Exploring Self-Supervised Representation Ensembles for COVID-19 Cough Classification

Speaker: Zi-Xin, Chen

Advisor: Jia-Ling, Koh

Date: 2022/02/22

Source: KDD '21

Introduction

Goal

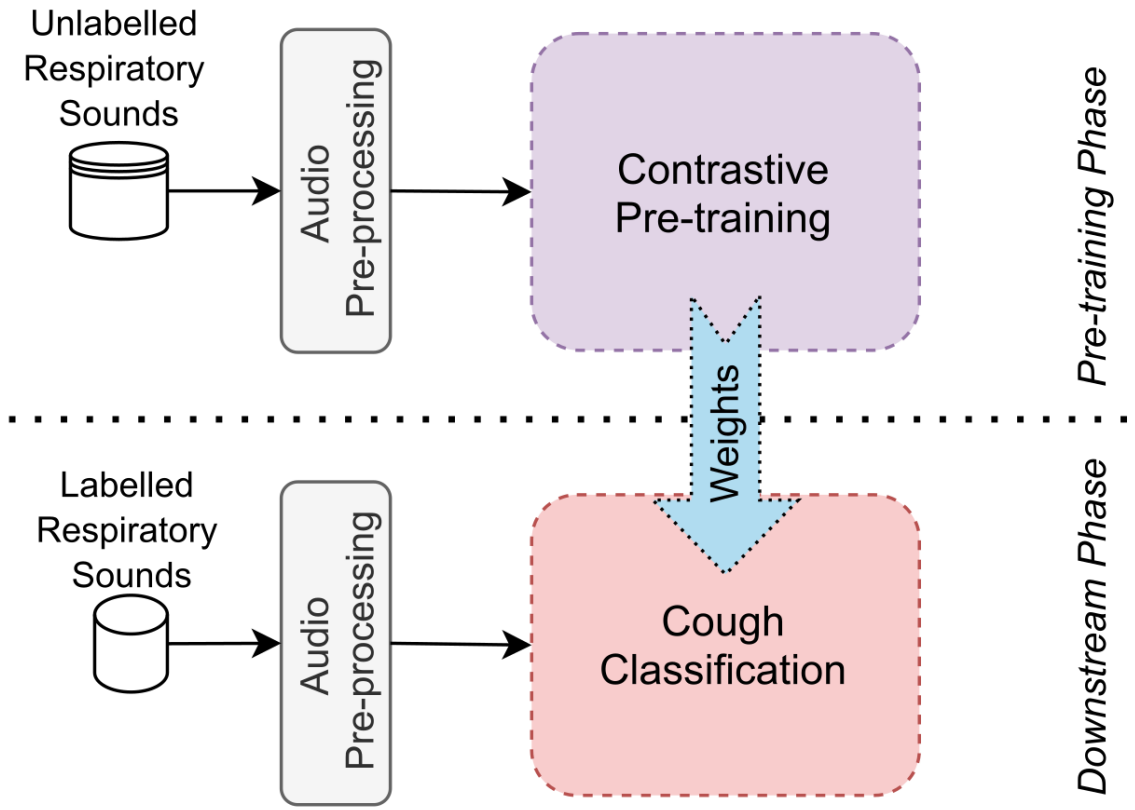
Propose a novel **self-supervised learning** enabled framework for COVID-19 cough classification.

Motivation

1. It is significant to develop a reliable, easily-accessible, and contactless approach for preliminary diagnosis of COVID-19.
2. Existing sound-based diagnostic approaches are trained in a **fully-supervised** manner, which requires large scale well-labeleed data.

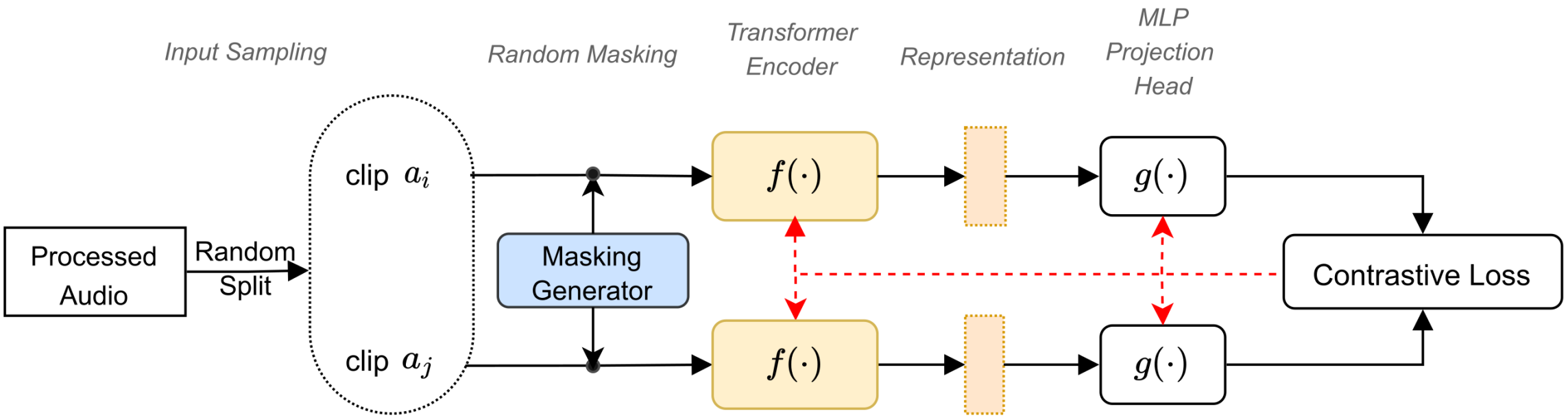
Framework

- **Input:** respiratory audio recordings
- **Output:** probability that the audio is COVID-19 possitive



Method

Contrastive Learning



Pre-processing and Sampling

- Method: **log-compressed mel-filterbanks**
- each raw audio file is mapped to a feature
  - N: the number of frequency bins
  - T: the total number of time frames

$$a \in \mathbb{R}^{N \times T}$$

- with sliding window

$$a \in \mathbb{R}^{N \times T_w}$$

Feature Encoder

embed each clip $a_i \in \mathbb{R}^{N \times T_w}$ into a representation vector $h_i \in \mathbb{R}^d$

$$h_i = f(a_i; W_f)$$

with random masking

$$h_i = f(a_i, M_i; W_f)$$

Transformer [31]

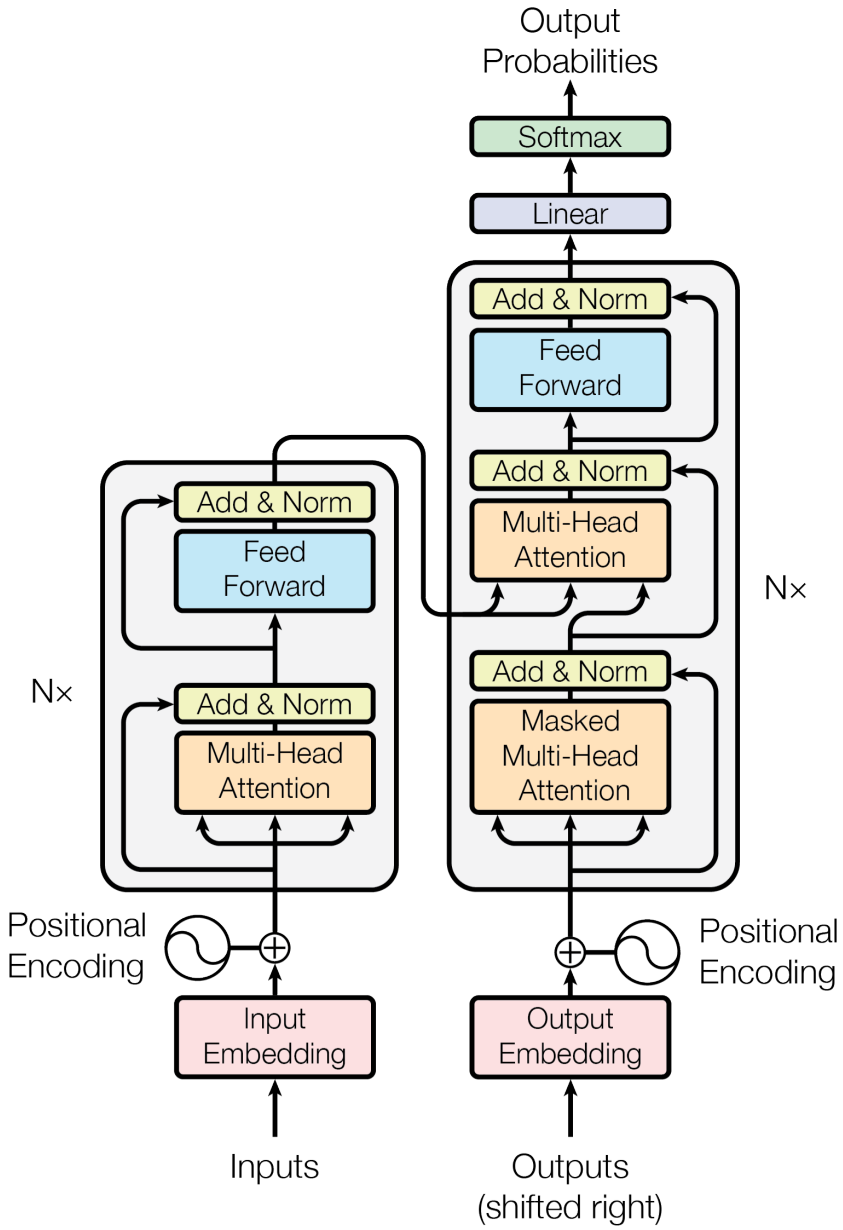
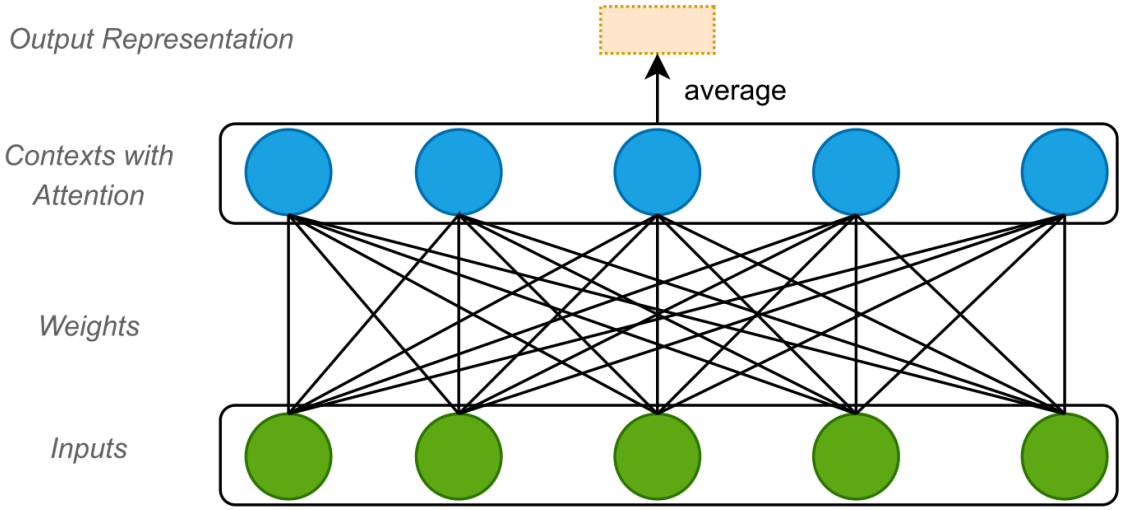


Figure 1: The Transformer - model architecture.

Random Masking

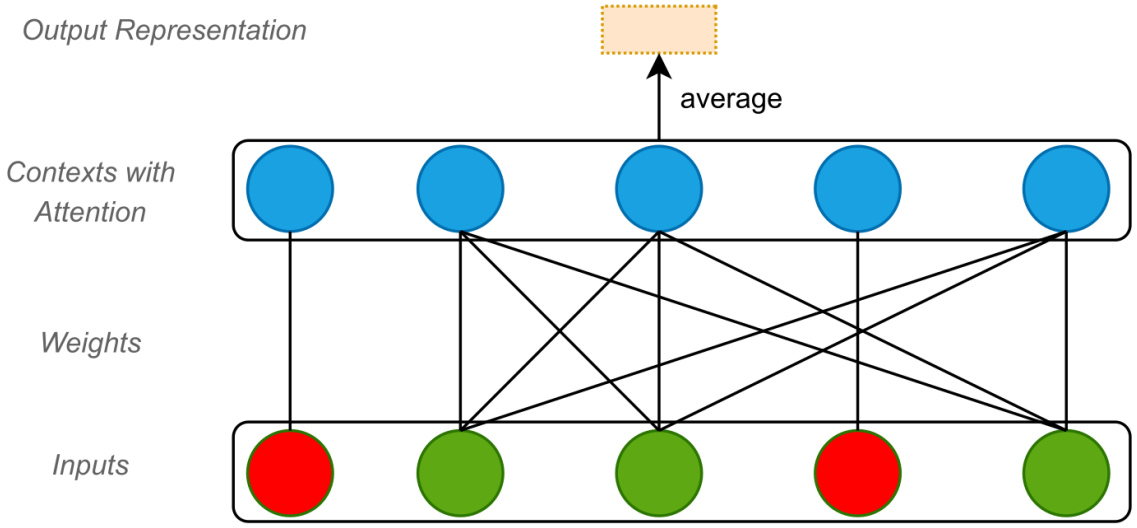
1. Avoid overfitting
2. For a respiratory sound, the feature at each time step might not be always meaningful.



(a) Without Random Masking

Random Masking

$h_i = f(a_i, M_i; W_f)$, where M_i is the masking matrix for clip a_i



(b) With Random Masking

Projection Head

- ***[7]*** $g(\cdot)$ that maps representations to the space where contrastive loss is applied. We use a MLP with one hidden layer to obtain $z_i = g(h_i) = W^{(2)}\sigma(W^{(1)}h_i)$, where σ is a ReLU non-linearity
- ***[26]*** $g(\cdot)$ contains a fully-connected layer with 512 units followed by a Layer Normalization and a tanh activation
- Both **drop** projection head in downstream phase, because it will cause information loss

Similarity Metrics:

Cosine Similarity:

$$\text{sim}(a_i, a_j) = \frac{g(h_i)^\intercal \cdot g(h_j)}{||g(h_i)|| \cdot ||g(h_j)||}$$

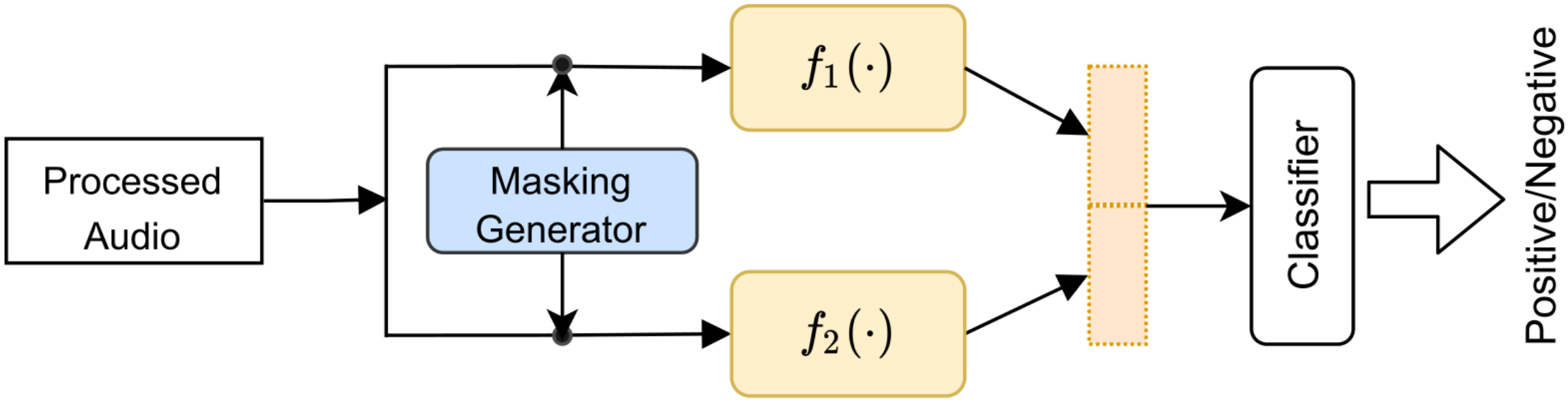
Bilinear Similarity:

$$\text{sim}(a_i, a_j) = g(h_i)^\intercal \cdot W_s g(h_j)$$

Loss Function

$$\mathcal{L}_{\text{contrasive}} = -\log \frac{\exp(\text{sim}(a_i, a_j)/\tau)}{\sum_{k=1}^{2\mathcal{B}} \exp(\text{sim}(a_i, a_k)/\tau)}, k \neq i$$

Downstream Cough Classification



Experiment

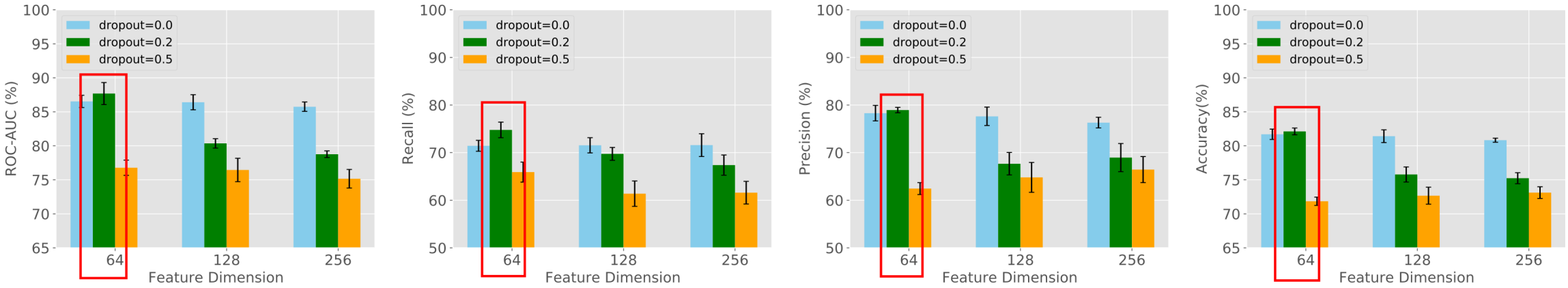
Dataset

1. Coswara Dataset - used in pre-training phase
    - **1,486** crowdsourced samples (1,123 males and 323 females)
    - **Four** types of sounds(breathing, coughing, counting, sustained phonation of vowel sounds) from each participant
2. COVID-19 Sounds - used in downstream phase
    - crowdsourced with an APP
    - **141** COVID-19 positive recordings from 62 participents and **298** negative recordings from 220 participants
    - coughs and breaths

Evaluation Metrics

- ROC-AUC
- Precision
- Recall
- Accuracy
- Average F1

Hyperparameter Fine-Tunning



(a) ROC-AUC (b) Recall (c) Precision (d) Accuracy

Figure 5: The performance of different hyperparameter settings on the validation set.

Performance Comparison

Table 1: Results (on the testing set) of different models and configurations. For each result, the standard deviation is reported in a bracket.

Model	Self-supervise	Pre-train	Fine-tune	ROC-AUC	Recall	Precision	Accuracy	Average F1
VGGish	×	×	N/A	76.14 (0.21)	53.19 (2.29)	73.17 (1.87)	74.88 (0.22)	61.60
	×	✓	×	85.02 (1.73)	67.42 (2.06)	78.55 (2.21)	80.71 (1.32)	72.56
	×	✓	✓	87.34 (1.14)	69.49 (1.44)	**83.15 (1.46)**	83.12 (0.31)	75.71
GRU	×	×	N/A	84.43 (0.88)	65.60 (1.43)	82.67 (1.89)	81.76 (0.39)	73.15
Transformer	×	×	N/A	87.60 (0.71)	71.53 (1.12)	80.64 (1.19)	82.73 (0.41)	75.81
GRU-CP	✓	✓	×	83.20 (0.43)	63.43 (1.82)	78.63 (1.22)	79.63 (0.31)	70.22
	✓	✓	✓	87.08 (0.35)	71.72 (2.53)	81.61 (2.26)	83.15 (0.32)	76.35
Transformer-CP	✓	✓	×	84.34 (0.71)	64.94 (1.80)	78.56 (1.40)	80.02 (0.42)	71.10
	✓	✓	✓	**88.83 (0.53)**	**73.07 (0.65)**	81.99 (0.92)	**83.74 (0.39)**	**77.27**

Different Similarity Metrics

Table 2: Results (on the testing set) of two types of similarity metrics that are used in the contrastive pre-training phase.

Model	Similarity Metric	ROC-AUC	Recall	Precision	Accuracy	Average F1
GRU-CP	Cosine	87.08 (0.35)	71.72 (2.53)	81.61 (2.26)	83.15 (0.32)	76.35
	Bilinear	87.25 (0.65)	71.99(1.97)	82.62 (1.92)	83.65 (0.32)	76.94
Transformer-CP	Cosine	87.02 (0.53)	67.79 (2.12)	81.73 (1.54)	82.06 (0.33)	74.11
	Bilinear	88.83 (0.53)	73.07 (0.65)	81.99 (0.92)	83.74 (0.39)	77.27

Random Masking Performance (Pre-training)

Table 3: Cough classification results (on the testing set) of different masking rates used in the contrastive pre-training phase.

Model	Masking Rate (CP)	ROC-AUC	Recall	Precision	Accuracy	Average F1
Transformer	NA	87.60 (0.71)	71.53 (1.12)	80.64 (1.19)	82.73 (0.41)	75.81
Transformer-CP	0%	88.83 (0.53)	73.07 (0.65)	81.99 (0.92)	83.74 (0.39)	77.27
	25%	89.17 (0.13)	73.05 (0.83)	82.29 (1.13)	83.84 (0.38)	77.40
	50%	**89.42 (0.51)**	**73.09 (0.43)**	**83.26 (0.28)**	**84.26 (0.12)**	**77.84**
	75%	89.13 (0.95)	72.66 (1.20)	82.01 (1.52)	83.62 (0.41)	77.05
	100%	88.37 (0.57)	72.41 (0.64)	81.65 (0.82)	83.41 (0.41)	76.75

Ensembles Performance

Table 4: Cough classification results (on the testing set) of different ensemble configurations.

Ensemble_1	Ensemble_2	ROC-AUC	Recall	Precision	Accuracy	Average F1
VGGish	Transformer	87.58 (0.73)	70.30 (1.05)	82.46 (1.44)	83.10 (0.26)	75.90
GRU	Transformer	87.24 (0.59)	72.56 (1.81)	81.10 (1.54)	83.20 (0.38)	76.59
GRU-CP	Transformer-CP	88.90 (0.38)	72.77 (1.85)	83.59 (1.62)	84.04 (0.35)	77.81

Random Masking Performance (Downstream)

Table 5: Results (on the testing set) of combining different masking rates with ensembles in the downstream phase.

Ensemble_1	Ensemble_2	Masking (DS)	ROC-AUC	Recall	Precision	Accuracy	Average F1
Transformer-CP	Transformer-CP	0%	88.77 (0.87)	71.98 (0.67)	82.81 (0.59)	83.75 (0.38)	77.02
		25%	89.02 (0.55)	71.93 (1.01)	82.85 (0.48)	84.15 (0.20)	77.00
		50%	**90.03 (0.41)**	**73.24 (0.22)**	**84.57 (1.24)**	**84.43 (0.25)**	**78.50**
		75%	89.22 (0.24)	72.12 (0.32)	83.17 (0.21)	84.03 (0.58)	77.25
		100%	89.55 (0.91)	71.21 (1.79)	82.85 (2.33)	83.93 (0.17)	76.59

Inference Speed

Table 6: Comparison of inference speed of different model and configurations. Each method is benchmarked on the same NVIDIA GeForce RTX-2080 Ti GPU.

Ensemble	Model	Self-supervise	Pre-train	Masking Rate (DS)	Inference time (10^{-6} seconds)
×	VGGish	×	×	N/A	6.39
		×	✓	N/A	6.39
	GRU	×	×	N/A	8.34
	Trasformer	×	×	N/A	8.48
	GRU-CP	✓	✓	N/A	8.60
	Transformer-CP	✓	✓	N/A	8.52
✓	VGGish + Transformer	×	×	N/A	12.58
	GRU + Transformer	×	×	N/A	14.26
	GRU-CP + Transformer-CP	✓	✓	N/A	14.36
	Transformer-CP + Transformer-CP	✓	✓	0%	18.86
		✓	✓	25%	24.53
		✓	✓	50%	27.56
		✓	✓	75%	32.36
		✓	✓	100%	18.64

Conclusion

1. First study to leverage unlabelled respiratory audios in the area
2. the propsed contrastive pre-training, the random masking mechanism, and the ensemble architecture contribute to improving cough classification performance.

References

- https://haythamfayek.com/2016/04/21/speech-processing-for-machine-learning.html