Exploring wav2vec 2.0 model for heart murmur detection

Davoud Shariat Panah
School of Computer Science
Technological University Dublin
Dublin, Ireland
davoud.x.shariatpanah@mytudublin.ie

Andrew Hines
School of Computer Science
University College Dublin
Dublin, Ireland
andrew.hines@ucd.ie

Susan McKeever
School of Computer Science
Technological University Dublin
Dublin, Ireland
susan.mckeever@tudublin.ie

***Abstract*—The lack of access to cardiology resources in many regions of the world has motivated the development of automatic diagnostic systems based on cardiac signals. In recent years, a wide range of supervised learning models have been proposed that can make an initial diagnosis of heart disease from heart sounds. To achieve high accuracy, however, such supervised learning models generally require a large amount of labeled data, which can be costly to obtain. In this regard, self-supervised learning has been recently employed to reduce the over-reliance on annotated data. Wav2vec 2.0 is an audio self-supervised learning model that has shown promising results in a variety of speech-related tasks. In this paper, we adapted the wav2vec 2.0 for murmur detection from heart sound signals. For this purpose, we pre-trained and fine-tuned this model on the Circor DigiScope heart sound dataset. The results confirm the feasibility of using the wav2vec 2.0 model for heart sound classification. The model shows a competitive performance by achieving a weighted accuracy of 0.80 and a UAR of 0.70 for murmur detection on the holdout test set. To investigate the impact of the fine-tuning data size on the downstream performance, we also fine-tuned the wav2vec 2.0 model on small sizes of annotated data. The results confirm that this model is robust to small fine-tuning data sizes, and as a result, can reduce our reliance on large, annotated heart sound data.**

***Index Terms*—Heart Sound, Murmur Detection, wav2vec 2.0, Self-supervised Learning**

I. Introduction

Cardiovascular diseases are a major cause of death around the world. Each year, over 18 million people die of heart disease which accounts for around one-third of all mortalities worldwide [1]. Early diagnosis using pervasive and low-cost techniques can reduce the high mortality rate due to heart disease. In this regard, using cardiac signals such as heart sounds towards developing automatic heart disease detection systems has recently become an active area of research. A wide range of data-driven classification models have been proposed that can provide a preliminary diagnosis of heart disease by detecting abnormalities (murmurs or extra sounds) in heart sound signals [2].

This work was conducted with the financial support of the Science Foundation Ireland Centre for Research Training in Digitally-Enhanced Reality (d-real) under Grant No. 18/CRT/6224. The authors wish to also acknowledge the Irish Centre for High-End Computing (ICHEC) for the provision of computational facilities and support.

Most of the existing heart sound classification models have been developed using supervised learning algorithms and in particular deep learning techniques. However, to achieve high accuracy rates, supervised learning algorithms generally require a large amount of annotated data which can be costly and time-consuming to obtain. This is more of a challenge for biomedical data, as domain-specific medical knowledge is required for accurate and consistent data labeling. Self-supervised learning (SSL) is another machine learning paradigm that rather than using human annotations, obtains supervisory clues from the data itself by solving a pretext task. SSL can leverage large amounts of unlabelled data to produce meaningful representations that can be subsequently used for a variety of downstream tasks and domains, including computer vision [3], natural language processing [4], and speech processing [5], [6]. SSL techniques have also been applied to biomedical signals [7], [8]. There are also a small number of examples of using SSL techniques to develop heart sound classification models [9], [10]. However, based on the limited existing research, it is hard to conclude whether such models can reduce our reliance on annotated heart sound data or not.

In the last few years, SSL has been adopted successfully in the audio processing field. BYOL-A [11], wav2vec 2.0 [6] and HuBERT [5] are just a few examples of recently proposed audio SSL models. Such models generally differ in terms of the architecture, input format (i.e., raw waveform vs. handcrafted features) and the task they are designed for (i.e., general audio vs. speech) [12]. Wav2vec 2.0 is a widely adopted SSL framework for learning speech representations. This model takes raw waveforms as input and benefits from a transformer-based architecture. The wav2vec 2.0 model has achieved promising results in a variety of speech-related tasks such as speech recognition [6], speaker recognition [13], and language identification [14]. While this model was originally designed for speech recognition, prior work has shown its potential to offer competitive performance for non-speech audio tasks such as music classification [15]. Although some efforts have been made towards adapting wav2vec 2.0 for the processing of brain signals [16], we could not find any adaptation of this model for biomedical audio signals, and in particular, heart sounds.

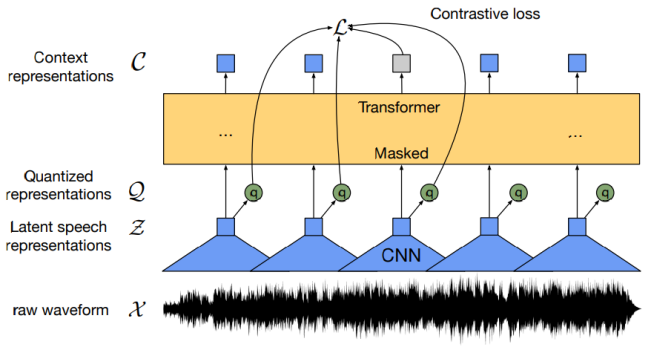

Fig. 1. – Illustration of the main components of the wav2vec 2.0 model (pre-training phase). Adapted from [6].

In this study, we employ wav2vec 2.0 as a successful and widely used audio SSL model to provide an end-to-end solution for heart murmur detection. In particular, we aim to answer the following questions: 1) Can we achieve a competitive accuracy for heart murmur detection by pre-training and fine-tuning the wav2vec 2.0 model on heart sound data? 2) How does the choice of data classes used in the pre-training phase impact the model's performance? 3) How does the fine-tuning data size affect the model's performance?

To answer the above questions, we pre-train and fine-tune the wav2vec 2.0 model on unlabeled and labeled heart sound data, respectively. For this purpose, we use a dataset that includes heart sounds captured at different auscultation points across patients' chest areas and annotated based on the presence or absence of heart murmurs. We investigate how classes of data used for pre-training of the model affect the downstream performance by training multiple models with data of different classes. We also explore the impact of fine-tuning data size on the downstream performance by fine-tuning a model with varied amounts of annotated data. To assess the performance of the models, we evaluate them on a separate test set and provide patient-level results for each model.

II. Methods

A. The wav2vec 2.0 Model

Fig. 1 illustrates the main components of the wav2vec 2.0 model [6]. As shown in Fig. 1, this model includes three modules: the feature encoder, context network, and quantization module. Following, we summarize the structure and function of each module:

Feature encoder: This module comprises a 7-layer single-dimensional convolutional neural network (1D-CNN) which takes raw waveform $\mathcal{X}$ as input and outputs $\mathcal{Z}$ latent speech representations.

Context network: This module consists of 12 transformer blocks with 8 attention heads each and a model dimension of 768 (Base model). Context network receives masked latent vectors as input and produces contextualized representations $\mathcal{C}$.

Quantization module: This module discretizes the continuous latent speech representations $\mathcal{Z}$ into quantized representations $\mathcal{Q}$. Quantized units are automatically learned by sampling from the Gumbel-Softmax distribution. These units are composed of codewords that are sampled from codebooks. The wav2vec 2.0 quantization module includes 2 codebooks with 320 codewords in each one.

To pre-train the wav2vec 2.0 model, a certain proportion of the feature encoder outputs are randomly masked before being fed into the context network. We should note that the inputs to the quantization module remain unmasked. The pre-training process uses a contrastive loss $\mathcal{L}$ which requires distinguishing the true quantized latent speech representation within a set of negative distractors. The contrastive loss encourages high similarity with the true positive target while at the same time penalizes high similarity scores with the distractors. Readers can refer to [6] for more details regarding the wav2vec 2.0 and its pre-training process.

B. Data Pre-processing

For training and evaluation of the models, we use CirCor DigiScope heart sound dataset [17] which was introduced as part of the George B. Moody PhysioNet Challenge 2022 [18]. This dataset contains 3163 recordings from 963 patients. The duration of recordings varies between 5 to 65 seconds. For each patient, multiple recordings were captured from different auscultation locations on the patient's body. Each patient was annotated based on the presence or absence of the cardiac murmur in their recordings:

- **Present**: murmur waves were detected in at least one heart sound recording of the patient (179 patients)
- **Absent**: murmur waves were not detected in any heart sound recording of the patient (695 patients)
- **Unknown**: the presence or absence of murmurs was unclear for the annotator (68 patients)

We split each recording into 5-second segments with a stride of 2.5 seconds. Segments are labeled according to the patients' labels (e.g., if a patient is labeled as murmur present, the present label is assigned to all 5-second segments of that patient). According to [19], the average heartbeat cycle duration is 0.8 seconds, and as a result, a 5-second segment would include six heartbeat cycles on average. Also, the 2.5-second stride allows us to increase the amount of data which is used for the training and evaluation of the models. The original sampling rate of the recordings is 4 kHz. We resample the segments to 16 kHz in accordance with the training requirements of the wav2vec 2.0 model [6]. Also, amplitude normalization is applied to all segments. The segments are randomly apportioned into training, validation, and test sets. The training set includes 65% of the data (22 hours), 10% of the data (3.5 hours) goes to the validation set, and the test set contains 25% of the data (8.5 hours). We should note that the segments are stratified by class labels. Also, when splitting the segments into the aforementioned sets, we make sure that the data from the same patients do not appear in different sets.

C. Model Pre-training

Pre-training of the model is performed using the fairseq framework [20]. We randomly sample 80% of the segments in the train set and use them for pre-training of the model. Three pre-training configurations are explored: pre-training on all classes (absent, present, and unknown), pre-training on two classes (absent, present), and pre-training on one class only (absent). Analyzing the results of these configurations allows us to understand how the classes of data used in the pre-training phase affect the downstream performance. Depending on the configuration, pre-training took 24-31 hours on two Nvidia V100 GPUs.

D. Model Fine-tuning

To fine-tune the pre-trained model, an average pooling layer and a fully connected layer are added on top of the wav2vec 2.0 model. Then, fine-tuning is carried out using the entire labeled training set. To investigate the impact of different fine-tuning strategies on the downstream performance, three configurations are considered: 1) Freezing both the feature encoder and context network and doing feature extraction by only training the fully connected layer (FE), 2) Fine-tuning the context network while keeping the feature encoder frozen (FT1), and 3) Fine-tuning the whole network (FT2). Also, to understand how fine-tuning data size influences the model's performance on the downstream task, we decrease the amount of annotated training data and fine-tune the best-performing model using smaller data sizes. These data sizes include 1/2 (50%), 1/4 (25%), 1/8 (13%), and 1/16 (6%) of the training set. These data sizes are in fact specific percentages of all segments available in the training set.

For all configurations, training is performed for a maximum of 20 epochs. To avoid overfitting, the training process is stopped if validation loss does not decrease for 5 consecutive epochs. The optimization is performed using AdamW optimizer [21] and OneCycle learning rate scheduler [22] with a maximum learning rate of 10e-5. As mentioned in Section II-B, the dataset used in this study is imbalanced. To address the class imbalance, a weighted cross-entropy loss is employed, with weights equal to the inverse probability of the classes.

E. Evaluation

For each configuration, the best-performing epoch is chosen based on the performance on the validation set and evaluated on the test set. We should note that the model's predictions are segment-level. To determine the prediction for each patient, we need to first aggregate segment-level predictions to produce recording-level predictions. To this end, the average of the segments' probabilities for each class is calculated and the class with the highest probability determines the recording's prediction. Then, the following rules are employed to aggregate the recording-level predictions and produce the patient-level results:

- Assign *present* if at least one recording was classified as present.
- Assign *unknown* if none of the recordings was classified as present, and at least one recording was classified as unknown.
- Assign *absent* if all recordings were classified as absent.

The above aggregation rules give the highest priority to the murmur present class given the clinical priority of detecting heart problems, then the unknown class, and lastly the absent class.

We use three metrics to report the performance of the models: (1) *Recall* which is used to measure the models' performance across each class. (2) *Unweighted average recall* (UAR) [23] which is used to quantify the overall performance of the models. (3) *Weighted accuracy* (W.acc) [18] which was introduced in PhyioNet 2022 challenge and used to rank the models submitted to the challenge. The goal of the challenge was to identify the presence, absence or unclear cases of murmurs from multi-auscultation location heart sound recordings. This metric is similar to the accuracy metric, with the difference being that it gives more weight to the higher cost present and unknown classes than the absent class, as a missed diagnosis is more harmful than a false positive [18].

TABLE I
MODELS' EVALUATION RESULTS ON THE TEST SET. RESULTS ARE PATIENT-LEVEL.

Model #	FT config	W.acc	UAR	Recall (absent)	Recall (present)	Recall (unknown)
Pre-train on absent, present and unknown classes						
1	FE	0.75	0.65	0.91	0.73	0.30
2	FT1	0.77	0.68	0.82	0.82	0.41
3	FT2	0.78	0.69	0.84	0.82	0.41
Pre-train on absent and present classes						
4	FT1	0.78	0.70	0.81	0.82	0.48
5	FT2	**0.80**	0.70	0.83	**0.86**	0.41
Pre-train on absent class only						
6	FT1	0.77	**0.73**	0.85	0.75	0.59
7	FT2	0.77	0.71	0.83	0.77	0.53

III. RESULTS AND DISCUSSION

A. Pre-training and fine-tuning configurations

Table I summarizes the models' evaluation results on the test set for all pre-training and fine-tuning configurations. Following, we analyze the evaluation results for different pre-training and fine-tuning configurations:

Fine-tuning configurations: The results indicate that freezing the entire model and training the fully connected layer only (FE) leads to a lower performance compared to the other two fine-tuning configurations (FT1 and FT2) for most metrics. This is expected as in the FE configuration, we perform simple feature extraction and only train the fully connected layer. Therefore, the results of this fine-tuning configuration have only been reported in the case of the first pre-training scenario (pre-training on all classes). The other two fine-tuning configurations (FT1 and FT2) perform roughly similar in terms of the *UAR* and *weighted accuracy* metrics.

Pre-training configurations: The models that are pre-trained on all three classes (Models 1-3) show a slightly

lower *UAR* compared to models from the other pre-training scenarios (Models 4-7). Pre-training on both present and absent classes leads to the highest *weighted accuracy* and present class *recall* (Model 5) while pre-training on only the absent class leads to the highest *UAR* (Model 6). Crucially, however, pre-training on only the absent class leads to a lower present class *recall* compared to the other pre-training scenarios. This is expected as in the pre-training phase, the present class is excluded, and the model is not pre-trained on present class samples. These results show the importance of pre-training on both absent and present classes to achieve higher recall values for the present class, which is obviously the most important class for the murmur detection problem.

Overall analysis: The results indicate that we should avoid pre-training on only the absent class as it leads to a low present class *recall*. Also, pre-training on all three classes leads to a slightly lower downstream performance compared to pre-training on absent and present classes. In other words, adding unknown class samples to pre-training data reduces the downstream performance. Given that unknown class samples are likely noisier than the samples of the other two classes, this observation suggests that the cleanliness of the data influences the quality of the learned representations. Regarding the fine-tuning strategies, the results indicate that we should fine-tune both the feature encoder and context network (FT2) or at least the context network (FT1). Given that the weighted accuracy metric prioritizes the present class over the other two classes, we use this metric to determine **Model 5** as the best-performing model.

B. Comparison with Previous Work

In this section, we compare our best-performing model (Model 5) with previous work. Table II compares the results of this model with that of the PhysioNet 2022 challenge winners for the heart murmur detection task on the challenge training set. We should note that due to different evaluation strategies and data splits, we are not able to directly compare the evaluation results of our model with that of previous work. However, reporting these results would allow us to have an estimate of the position of the proposed method compared to the previous work.

As shown in Table II, the performance of our model (wav2vec 2.0) is similar to the previous work in terms of weighted accuracy. In terms of the UAR, the proposed method performs slightly better than the CUED_Acoustics [24] model. If we compare the recall at the class level, we can see that the CUED_Acoustics model performs better on the present class while our model performs better on absent and unknown classes. The results of the other two challenge winners (HearTech+ [25] and CAU_UMN [26]) in terms of the recall on each class have not been reported. Therefore, we cannot provide a comparison with those models in terms of the recall and UAR metrics. These results confirm the possibility of achieving competitive performance in murmur detection task by pre-training and fine-tuning the wav2vec 2.0 model on heart sound data.

TABLE II
COMPARING THE BEST-PERFORMING MODEL WITH PREVIOUS WORK

Model	W.acc	UAR	Recall (absent)	Recall (present)	Recall (unknown)
wav2vec 2.0	0.80	0.70	0.83	0.86	0.41
CUED_Acoustics	0.80	0.68	0.78	0.93	0.34
HearTech+	0.81	NA	NA	NA	NA
CAU_UMN	0.79	NA	NA	NA	NA

C. Impact of the Fine-tuning Data Size

The train set which is used for fine-tuning contains about 22 hours of heart sounds. Out of these, 17 hours belong to the absent class, 4 hours to the present class, and 1 hour to the unknown class. As discussed in Section II-D, in addition to the entire training set, we fine-tune the model using 1/2 (50%), 1/4 (25%), 1/8 (13%) and 1/16 (6%) of the training set. Fig. 2 shows the evaluation results of the best-performing configuration (Model 5) fine-tuned on different data sizes. To remove any potential bias due to the aggregation rules employed to produce patient-level results, segment-level results are provided. Following, we analyze the results for each class separately:

Absent class: Recall fluctuates slightly for different amounts of data (between 0.82 to 0.89).

Present class: Recall fluctuates between 0.60 and 0.68 in the case of the first four scenarios (100% - 13% of the data). For the last scenario (6% of the data), the recall falls to 0.48. However, we should note that only about 15 minutes of murmur present samples are used for training the model in the last scenario.

Unknown class: Recall fluctuates between 0.47 and 0.53 for the last four scenarios. For the first scenario (whole training set), recall of the unknown class is 0.32 which is the lowest recall compared to the other scenarios. However, if we consider the recall values of the other two classes (present and absent) in the first scenario, we can see that they are larger compared to that of the other four scenarios with smaller data sizes. In other words, it seems that in the case of the first scenario, higher recall values of the present and absent classes led to a lower recall of the unknown class. This is expected because any unknown sample in the dataset is ultimately murmur-absent or present, and a classifier that can detect absent and present samples more accurately would classify a larger number of unknown samples as absent or present. Also, given that the train set only contains 1 hour of unknown samples, it would be difficult to achieve a high recall on the unknown class, and as a result, higher recall values of the unknown class in the last four scenarios may not be realistic.

Overall analysis: The results indicate that reducing the amount of training data does not have a large impact on the performance of the model. We can achieve comparable results by fine-tuning the model on a considerably smaller amount of annotated data (13% of the training set or around 3 hours of heart sounds). In other words, these results show that the wav2vec 2.0 SSL model can reduce our reliance on

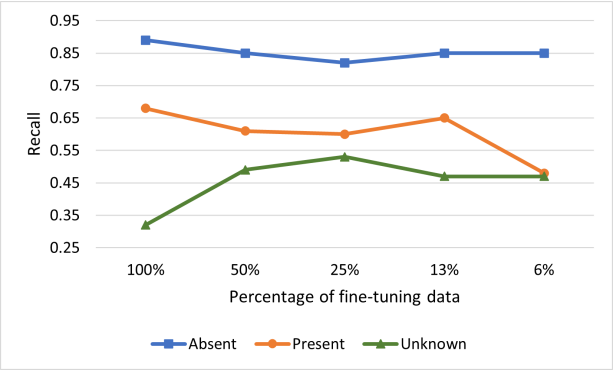

Fig. 2. Performance of the best-performing configuration fine-tuned on different data sizes of the training set (segment-level results)

the annotated heart sound data. This is an important finding as labeling heart sound data is hard and requires domain-specific medical knowledge. This finding is in line with previous studies that explored the influence of fine-tuning data size on the performance of the wav2vec 2.0 model for speech-related tasks [27].

IV. Conclusion

In this study, we showed the feasibility of employing the wav2vec 2.0 model for murmur detection task. We demonstrated that by pre-training and fine-tuning this model on heart sound data, we can achieve competitive murmur detection performance. We saw that including both present and absent classes while at the same time excluding the unknown class (noisier data) in the pre-training phase improves the model's downstream performance. By fine-tuning the model on different data sizes, we also showed that the wav2vec 2.0 model is robust to small data sizes. In other words, this model can decrease our dependency on large, annotated heart sound datasets without sacrificing much accuracy. This is particularly important in this domain because there is only a limited number of publicly available annotated heart sound datasets.

The wav2vec 2.0 SSL model has been originally designed for speech-related tasks. Although we achieved competitive performance on the heart murmur detection task using the original architecture and hyperparameters of the wav2vec 2.0, in the future, we will modify the model to tailor that to the heart sound classification task. Such modifications can potentially include changing the receptive field of the feature encoder, changing the context network architecture, and modifying the model's codebook size. We will also extend this work by exploring the impact of the size and quality of the data used in the pre-training phase on the downstream performance.

References

[1] "World health organization." [https://www.who.int/health-topics/cardiovascular-diseases; Online; accessed 2023-02-15].

[2] A. K. Dwivedi, S. A. Imtiaz, and E. Rodriguez-Villegas, "Algorithms for automatic analysis and classification of heart sounds–a systematic review," *IEEE Access*, vol. 7, pp. 8316–8345, 2019.

[3] I. Misra and v. d. L. Maaten, "Self-supervised learning of pretext-invariant representations," p. 6707–6717, 2020.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[5] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[6] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," 10 2020. arXiv:2006.11477 [cs, eess].

[7] P. Sarkar and A. Etemad, "Self-supervised ecg representation learning for emotion recognition," *IEEE Transactions on Affective Computing*, vol. 13, pp. 1541–1554, 7 2022.

[8] H. Banville, I. Albuquerque, A. Hyvarinen, G. Moffat, D.-A. Engemann, and A. Gramfort, "Self-supervised representation learning from electroencephalography signals," (Pittsburgh, PA, USA), pp. 1–6, IEEE, 10 2019.

[9] A. Ballas, V. Papapanagiotou, A. Delopoulos, and C. Diou, "Listen2yourheart: A self-supervised approach for detecting murmur in heart-beat sounds," 10 2022. arXiv:2208.14845 [cs, eess].

[10] P. N. Soni, S. Shi, P. R. Sriram, A. Y. Ng, and P. Rajpurkar, "Contrastive learning of heart and lung sounds for label-efficient diagnosis," *Patterns*, vol. 3, p. 100400, 1 2022.

[11] D. Niizumi, D. Takeuchi, Y. Ohishi, N. Harada, and K. Kashino, "Byol for audio: Self-supervised learning for general-purpose audio representation," (Shenzhen, China), pp. 1–8, IEEE, 7 2021.

[12] S. Liu, A. Mallol-Ragolta, E. Parada-Cabaleiro, K. Qian, X. Jing, A. Kathan, B. Hu, and B. W. Schuller, "Audio self-supervised learning: A survey," *Patterns*, vol. 3, p. 100616, 12 2022.

[13] N. Vaessen and D. A. van Leeuwen, "Fine-tuning wav2vec2 for speaker recognition," pp. 7967–7971, 5 2022. arXiv:2109.15053 [cs, eess].

[14] Z. Fan, M. Li, S. Zhou, and B. Xu, "Exploring wav2vec 2.0 on speaker verification and language identification," 1 2021. arXiv:2012.06185 [cs, eess].

[15] A. Ragano, E. Benetos, and A. Hines, "Learning music representations with wav2vec 2.0," 10 2022. arXiv:2210.15310 [cs, eess].

[16] D. Kostas, S. Aroca-Ouellette, and F. Rudzicz, "Bendr: using transformers and a contrastive self-supervised learning task to learn from massive amounts of eeg data," 1 2021. arXiv:2101.12037 [cs, q-bio].

[17] J. Oliveira, F. Renna, P. D. Costa, M. Nogueira, C. Oliveira, C. Ferreira, A. Jorge, S. Mattos, T. Hatem, T. Tavares, A. Elola, A. B. Rad, R. Sameni, G. D. Clifford, and M. T. Coimbra, "The circor digiscope dataset: From murmur detection to murmur classification," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, pp. 2524–2535, 6 2022.

[18] M. A. Reyna, Y. Kiarashi, A. Elola, J. Oliveira, F. Renna, A. Gu, E. A. Perez Alday, N. Sadr, A. Sharma, S. Mattos, M. T. Coimbra, R. Sameni, A. B. Rad, and G. D. Clifford, "Heart murmur detection from phonocardiogram recordings: The george b. moody physionet challenge 2022," tech. rep., 8 2022. DOI: 10.1101/2022.08.11.22278688.

[19] V. N. Varghees and K. Ramachandran, "A novel heart sound activity detection framework for automated heart sound analysis," *Biomedical Signal Processing and Control*, vol. 13, pp. 174–188, 9 2014.

[20] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli, "fairseq: A fast, extensible toolkit for sequence modeling," 4 2019. arXiv:1904.01038 [cs].

[21] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[22] L. N. Smith and N. Topin, "Super-convergence: Very fast training of neural networks using large learning rates," vol. 11006, p. 369–386, SPIE, 2019.

[23] Z. Ren, K. Qian, F. Dong, Z. Dai, W. Nejdl, Y. Yamamoto, and B. W. Schuller, "Deep attention-based neural networks for explainable heart sound classification," *Machine Learning with Applications*, vol. 9, p. 100322, 2022.

[24] A. McDonald, M. Gales, and A. Agarwal, "Detection of heart murmurs in phonocardiograms with parallel hidden semi-markov models,"

[25] Y. Xu, H.-K. Lam, and E. N. Kamavuako, "Hierarchical multi-scale convolutional network for murmurs detection on pcg signals,"

[26] J. Lee, T. Kang, N. Kim, S. Han, H. Won, and W. Gong, "Deep learning based heart murmur detection using frequency-time domain features of heartbeat sounds,"

[27] H. Becerra, A. Ragano, and A. Hines, "Exploring the influence of fine-tuning data on wav2vec 2.0 model for blind speech quality prediction," pp. 4088–4092, ISCA, 9 2022.