [17,18].

Compared with the general audio processing, cough detection has its own challenges. In speech recognition, the same pronunciation clues can be discovered from different people. On the other hand, the cough sound varies a lot due to the different causes, courses of diseases and anatomies. In addition, the sudden occurrence of cough makes it difficult to record it. All these reasons hinder achieving an accuracy similar to that

* Corresponding author at: Department of Control Science and Engineering, Tongji University, Shanghai, China.
E-mail address: myyou@tongji.edu.cn (M. You).

https://doi.org/10.1016/j.bspc.2021.103304
Received 13 June 2021; Received in revised form 8 October 2021; Accepted 23 October 2021
Available online 11 November 2021

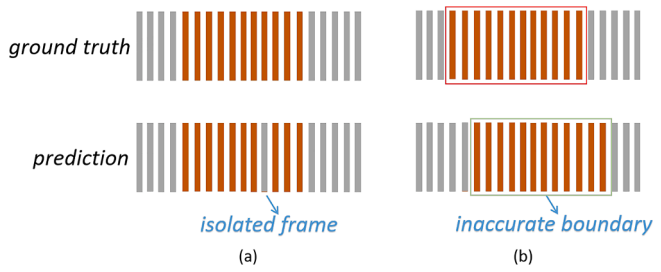


Fig. 1. Orange columns represent the cough frames and gray columns represent other audio frames. Previous works on cough detection may suffer from (a) Isolated non-cough frames. (b) Inaccurate boundaries.

of voice detection in the task of cough detection [19–21].

Previous works focused on the feature extraction or model construction, but they ignored the reasons behind the unsatisfactory cough detection results [22]. While a large number of samples is necessary for deep learning-related methods, there is currently no open access dataset containing enough cough samples, which makes it difficult to train deep models on this task [23]. Similar to the processing of many types of sequential data, cough detection is usually conducted frame by frame. Consecutive frames are merged into a cough event; thus, a misjudgment in the middle of an event yields the detection of an extra boundary (Fig. 1(a)). Accurate cough event boundaries are essential for subsequent analysis. Both the etiology and course analyses require a complete cough event. With precise boundaries, the misjudgment of middle frames can also be avoided. Previous works seldom took the accurate positioning of the boundaries into consideration (Fig. 1(b)).

Taking the above-mentioned observations into account, we collected and labelled a comprehensive dataset named the Corp Dataset, which includes recordings from 42 patients at Tongji hospital, China. With the extracted acoustic features, we designed a detection model combining a convolutional neural network (CNN) and a deep bidirectional long short-term memory (BiLSTM). A joint generator of a score classifier and a boundary regressor was used to get the confidence and boundaries of the cough events on the feature map of the last layer. As a result, the boundary regressor locates the cough boundaries based on the cough frames predicted by the score classifier. In summary, our main contributions can be listed as follows:

1) We present the publicly available Corp Dataset, which contains the recordings of 9969 coughs over 168 h from 42 real patients of both genders affected by respiratory diseases.
2) We propose a deep C-BiLSTM network, combining a CNN and a deep bidirectional LSTM. The CNN analyzes the audio spectrum, while the BiLSTM predicts the current frame by referencing the adjacent frames.
3) We implement bounding boxes on the feature map of the audio data to accurately locate cough events in the detection results, which is presented for the first time.
4) We evaluate the C-BiLSTM network with the boundary regression method using the Corp Dataset. Competitive results are obtained compared with the state-of-the-art cough detection models.

2. Materials and methods

2.1. Corp dataset

In this section, we introduce the properties of the Corp Dataset, a carefully labelled cough audio dataset. The construction of the dataset took five years with the following 5 stages, and still continues: i) Dataset design; ii) Ethical approval application; iii) Volunteer recruitment; iv) Audio acquisition; v) Label annotation.

2.1.1. Dataset collection

The Corp Dataset is the first large-scale audio dataset that includes real-life audio recordings containing lots of coughs from volunteer patients in their living environment. It currently includes 42 volunteer inpatients in Tongji hospital with diverse respiratory diseases including community acquired pneumonia (CAP, n = 18), bronchial asthma (BA, n = 4), chronic obstructive pulmonary disease (COPD, n = 17) and unknown diseases (n = 3). All the volunteers signed informed consents before collecting the data. Audio segments including a patient's private content were removed and only conversation fragments without private content were retained.

The recording system is composed of a SONY ICD-LX30 portable digital recorder and an ECM-CS10 microphone. The audio data are recorded at a sampling frequency of 44.1 kHz and a bit rate of 192kbps. The microphone is attached to the patient's collar and the recorder is kept in their pockets. Recording the sounds of each patient took 48 h or longer. We finally keep all the cough segments and remove private or

Table 1
Information of patients.

Corp I					Corp II				
ID	Age	Disease	Cough Events	Duration (min)	ID	Age	Disease	Cough Events	Duration (min)
1	47	CAP	139	226.2	21	69	COPD	120	236
2	48	CAP	88	160	22	76	COPD	150	360
3	49	CAP	251	216	23	84	COPD	220	157
4	51	CAP	121	78.6	24	61	CAP	383	400
5	52	CAP	111	167	25	76	CAP	167	226.2
6	53	COPD	154	186.8	26	50	COPD	154	206.5
7	54	CAP	21	167.2	27	89	COPD	556	170
8	55	CAP	112	190	28	66	COPD	64	157.3
9	56	CAP	72	186.8	29	70	COPD	29	190
10	56	COPD	58	180.6	30	72	CAP	41	108
11	59.5	CAP	274	127.8	31	67	COPD	19	285.1
12	61	COPD	55	150	32	59	CAP	568	290
13	66	COPD	7	400	33	54	CAP	709	400
14	69	CAP	385	220	34	63	BA	418	800
15	72	COPD	22	168.5	35	42	CAP	295	353.5
16	76	BA	68	186.8	36	69	CAP	265	400
17	79	BA	195	231.5	37	64	BA	834	360
18	80	COPD	452	186.8	38	54	CAP	265	300
19	66	COPD	179	147.5	39	71	COPD	656	400
20	90	COPD	134	157.3	40	–	–	35	118
					41	–	–	889	350
					42	–	–	234	77

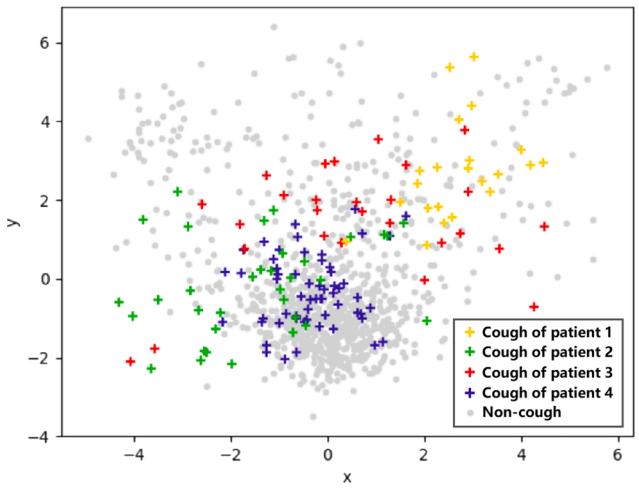


Fig. 2. The compressed 2D MFCC feature points are from the coughs of four patients (colored points, different colors for different patients) and background noise (gray points).

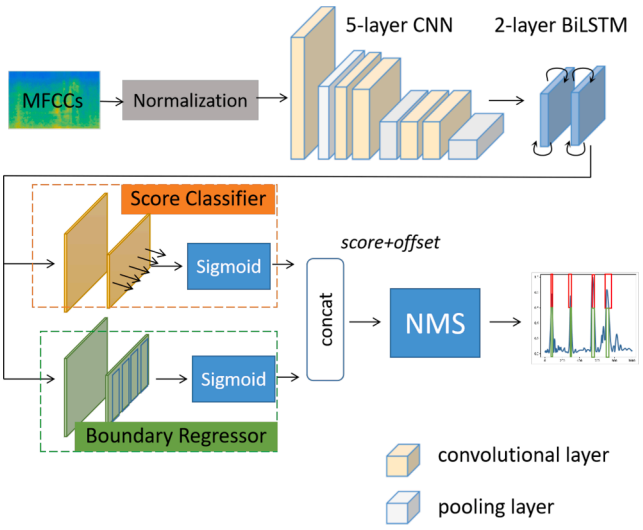


Fig. 3. The complete framework of C-BiLSTM. It contains a normalization layer, 5 convolutional layers and 2 BiLSTM layers. The results are generated by the score classifier and boundary regressor. The non-maximum suppressor (NMS) is only used in the testing process.

blank parts. In Table 1, we show the remaining audio duration and the number of coughs recorded from each patient.

During the five dataset construction years, three batches of volunteer patients were respectively recruited: 20 patients, 11 patients and another 11 patients. All the patients had respiratory diseases. The disease distribution is shown in Table 1. Due to the different labeling rules, the data of the first 20 patients were organized as Corp Dataset I and those of the remaining 22 patients as Corp Dataset II. Most of these patients had the disease of CAP (18/42 patients), while the number of BA cases was the lowest, with only 4 cases. Correspondingly, there were 4,267 coughs that belonged to the patients with CAP, accounting for about 50% of the total number of recorded coughs. Finally, there were 3029 coughs that belonged to the cases of COPD and 1515 ones belonging to the cases of BA, accounting for 32% and 16% of the total, respectively. And the rest 2% remain unknown disease.

2.1.2. Dataset labeling

In order to simplify the labeling process, the data were cut into 10-min segments. Segments shorter than ten minutes were discarded. Labeling was done using PRAAT [24]. PRAAT is a free computer software package for speech analysis. Segmentation and labelling can be completed using TextGrid, which is a type of objects in PRAAT used for annotation. Both the start and end positions of every cough event were carefully marked while the remaining part was regarded as a non-cough part. When the patient continued to cough, we separately labeled the boundaries of each cough. Every 10-min audio segment file had a corresponding label file exported from PRAAT. Considering the professional requirements, nine experts were employed from the respiratory clinicians. Each audio segment was labeled by two experts. Audio segments with the same labels retained the average boundaries as the final label. The controversial parts were left for another expert to cast a vote.

For the Corp Dataset I, the used labels were “c” for cough and “n” for others. As more labeling experts joined in the Corp Dataset II, refined labels were adopted. Confident cough was labeled as “1”, and unconfident but so much alike one was labeled as “0”. A possible cough may be the sound between a clear throat and a cough, or a short cough that is difficult to be distinguished by human ears. If the cough sounds a little away from the microphone, it may be from other patients and is labeled as “2”. Precise labels prepare for a detailed analysis. The Corp Dataset I contained 2,898 coughs in 62.3 h of recordings, while the Corp Dataset II contained 7,071 coughs in 105.7 h. Besides, there were 301 possible coughs and 4,946 coughs from other patients in the Corp Dataset II. In general, the frequency of the cough events was 59/h.

2.1.3. Dataset analysis and access

In order to record real-life cough sounds, we did not restrict the patients’ normal activities during the collection process. There was a lot of noise in the recordings. Some noise sounds overlaid the cough sounds while others sounded like cough (such as sneezing, throat clearing, snoring, laughing, etc.). It was impossible to precisely calculate the SNR (signal-to-noise ratio) of the coughs due to the difficulty of separating the coughs from the background sounds. We just estimated the SNR of the cough sounds, which ranged from −10 dB to 10 dB. By examining all the labels, we found that the coughs were about 0.1 to 0.3 s. These short coughs from different people have different acoustic features. The cough frequency and sound differ between different diseases and between men and women. In order to visualize the differences among coughs from different patients and the distance between the cough and background sounds through visualization, we extracted the MFCC (Mel Frequency Cepstral Coefficient) features as described in Section 2.2.1 of each frame and compressed them to two dimensions using Principal Component Analysis (PCA) method. Fig. 2 shows the plotted 2D cough features from four patients and background noise. By observing the distribution of these points, the following observations can be made:

1) Although cough points from the same patient are relatively close, they scatter in space. This makes it difficult to learn the commonalities among different coughs.
2) Feature points from background and cough sounds are intertwined, making it difficult to distinguish the background sounds from the cough ones. In real environment, a large number of complex background sounds bring difficulties to the detection of cough, which has a great interference effect.

In light of the above-mentioned considerations, the cough sound does not have a strong regularity and is similar to the background sounds. In real environments, strong models are expected to accurately distinguish the difference between cough and background sounds.

Samples of the Corp Dataset are published on the homepage of the MARI Lab.[1] Limited by the storage space of the homepage, we will provide download links replying to any requests. The full dataset contains two folders named as the Corp Dataset I and Corp Dataset II.

[1] https://mari.tongji.edu.cn/info/1012/1030.htm.

Table 2
Convolutional structure settings.

1	conv 64@7 × 1, padding = 3, stride = 1
2	conv 64@7 × 1, padding = 3, stride = 1
3	conv 128@3 × 1, padding = 1, stride = 1
4	conv 128@3 × 1, padding = 1, stride = 1
5	conv 128@3 × 1, padding = 1, stride = 1

Researchers who benefit from the Corp Dataset are kindly asked to cite this paper.

2.2. Approach

As shown in Fig. 3, the extracted frame-level audio features of a 10-s segment are transformed into a spectrogram, which is normalized and fed into the deep network. The detection network finally outputs the probability of each frame to belong to a cough event by the end-to-end score classifier. Boundary regression, as the other branch of the network, predicts cough boundaries for each frame on the feature map of the last layer of the C-BiLSTM network.

2.2.1. C-BiLSTM network

Acoustic features. The features of the framed audios are extracted using the MFCC [25] method, which is widely used in automatic speech and audio processing. The MFCC designs a set of filters from low to high frequency and from dense to sparse signals, which is more in line with the auditory characteristics of the human ear. Based on the MFCCs, we further concatenate the first and second temporal derivatives to incorporate the dynamic characteristics into the extracted features. For the convergence of the neural network, we further normalize the feature vector to a standard normal distribution with a mean value of 0 and a variance of 1.

Convolutional structure. CNNs are used to extract high-level features of the time–frequency maps of audios. Following the structure of AlexNet [26], we design the structure of CNN with five convolutional layers [27] as shown in Table 2. Since the MFCCs are uncorrelated features, we split the cepstral parameters into different channels and use several 1-D convolutional kernels. The first layer of the CNN "64@7 × 1" has 64 convolutional kernels, each with the size of 7×1, 7 on the temporal axis and 1 on the feature axis. In each convolutional layer, we use padding along the temporal axis and set the stride to 1 to keep the length of the feature map unchanged.

Deep bidirectional long short-term memory. RNN is the most prominent network used in acoustic problems. It maintains the historical information by combining the output of the previous moment with the input of the current moment and feeding them into a loop unit. Based on the structure of RNN, LSTM attempts to solve the gradient vanishing and gradient explosion problems [28]. The LSTM unit is composed of a memory cell and three gates, called the input, forget and output gates. At time step t, the LSTM unit accepts the input x_t and the hidden unit outputs from previous time step h_{t-1}. With the information from its memory cell, the LSTM unit updates the states of the three gates and its memory unit. Then, it outputs the hidden output h_t through the activation function S and the gate result o_t. In order to seek a stronger ability, we double the layers of the LSTM network with 100 nodes in each layer and use the bidirectional structure of BiLSTM to respectively judge the current frame in the forward and backward directions.

2.2.2. Boundary regression on feature map

Proposals. In the feature map from the last layer of BiLSTM, three values ($score, \Delta y_1$ and Δy_2) are calculated for each frame by the two network branches. These three values make up a proposal for a frame that helps to locate the cough event to which this frame belongs. In the first branch, two fully connected layers and a sigmoid activation function form the score classifier. It outputs the value of *score* which ranges from 0 to 1 and indicates the possibility of the frame to be a cough frame. In the other branch, another set of two fully connected layers and a sigmoid activation function form the boundary regressor. It outputs the values of Δy_1 and Δy_2 when the frame is predicted to be a cough frame. The two values of Δy_1 and Δy_2 stand for the two boundary offsets from the start frame to the current frame and from the current frame to the end frame in the same cough event, respectively. Let $b_{left,t}$ and $b_{right,t}$ represent the left and right boundary of a cough event, respectively, then $\Delta y_{1,t}$ and $\Delta y_{2,t}$ can be calculated from the position t of the current frame and the cough boundaries, as depicted in (1). A large variance of the relative positions of the coughs is troublesome for the boundary regressor training. Therefore, normalized $\sigma(\Delta y_{1,t})$ and $\sigma(\Delta y_{2,t})$ can be helpful, as depicted in (2).

$$\begin{cases} \Delta y_{1,t} = t - b_{left,t} & \text{(1a)} \\ \Delta y_{2,t} = b_{right,t} - t & \text{(1b)} \end{cases} \tag{1}$$

$$\begin{cases} \sigma(\Delta y_{1,t}) = \Delta y_{1,t}/(b_{right,t} - b_{left,t}) & \text{(2a)} \\ \sigma(\Delta y_{2,t}) = \Delta y_{2,t}/(b_{right,t} - b_{left,t}) & \text{(2b)} \end{cases} \tag{2}$$

Network training. For network training, the loss function L is designed as depicted in line 5 of Algorithm 1. It is a weighted average of the score loss (line 3 in Algorithm 1) and boundary loss (line 4 in Algorithm 1). The ground truth values score, left boundary and right boundary of a frame at time t are denoted by $s_t, gtb_{left,t}$ and $gtb_{right,t}$, respectively. When the frame is in a cough segment, s_t is equal to 1 and $gtb_{left,t}$ and $gtb_{right,t}$ are concerned, otherwise s_t is equal to 0 and $gtb_{left,t}$ and $gtb_{right,t}$ are ignored. We use the cross-entropy loss function (*CEH*) in (3) and Smooth L1 loss function in (4). η is a hyperparameter used to balance the losses of the classifier and regressor. The total loss will be updated as shown in Algorithm 1. In the practical training process, the model will be pre-trained by minimizing the weighted total loss with a balance hyperparameter η 0.8 (L is designed as line 5 in Algorithm 1). Then, the parameters of the two fully connected layers in the score classifier and boundary regressor are alternately refined, updating one by fixing the other until the best result is achieved on the validation set.

$$CEH(p,q) = -[plogq + (1-p)log(1-q)] \tag{3}$$

$$SmoothL1(p,q) = \begin{cases} 0.5(p-q)^2, & if\ |p-q| < 1 \\ |p-q| - 0.5, & otherwise \end{cases} \tag{4}$$

Non-maximum suppression in testing process. In the testing stage, every frame gets a cough event proposal. All these proposals constitute the set *EP*. Ideally, the proposals for the frames in the same cough event exactly coincide. However, there still exists a calculation error. We filter these proposals by using the non-maximum suppression (NMS) algorithm. The proposals are firstly sorted in descending order based on the *score*, and the one with the highest *score* is selected, denoted as Proposal *M*. Proposal *M* and the proposals overlapping Proposal *M* are removed from the *EP* set. We cyclically sort the remaining proposals in the *EP* set, select the one with the highest score and delete the overlapping proposals, until *EP* is empty.

Algorithm 1. TRAINING INTERATION PROCEDURE

Require: B is the batch size, β is a batch of data with size B, and ε donates each sample in a batch. L_{score} is score loss, $L_{boundary}$ is boundary loss, η is a hyperparameter, θ denotes the model parameters, $\widetilde{\theta}$ denotes the updated model parameters. α is the learning rate.

1: $\beta = RandomSample(\{\varepsilon_1, \cdots, \varepsilon_N\}, B)$
2: **for** $\varepsilon^j \in \beta$
3: $L_{score} += \sum_{t=0}^{j} CEH(score_t, s_t)$
4: $L_{boundary} += \sum_{t=0}^{j} (SmoothL1(2 * \sigma(\Delta y_{1,t}), 2 * \sigma(t - gtb_{left,t})) + SmoothL1(2 * \sigma(\Delta y_{2,t}), 2 * \sigma(gtb_{right,t} - t)))* s_t$
5: $L += \eta * L_{score} + (1-\eta) * L_{boundary}$
6: **end for**
7: $\widetilde{\theta} = \theta - \alpha \nabla_\theta L$
8: **return** $\widetilde{\theta}$

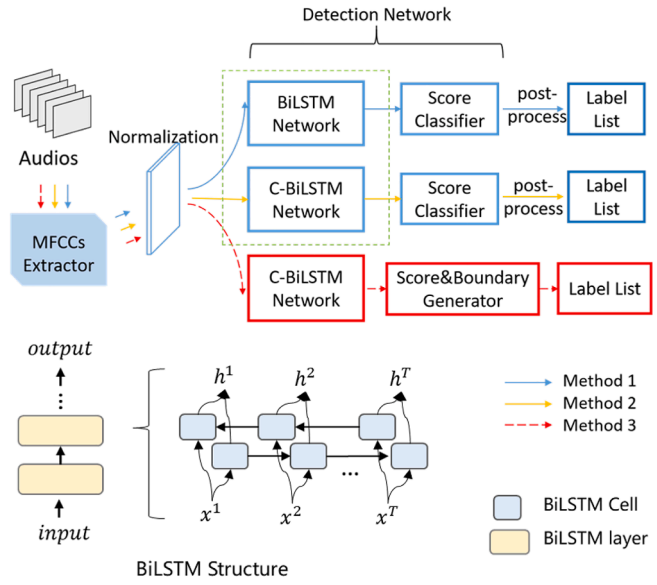


Fig. 4. Three network structures are designed. The red part is our C-BiLSTM network with a score classifier and a boundary regressor. Detailed schematic diagram of BiLSTM is demonstrated below.

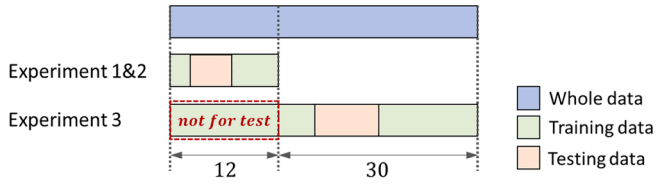


Fig. 5. Demonstration of data split of training data and testing data. Note that the data used in experiment 1 and 2 cannot be used as testing data in experiment 3. Each experiment is run for 3 times.

3. Results

As shown in Fig. 4, we design three network structures for the model upgrade and three experiments are performed accordingly to search for the optimal settings. In experiment 1, we test the performance of the BiLSTM network with different layers (Method 1). In experiment 2, we search for the most optimal C-BiLSTM network structure in different CNN layers (Method 2). In order to solve the wrong break in Fig. 1(a), the label of the isolated cough/non-cough frame is fixed in post-processing. In experiment 3, based on the selected C-BiLSTM structure in experiment 2, we evaluate the function of boundary regression (Method 3) and compare the three methods with related work [9,13,17].

3.1. Experiment setup

In pre-processing, the audio segment is downsampled to 16 kHz. With 25 ms Hamming windows and 15 ms overlap, every 10-s audio segment is divided into 998 frames. Then, 13-dimensional MFCC features and the corresponding first and second temporal derivatives are calculated.

The data are divided into testing data and training data as shown in Fig. 5. In experiment 1 and 2, the data of 12 patients are separated from the rest. The experiment is run for 3 times with non-overlapped testing data of 4 patients, which is evenly divided from the 12 patients, and the rest for training. In experiment 3, total 42 patients are involved. Similarly, this experiment is run for 3 times with non-overlapped testing data of 10 patients. Worth noting that, in order to prevent information leakage, the testing data is obtained from the remaining 30 patients, w.r.t. the patients not involved in experiment 1 and 2. Finally, the averaged test results of three runs are reported for each experiment.

Table 3
Results of BiLSTM with different layers. BiLSTM with 5 layers and 7 layers both obtain the best results, and the 5-layer BiLSTM is chosen with the consideration of the balance between performance and computational complexity.

Layers	SENS (%)	SPEC (%)	ACC (%)	MCC (%)	PPV (%)	NPV (%)
1	69.11	99.65	99.27	69.42	70.48	99.62
2	71.72	99.62	99.28	70.39	69.8	99.65
3	74.33	99.64	99.34	72.65	71.66	**99.69**
4	74.22	99.69	99.39	74.34	75.08	99.68
5	**75.28**	99.7	**99.41**	75.12	75.56	**99.69**
6	74.94	99.69	99.39	74.43	74.54	**99.69**
7	75.05	**99.71**	**99.41**	**75.29**	**76.15**	**99.69**

The network is trained using the Adam optimization algorithm with an initial learning rate of 0.001 and a batch size of 128. The model quickly converges in the training phase. In order to accelerate the training process, we use 4 NVIDIA Tesla K20 GPUs, each containing 5 GB of GDDR5 RAM and 2496 processing cores.

3.2. Performance metrics

Before stating the performance metrics, some calculation indexes are defined. Every detected cough event is counted as a true positive (*TP*) if its middle frame is contained in a ground truth positive, otherwise it is counted as a false positive (*FP*). The false negatives (*FN*) count the ground truth positives that are not detected, while the true negatives (*TN*) are non-cough events that are correctly not detected as coughs. The Corp Dataset is collected in uncontrolled environments; thus, it contains different noise types. Non-cough events are difficult to be defined, so we only label the cough events. In the calculation of *TN*, segments with twice the average length of the cough are counted as non-cough events. After calculating TP, FP, TN and FN, the values of the sensitivity (*SENS*), specificity (*SPEC*), accuracy (*ACC*), Matthews Correlation Coefficient (*MCC*), precision (*PPV*), and negative predictive value (*NPV*) are stated based on the following definitions in (5a)–(5f):

$$\begin{cases} SENS = TP/(TP+FN) & \text{(5a)} \\ SPEC = TN/(FP+TN) & \text{(5b)} \\ ACC = (TP+TN)/(TP+TN+FP+FN) & \text{(5c)} \\ MCC = \dfrac{TP^{*}\ TN - FP^{*}\ FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} & \text{(5d)} \\ PPV = TP/(TP+FP) & \text{(5e)} \\ NPV = TN/(TN+FN) & \text{(5f)} \end{cases} \tag{5}$$

MCC is a metric of the whole performance of a classification process. Particularly, *MCC* is equivalent to *ACC* when the classes are unbalanced. *MCC* is also a commonly employed evaluation metric in cough detection as in [9,13].

Since the value of *TP* only considers the middle frame of the detected events, this results in ignoring the boundary accuracy of the detection ignored. For subsequent analysis, it is worth measuring the accuracy of the boundaries. Here we introduce *IoU* (intersection-over-union), a concept from the traditional object detection domain. It is concerned with the degree of overlap between the prediction box and the ground truth box. *IoU* is equal to the intersection of these two boxes divided by their union. When these two boxes greatly deviate, the value of *IoU* will inevitably approach to zero. On the other hand, if the detected cough segment coincides with the ground truth, the value of *IoU* reaches 1. The final *IoU* is the average of all the *IoU* values of the correctly detected coughs.

Table 4
Results of C-BiLSTM with different convolutional layers. C-BiLSTM network with 5 convolutional layers achieves the highest result on all metrics. Since the promotion on performance is not significant when convolutional layers varies from 3 to 5, the number of convolutional layers is set to 5.

Layers	SENS (%)	SPEC (%)	ACC (%)	MCC (%)	PPV (%)	NPV (%)
1	72.16	99.58	99.25	69.64	67.94	99.66
3	77.65	99.64	99.37	74.76	72.59	99.72
5	**78.38**	**99.71**	**99.45**	**77.4**	**76.99**	**99.73**

3.3. Results and analysis

3.3.1. Optimal structure of C-BiLSTM

In this experiment, the separated data of 12 patients were used. Table 3 lists the detection results of BiLSTM from layer 1 to layer 7, each with 100 hidden nodes. The network achieves better results with more layers, with the best results achieved by BiLSTMs with 5 layers and 7 layers. The *SENS* and *ACC* of BiLSTM with 5 layers are the highest, and the 7-layer BiLSTM has the highest $SPEC, ACC, MCC$ and *PPV* values. Considering the balance between the performance and computational complexity, we used the 5-layer BiLSTM in the following experiments.

For a powerful feature expression, we combined CNN with BiLSTM to form a C-BiLSTM network, which is composed of several convolutional layers and two BiLSTM layers. As shown in Table 2, the number of the CNN layers was carefully selected as 1, 3, and 5. From Table 4, it can be observed that the network containing 5 convolutional layers achieves the highest result on all the metrics. Since the difference between the results of the 3-layer and 5-layer structures is not significant, we set the CNN in C-BiLSTM to have 5 layers.

3.3.2. Comparison with prior works

We evaluated our proposed Methods 1–3, shown in Fig. 4 by comparing them with the cough detection methods presented in [9,13,17]. Since the models are not publicly accessible, we implemented these methods ourselves on the Corp Dataset according to the descriptions in the corresponding articles. In order to build the model in [9], we downsampled the audios from 44.1 kHz to 16 kHz and extracted 39 features for each frame as the window settings in our experiment. Then, we trained HMM models and set Gaussian functions for each model as outlined in [9]. In [13], the best performance is achieved by *k*-nearest neighbor (*k*-NN) classification with Hu moments. We reserved the hyper-parameters in [13] to get 13-dimensional Hu moments for each frame, and trained a 1-NN model with standardized Euclidean distance. The detected coughs and non-coughs were sorted into a sequence in the post-processing. As described in [17], the 44.1 kHz audio samples we collected were downsampled to 22.05 kHz, and 1024-dimensional Mel-scaled spectrograms were extracted for each frame. We built the proposed 5-layer CNN in [17]. The labels of the isolated frames in the middle of the sequence were all revised.

All the methods were evaluated using the metrics described in Section 3.2. From Table 5, we can observe that: 1) Deep networks are stronger than the traditional machine learning models in cough detection. The *SENS* values of the HMM and *k*-NN models are lower than 70%, while those of the deep learning models are higher than 74%; 2) The results of HMM are a little better than those of *k*-NN. HMM was suitable for temporal tasks such as speech recognition before the rise of deep learning, and we once again witness its ability here. *k*-NN is not bad, as its performance on *PPV* and *IoU* is still competitive, and it is a simple method; 3) All the *SENS* values are absolutely lower than the *SPEC* values. Imbalanced data distribution is one of the reasons. Some non-coughs that are mistakenly classified as coughs will decrease the *SPEC* values a little bit. At the same time, as the number of coughs is limited, any missed cough could be discovered in the *SENS* value. Besides, due to the similarity between coughs and noise, the possibility of a cough being missed is indeed high. It makes the cough detection on the real dataset of Corp more challenging; 4) CNN and BiLSTM achieve a comparable performance, although CNN can sometimes be better. Cough detection is a temporal task, at which LSTM is good. However, beyond our expectations, LSTM cannot outperform CNN. The temporal information in the spectrum must be helping CNN a lot; 5) With the help of CNN and BiLSTM, C-BiLSTM stands at a higher point. It gets better results than all the previous structures on all the evaluation metrics; 6) The seven evaluation metrics indicate almost the same rankings. *SENS* totally agrees with *MCC*, and *SPEC* totally agrees with *PPV*. On all these metrics, C-BiLSTM with boundary regression achieves the best result. The superiority of the model design has been confirmed. It significantly improves *IoU*, verifying the effect of the boundary regressor. More accurate boundary locations ensure the subsequent cough analysis. The method of C-BiLSTM with boundary regression has a *SENS* value of more than 84%, which is close to the application requirements. It's worth looking forward to the clinical trials.

4. Discussion and conclusion

In previous works on cough detection, the authors collected materials in controlled environment or used synthesized audios to simulate real-life scenarios, which were somehow different from the real data. Based on different datasets, they designed models and reported their results. Because the results are based on different evaluation data, the comparison of these results seems meaningless. Without enough and suitable data, the task of cough detection cannot flourish like the image processing, and it is difficult for more researchers to join this field. The Corp Dataset is a novel attempt to make cough audio data publicly accessible, hoping to promote the development of this topic. Cough is a popular health problem, and we would like to do something to support the medical development.

Based on the Corp Dataset, we proposed a C-BiLSTM cough detection method with boundary regression and obtained better results compared with the state-of-the-art methods. 84.13% of the true coughs were correctly detected and the remaining 15.87% were still missed. At the same time, 84.06% of the detected coughs were true positives and 15.94% of them were mistakenly detected. These two metrics are contradictory to a certain extent. When the model is tuned to be very sensitive to cough sounds, the number of correctly detected coughs will

Table 5
The Mean Testing Detection Results of GMM+HMM with MFCCs, *k*-NN with Hu Moments, 5-layer CNN with Mel-scaled Spectrograms, BiLSTM, C-BiLSTM and C-BiLSTM with Boundary Regression on Corp Dataset over 3 Runs.

Class.	SENS(%)	SPEC(%)	ACC(%)	MCC(%)	PPV(%)	NPV(%)	IoU
Prior Works							
GMM+HMM with MFCCs [9]	66.12	99.66	99.28	67.48	69.63	99.6	0.66
k-NN with Hu Moments [13]	63.89	99.72	99.31	67.87	72.92	99.58	0.72
5-layer CNN with Mel-scaled Spectrograms [17]	78.54	99.78	99.53	79.25	80.47	99.75	0.79
Our Works							
Deep BiLSTM (Method 1)	74.83	99.76	99.48	76.51	78.84	99.71	0.73
C-BiLSTM (Method 2)	82.76	99.77	99.57	81.53	80.74	99.8	0.75
C-BiLSTM with boundary regression (Method 3)	**84.13**	**99.82**	**99.64**	**83.91**	**84.06**	**99.81**	**0.89**

increase, but the number of falsely detected coughs will also increase. 84.13% and 84.06% are an acceptable trade-off. The model has a very high *SPEC* value, reaching 99.82%. It means that a large number of non-cough events are correctly distinguished. Facing lots of unknown non-cough events, the model shows a strong generalization ability.Further clinical analysis of the detected coughs can help the specialists to diagnose and monitor the patient's condition. By recording the patient's real-life sounds, coughs can be automatically detected. We have adopted the model to an online service API, and users can upload the files to the detection system and get the detection results.

CRediT authorship contribution statement

Mingyu You: Conceptualization, Methodology, Data curation, Validation, Formal analysis. **Weihao Wang:** Methodology, Validation, Visualization. **You Li:** Methodology, Validation, Visualization. **Jiaming Liu:** Investigation. **Xianghuai Xu:** Data curation. **Zhongmin Qiu:** Data curation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

The protocol was approved by the Ethics Committee of Tongji Hospital (No. KYSB-2016-003) and registered in the Chinese Clinical Trials Register (ChiCTR-DDD-17012587) (http://www.chictr.org.cn/). This research has been supported by the National Natural Science Foundation of China under Grant No. 62073244, the Shanghai Innovation Action Plan under Grant No. 20511100500 and the Fundamental Research Funds for the Central Universities (22120190205).

References

[1] R.V. Sharan, U.R. Abeyratne, V.R. Swarnkar, P. Porter, Automatic croup diagnosis using cough sound recognition, IEEE Transactions on Biomedical Engineering 66 (2) (2019) 485–495.
[2] Z. Xu, L. Shi, Y. Wang, J. Zhang, L. Huang, C. Zhang, S. Liu, P. Zhao, H. Liu, L. Zhu, et al., Pathological findings of covid-19 associated with acute respiratory distress syndrome, The Lancet Respiratory Medicine 8 (4) (2020) 420–422.
[3] S. Birring, T. Fleming, S. Matos, A. Raj, D. Evans, I. Pavord, The leicester cough monitor: preliminary validation of an automated cough detection system in chronic cough, European Respiratory Journal 31 (5) (2008) 1013–1018.
[4] C.J. Ryerson, D. Donesky, S.Z. Pantilat, H.R. Collard, Dyspnea in idiopathic pulmonary fibrosis: a systematic review, Journal of Pain and Symptom Management 43 (4) (2012) 771–782.
[5] S.J. Barry, A.D. Dane, A.H. Morice, A.D. Walmsley, The automatic recognition and counting of cough, Cough 2 (1) (2006) 8.
[6] E.C. Larson, T. Lee, S. Liu, M. Rosenfeld, S.N. Patel, Accurate and privacy preserving cough sensing using a low-cost microphone, in: Proceedings of the 13th International Conference on Ubiquitous Computing, ACM, 2011, pp. 375–384.
[7] A. Windmon, M. Minakshi, P. Bharti, S. Chellappan, M. Johansson, B.A. Jenkins, P. R. Athilingam, Tussiswatch: A smart-phone system to identify cough episodes as early symptoms of chronic obstructive pulmonary disease and congestive heart failure, IEEE Journal of Biomedical and Health Informatics 23 (4) (2018) 1566–1573.
[8] S. Matos, S.S. Birring, I.D. Pavord, D.H. Evans, An automated system for 24-h monitoring of cough frequency: the leicester cough monitor, IEEE Transactions on Biomedical Engineering 54 (8) (2007) 1472–1479.
[9] S. Matos, S.S. Birring, I.D. Pavord, H. Evans, Detection of cough signals in continuous audio recordings using hidden markov models, IEEE Transactions on Biomedical Engineering 53 (6) (2006) 1078–1083.
[10] J.-M. Liu, M. You, Z. Wang, G.-Z. Li, X. Xu, Z. Qiu, Cough detection using deep neural networks, in: 2014 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), IEEE, 2014, pp. 560–563.
[11] H.-H. Wang, J.-M. Liu, M. You, G.-Z. Li, Audio signals encoding for cough classification using convolutional neural networks: A comparative study, in: 2015 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), IEEE, 2015, pp. 442–445.
[12] J.-M. Liu, M. You, Z. Wang, G.-Z. Li, X. Xu, Z. Qiu, Cough event classification by pretrained deep neural network, BMC Medical Informatics and Decision Making 15 (4) (2015) S2.
[13] J. Monge-Álvarez, C. Hoyos-Barceló, P. Lesso, P. Casaseca-de-la Higuera, Robust detection of audio-cough events using local hu moments, IEEE Journal of Biomedical and Health Informatics 23 (1) (2018) 184–196.
[14] J. Monge-Álvarez, C. Hoyos-Barceló, L.M. San-José-Revuelta, P. Casaseca-de-la Higuera, A machine hearing system for robust cough detection based on a high-level representation of band-specific audio features, IEEE Transactions on Biomedical Engineering 66 (8) (2018) 2319–2330.
[15] A. Teyhouee, N.D. Osgood, Cough detection using hidden markov models, in: International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation, Springer, 2019, pp. 266–276.
[16] P. Mouawad, T. Dubnov, S. Dubnov, Robust detection of covid-19 in cough sounds, SN Computer Science 2 (1) (2021) 1–13.
[17] F. Barata, K. Kipfer, M. Weber, P. Tinschert, E. Fleisch, T. Kowatsch, Towards device-agnostic mobile cough detection with convolutional neural networks, in: 2019 IEEE International Conference on Healthcare Informatics (ICHI), 2019, pp. 1–11.
[18] N. Simou, N. Stefanakis, P. Zervas, A universal system for cough detection in domestic acoustic environments, in: 2020 28th European Signal Processing Conference (EUSIPCO), 2021, pp. 111–115, https://doi.org/10.23919/Eusipco47968.2020.9287659.
[19] C. Parkinson, J. Woodall, Automatic speech recognition (asr) feedback for head mounted displays (hmd), uS Patent App. 14/540,943 (May 21 2015).
[20] O. Ghahabi, W. Zhou, V. Fischer, A robust voice activity detection for real-time automatic speech recognition, Proc of ESSV.
[21] M.R. Price, J.R. Glass, A.P. Chandrakasan, Low-power automatic speech recognition device, uS Patent App. 16/099,589 (May 16 2019).
[22] E. Nemati, K. Vatanparvar, V. Nathan, T. Ahmed, Listen2cough: Leveraging end-to-end deep learning cough detection model to enhance lung health assessment using passively sensed audio.
[23] O.M. Parkhi, A. Vedaldi, A. Zisserman, et al., Deep face recognition., in: bmvc, vol. 1, 2015, p. 6.
[24] P. Boersma, et al., Praat, a system for doing phonetics by computer, Glot International 5.
[25] M. Xu, L.-Y. Duan, J. Cai, L.-T. Chia, C. Xu, Q. Tian, Hmm-based audio keyword generation, in: Pacific-Rim Conference on Multimedia, Springer, 2004, pp. 566–574.
[26] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.
[27] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Learning deep features for discriminative localization, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2921–2929.
[28] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Computation 9 (1997) 1735–1780, https://doi.org/10.1162/neco.1997.9.8.1735.