Identifying Subjects Wearing a Mask from the Speech by Means of Encoded Speech Representations

José Vicente Egas-López[2(✉)] and Gábor Gosztolya[1,2]

[1] Institute of Informatics, University of Szeged, Szeged, Hungary
ggabor@inf.u-szeged.hu
[2] ELKH-SZTE Research Group on Artificial Intelligence, Szeged, Hungary
egasj@inf.u-szeged.hu

Abstract. In the current pandemic situation, one of the tools used to fight Covid-19 is wearing face masks in specific public spaces. As previous research on the Mask Augsburg Speech Corpus had verified, speech might be eligible to automatically determine whether the speaker is wearing a mask or not, but the performance of classification models is far from perfect at the moment. This paper employs seven transformer-based wav2vec2 models on this dataset, extracting the activations from the lower, convolutional blocks as well as from the higher, contextualized transformer blocks. We show that models obtained via the self-supervised pre-training phase lead to similar performances with both activation types. However, after fine-tuning the models for direct ASR purposes, the performance achieved by the contextualized representations dropped significantly. Here, we report the highest Unweighted Average Recall value on this corpus that was achieved by a standalone method.

Keywords: speech analysis · surgical mask · wav2vec2 · computational paralinguistics · transformers

1 Introduction

Although with the introduction of vaccines, the peak of the COVID-19 pandemic seems to be over, the virus is still widely spread worldwide. To reduce the number of new infection cases, besides social distancing, an effective tool was the compulsory wearing of masks. Automatic speech analysis might offer a solution to enforce and monitor whether this regulation is kept. Furthermore, forensics and *'live'* communication between surgeons may also benefit from a system that could determine whether a subject is wearing a mask based on their speech [20]. This task belongs to the area of computational paralinguistics, which focuses on information present in speech other than the actual words uttered.

This research was supported by the Hungarian Ministry of Innovation and Technology NRDI Office (grant TKP2021-NVA-09) and by the Artificial Intelligence National Laboratory (MILAB, RRF-2.3.1-21-2022-00004).

K. Ekštein et al. (Eds.): TSD 2023, LNAI 14102, pp. 131–140, 2023.
https://doi.org/10.1007/978-3-031-40498-6_12

It is well known that both Automatic Speech Recognition (ASR) and Speech Verification techniques can be applied to the field of computational paralinguistics and pathological speech processing. For instance, x-vectors [22] (a former SOTA for Speaker Recognition) have been successfully adapted to classify emotions [15] and for sleepiness detection [11]. Furthermore, ASR-based solutions have also been adapted to these fields, e.g. for detecting states of dementia [8] and for speech emotion recognition [7].

Nowadays, feature-encoder approaches are increasingly being applied by researchers in Speech Recognition. For instance, ASR has benefited from wav2vec 2.0 [3,6] and BERT [10,21], which are able to generate rich contextual representations from large amounts of unlabeled instances. Wav2vec 2.0 has been successfully applied in computational paralinguistics and pathological speech tasks, where pre-trained models were used to assess the emotions [16], to screen Alzheimer's Disease [17], or even to detect COVID-19 [4] from the speech and the coughing of subjects. The wav2vec 2.0 method is said to be a state-of-the-art method for Speech Recognition, as it has the lowest Phonetic Error Rate (8.3%) [3] and lowest Word Error Rate (WER) (1.4%) [24] on two of the most popular speech datasets, namely TIMIT and LibriSpeech, respectively[1].

In this paper, we utilize several (pre-trained) wav2vec 2.0 speech encoder models and extract two distinct types of embeddings from them. The basis of wav2vec relies on the goal of extracting new types of input vectors from raw (unlabeled) audio, which can be used to build an acoustic model [19]. Wav2vec 2.0 relies on the same self-supervised principle, but it encodes speech representations from masked audio-segments and passes them to a transformer network that builds contextualized representations. This self-supervised approach was able to outperform traditional ASR systems that are based on transcribed audio, using much less labeled training data [3].

Our main contributions are: (i) Exploring the sufficiency of wav2vec 2.0 encoder (pre-trained) models for a task specifically related to computational paralinguistics; (ii) Analyzing the difference in the quality of the embeddings produced by each of the encoders; (iii) Applying a more straightforward method in order to avoid the time-consuming and computationally expensive fusion or ensemble approaches; (iv) Investigating the robustness of both language-domain matching and cross-lingual pre-trained encoders for the original language of the corpus utilized. Our approach gives the highest Unweighted Average Recall (UAR) score achieved by a stand-alone method on the above-mentioned corpus, while our performance stays above most of earlier studies that utilized fusion of methods as well.

2 Data

The Mask Augsburg Speech Corpus (MASC) comprises recordings of 32 German native speakers. The subjects were asked to perform specific types of tasks and their speech was recorded while wearing and not wearing a surgical mask. It

[1] Source: https://paperswithcode.com/task/speech-recognition/latest, Oct 2022.

has a total duration of 10 h, 9 min and 14 s, segmented into chunks of 1 s. The recordings have a sampling rate of 16 kHz. The total number of utterances is 36,554: 10,895 for train, 14,647 for development, and 11,012 for test. This task was also included in the Computational Paralinguistics Challenge (ComParE) in 2020 [20].

3 Self-supervised Learning

Self-supervised learning makes it possible for models to learn from orders of magnitude more data, which is the key to process patterns of less common phenomena. Usually, speech recognition systems require massive amounts of transcribed (labeled) training data to perform well [1]. A good way to tackle this is to *pre-train* neural networks, which allows a model to learn general representations from massive amounts of (labeled or unlabeled) information, and then it can be used for downstream tasks where the number of samples is limited. Now, we shall discuss concepts concerning pre-training, wav2vec, and wav2vec 2.0 frameworks.

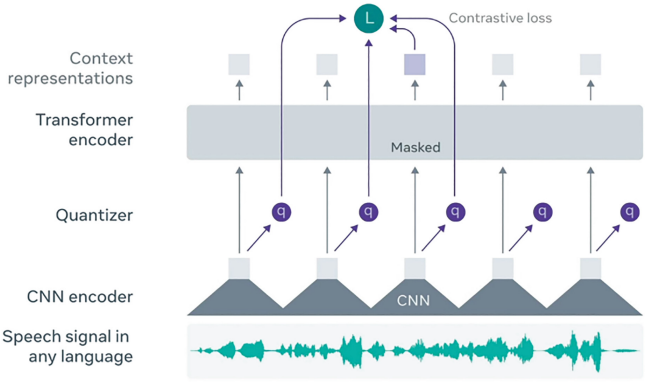

Fig. 1. Fine-tuned wav2vec 2.0 framework structure. Source: https://ai.facebook.com/blog

3.1 Pre-training and wav2vec

Pre-training consists of fitting a first neural network where huge amounts of data are available. The final weights from the training are then saved and this can be used to initialize a second neural network. This allows us to learn general representations from the large corpora; that is, representations that could be used for new tasks where the corpora size is limited.

wav2vec is basically a CNN that takes raw audio as input, and calculates a representation that can be fed into an ASR system. The wav2vec model is

optimized to predict the next observations of a given speech sample. This would require us to accurately model the distribution of the data $p(x)$. To tackle this, the dimensions of the speech sample are first reduced by means of an encoder network; then a context network is used to predict the subsequent values [19].

3.2 wav2vec 2.0

This model, being the successor to wav2vec, also uses a self-supervised approach to learn representations from raw audio. Similar to wav2vec, it learns to predict the correct speech unit, but it does so for masked chunks of the audio. More specifically, wav2vec 2.0 encodes raw audio using a block of convolutional neural networks, then akin to masked language modeling, it masks small segments (shorter than phonemes) of the latent speech representations. These representations are fed to a quantizer as well as to a transformer network. The former selects a speech unit for the latent audio representation, while the latter appends data from the whole utterance. Afterwards, the transformer network is exposed to a contrastive loss function [3]. After pre-training has been finished, the model is fine-tuned using labeled data relying on a Connectionist Temporal Classification (CTC) loss, which is used for aligning sequences. After doing this, the model can be utilized for downstream speech recognition tasks. Figure 1 shows the layout of the (fine-tuned) wav2vec 2.0 structure described here.

3.3 Cross-Lingual Representation Learning

A multi-lingual representation approach based on wav2vec2 named XLSR (Cross-lingual Speech Representations) addresses the issue of languages even with a limited amount of *unlabeled* data. XLSR pre-trains a model on multiple corpora from different languages simultaneously. XLSR uses a similar DNN structure to that shown in Fig. 1, i.e. it is trained to jointly learn context representations along with a discrete vocabulary of latent speech audio representations. The XLSR architecture differs from that of the wav2vec2 in the quantization module: in XLRS it delivers multilingual quantized speech units, which are then fed to the transformer block as targets to learn via a contrastive task. This way, the model is capable of handling tokens across different languages [5].

3.4 wav2vec 2.0 for Feature Extraction

The outputs from the multi-layer convolutional block are the sequence of extracted feature vectors of the last convolutional layer, while the outputs from the second block comprise the sequence of the hidden states at the output of the last layer of the block. These two types of feature vectors, the *convolutional embeddings*, and the *contextualized representations* may carry relevant information related to speakers [13] and also other information encoded in the speech signal [6]. Due to this, they will be exploited for deriving features for our paralinguistic classification task (i.e. determining whether the speaker is wearing

a mask). Of course, the actual classification step will be performed by another method, and wav2vec 2.0 will just be used for feature extraction. Also, since the number of wav2vec 2.0 embedding vectors is proportional to the length of the utterance, they have to be aggregated in some way, for which we simply took the mean of them over the time axis.

4 Experimental Setup

We extracted embeddings using seven different wav2vec 2.0 pre-trained models. The *first* is the so-called wav2vec2-base [3], which was pre-trained on 53k hours of unlabeled data of LibriSpeech, and it is not fine-tuned. The *second* is the wav2vec2-base-960h [3], pre-trained and fine-tuned using 960 h of labeled data. The *third* is a larger version of the previous one called wav2vec2-large-960h [3]. The main difference between these two is the number of parameters: *base* has 95 million, while *large* has 317 million parameters.

A cross-lingual wav2vec2 XLSR-53 model, trained on 53 different languages was our *fourth* model. Later, the successor of XLSR called XLS-R was introduced, which was pre-trained on about half million of hours of data in 128 languages [2]. Three different checkpoints of the model are available according to the number of parameters. Due to computational limitations, we just used the two smaller networks: wav2vec2-XLS-R-300M and wav2vec2-XLS-R-1B (300 million and 1 billion parameters, respectively). Lastly, to experiment with a model fine-tuned for the same language (i.e. German) as that in the MASC corpus, as the *seventh* model we employed the wav2vec2-XLSR-German-53 [9] encoder that was fine-tuned on the CommonVoice dataset.

We used a linear Support Vector Machine (SVM) for classification; the C complexity parameter was set in the range 10^{-5}, ..., 10^{1}, based on the performance on the dev set. As for the metrics, since it is the standard on the MASC corpus, we relied on Unweighted Average Recall (UAR).

5 Results and Discussion

Table 1 shows the UAR scores for each of the pre-trained models with their corresponding type of embeddings. Every XLSR and XLS-R encoder surpassed the baseline scores from the ComParE challenge [20], except for the wav2vec2-base and -large models that gave slightly lower scores. This might be due to the size of the data and the language-domain of the pre-training process for these models. Also, fine-tuning itself relies on adjusting the inherited initialization weights to fit a function that performs well on a specific downstream task (i.e., speech recognition on a given language). While the adaptation to this new task is being performed, the fine-tuning process may drop some information that might not be relevant for ASR but may be crucial for applications unrelated to this field (such as pitch, speaking rate, irregularity and breathiness). This may be the reason for the superior performance scores of wav2vec2 models specifically fine-tuned for ASR.

Table 1. UAR (%) on the MASC dataset. Models marked with * denote fine-tuned models.

Model Type	Embedding Type	Dev	Test
wav2vec2-base	convolutional	67.6	70.1
	contextualized	63.3	69.6
wav2vec2-base-960h*	convolutional	67.6	69.1
	contextualized	53.0	54.6
wav2vec2-large-960h*	convolutional	65.0	70.8
	contextualized	52.1	53.7
Cross-Lingual Models			
XLSR-53	convolutional	67.9	71.9
	contextualized	68.2	72.1
XLS-R-300M	convolutional	69.0	71.9
	contextualized	**70.3**	**76.9**
XLS-R-1B	convolutional	68.2	73.0
	contextualized	66.1	74.6
XLSR-German-53*	convolutional	67.9	71.9
	contextualized	57.1	62.4

In the models and their representations, a trend can be seen: for the *base* and *fine-tuned* models (see Table 1), the convolutional embeddings had a better quality than their contextualized counterparts; but the opposite was the case for the other models. This is probably due to the convolutional embeddings being more sensitive to mono-lingual training than the contextualized representations. The two best UAR scores on the test set were achieved with the *XLS-R-300M* and *XLS-R-1B* models using the contextualized representations, while their convolutional features had slightly lower performances.

The baseline scores reported by the organizers of the ComParE Mask Sub-Challenge can be seen at the top of Table 2: a UAR of 70.8% that corresponds to a non-fused score, and a 71.8% score for the fusion of the best four configurations [20]. The same table shows the performances of the most competitive previous studies on the same task. Szep et al. [23] reported an UAR score of 80.1% on test, being the highest one on MASC at the time of writing, achieved by training multiple image classifiers, a K-fold cross-validation approach, along with an ensembling of both the CNN classifiers and distinct types of spectrograms. Similarly, Koike et al. [12] reported a UAR score of 77.5% by transfer learning, two kinds of augmentation techniques, and a fusion based on several snapshots taken during DNN training. Markitantov et al. [14] used ensembles of different CNN architectures along with raw data plus two types of frame-level audio representations. Lastly, Ristea et al. [18] made use of an ensemble of GANs with a cycle-consistency loss along with a data augmentation method based on those GANs.

Table 2. Results of former studies on the same MASC corpus. * denotes the scores achieved by a fusion of multiple models.

Features in the ComParE 2020 paper [20]	Dev	Test
ComParE functionals	62.6	66.9
Bag-of-Audio-Words (BoAW)	64.2	67.7
Deep Spectrum	63.4	70.8
AuDeep	64.4	66.6
Four-wise fusion*	–	71.8
Former Studies		
Szep et al.* [23]	70.5	80.1
Markitantov et al.* [14]	84.3	75.9
Ristea et al.* [18]	71.8	74.6
Koike et al.* [12]	–	77.5
This work		
XLS-R-300M	70.3	76.9

The above studies carried out late fusion or ensembling techniques in order to boost their configurations, which is a usual strategy for these kinds of challenges. Although these techniques might improve our performance scores as well, in this study we were interested in the results obtainable with wav2vec2 models alone. The method presented in our paper is more straightforward and led to competitive results while keeping the machine learning pipeline much simpler. Our best performance is competitive with [23] and [12], and it outperforms the other studies listed in Table 2.

Lastly, to investigate if there was any redundancy in the wav2vec 2.0 models, we further experimented with transforming the features obtained from the contextualized layer of the XLS-R-300M model by PCA and Gaussian random projection. We kept 90%, 95% and 99% of the information present in the original 512 attributes. The results (and the sizes of the transformed feature vectors) can be seen in Fig. 2. Clearly, features compressed by random projection produced lower scores than those using PCA (with the same feature vector lengths). Even by retaining 95% of the information, the resulting UAR values were relatively low (64.7–71.3%). When we kept most of the information (99%), the feature vectors became almost as large as those without compression (467–470 attributes out of the original 512). And although there was only a slight drop in performance on the development set (0.8% absolute in both cases), the test set UAR scores were significantly lower (74.26% and 73.46%, PCA and random projection, respectively). This, in our opinion, indicates that the feature vectors are redundant to such a low degree that even a slight compression (PCA 99%) leads to a notable drop in classification performance.

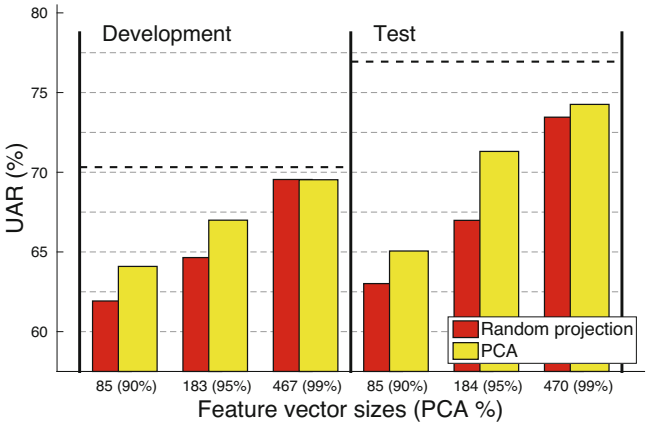

Fig. 2. UAR values after using PCA and random projection on the XLS-R-300M feature vectors. The dashed lines represent the scores obtained with all the attributes (i.e. Table 1)

6 Conclusions

Here, we investigated the effectiveness of employing wav-2-vec 2.0 embeddings for the identification of subjects wearing a mask based on their speech. We experimented with seven distinct pre-trained encoders for extracting convolutional and contextualized embeddings. It appears that the former were more sensitive to mono-lingual training than the latter, based on the quality difference of their corresponding feature vectors. The opposite occurred with the contextualized representations, which had lower performance scores when extracted using the fine-tuned models, which might discard information that is irrelevant for the ASR but important for computational paralinguistics. Based on the pre-trained cross-lingual encoders, both types of embeddings performed competitively and we demonstrated that the wav2vec2 architecture was capable of capturing speech and speaker traits that are relevant for paralinguistic approaches. Furthermore, we found that the number of training parameters is quite influential as models with 300 m provided better features than those with fewer (95 m) or more (1 billion) parameters both for pre-trained and fine-tuned encoders. Unlike earlier studies on the same dataset, we retained a simple yet effective and reproducible pipeline by dispensing with ensemble or fusion approaches while maintaining the competitiveness and even surpassing the performance score of most other studies. Overall, we achieved the highest UAR score (76.9%) reported on the MASC corpus obtained by a single (stand-alone) method.

References

1. Amodei, D., et al.: Deep speech 2: end-to-end speech recognition in English and mandarin. In: Proceedings of ICML, pp. 173–182 (2016)

2. Babu, A., et al.: XLS-R: self-supervised cross-lingual speech representation learning at scale. In: Proceedings of Interspeech, pp. 2278–2282 (2022)
3. Baevski, A., Zhou, Y., Mohamed, A., Auli, M.: wav2vec 2.0: a framework for self-supervised learning of speech representations. In: Advances in Neural Information Processing Systems, vol. 33, pp. 12449–12460 (2020)
4. Chen, X., Zhu, Q., Zhang, J., Dai, L.: Supervised and self-supervised pretraining based COVID-19 detection using acoustic breathing/cough/speech signals. In: Proceedings of ICASSP, pp. 561–565 (2022)
5. Conneau, A., Baevski, A., Collobert, R., Mohamed, A., Auli, M.: Unsupervised cross-lingual representation learning for speech recognition. In: Proceedings of Interspeech, pp. 2426–2430 (2021)
6. Fan, Z., Li, M., Zhou, S., Xu, B.: Exploring wav2vec 2.0 on speaker verification and language identification. In: Proceedings of Interspeech, pp. 1509–1513 (2021)
7. Feng, H., Ueno, S., Kawahara, T.: End-to-end speech emotion recognition combined with acoustic-to-word ASR model. In: Proceedings of Interspeech, pp. 501–505 (2020)
8. Fors, K., Fraser, K., Kokkinakis, D.: Automated syntactic analysis of language abilities in persons with mild and subjective cognitive impairment. In: Proceedings of MIE, pp. 705–709 (2018)
9. Grosman, J.: XLSR wav2vec2 German by Jonatas Grosman (2021)
10. Huang, W., Wu, C., Luo, S., Chen, K., Wang, H., Toda, T.: Speech recognition by simply fine-tuning BERT. In: Proceedings of ICASSP, pp. 7343–7347 (2021)
11. Huckvale, M., Beke, A., Ikushima, M.: Prediction of sleepiness ratings from voice by man and machine. In: Proceedings of Interspeech, pp. 4571–4575 (2020)
12. Koike, T., Qian, K., Schuller, B., Yamamoto, Y.: Learning higher representations from pre-trained deep models with data augmentation for the ComParE 2020 challenge mask task. In: Proceedings of Interspeech, pp. 2047–2051 (2020)
13. Lin, W.W., Mak, M.W.: Wav2Spk: a simple DNN architecture for learning speaker embeddings from waveforms. In: Proceedings of Interspeech, pp. 3211–3215 (2020)
14. Markitantov, M., Dresvyanskiy, D., Mamontov, D., Kaya, H., et al.: Ensembling end-to-end deep models for computational paralinguistics tasks: ComParE 2020 mask and breathing sub-challenges. In: Proceedings of Interspeech, pp. 2072–2076 (2020)
15. Pappagari, R., Wang, T., Villalba, J., Chen, N., Dehak, N.: X-vectors meet emotions: a study on dependencies between emotion and speaker verification. In: Proceedings of ICASSP, pp. 7169–7173 (2020)
16. Pepino, L., Riera, P., Ferrer, L.: Emotion recognition from speech using wav2vec 2.0 embeddings. In: Proceedings of Interspeech, pp. 3400–3404 (2021)
17. Qin, Y., et al.: Exploiting pre-trained ASR models for Alzheimer's disease recognition through spontaneous speech. arXiv preprint arXiv:2110.01493 (2021)
18. Ristea, N., Ionescu, R.: Are you wearing a mask? Improving mask detection from speech using augmentation by cycle-consistent GANs. In: Proceedings of Interspeech, pp. 2102–2106 (2020)
19. Schneider, S., Baevski, A., Collobert, R., Auli, M.: wav2vec: unsupervised pretraining for speech recognition. In: Proceedings of Interspeech, pp. 3465–3469 (2019)
20. Schuller, B.W., et al.: The INTERSPEECH 2020 computational paralinguistics challenge: elderly emotion, breathing & masks. In: Proceedings of Interspeech, Shanghai, China (2020)
21. Shin, J., Lee, Y., Jung, K.: Effective sentence scoring method using BERT for speech recognition. In: Proceedings of ACML, pp. 1081–1093 (2019)

22. Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., Khudanpur, S.: X-vectors: robust DNN embeddings for speaker verification. In: Proceedings of ICASSP, pp. 5329–5333 (2018)
23. Szep, J., Hariri, S.: Paralinguistic classification of mask wearing by image classifiers and fusion. In: Proceedings of Interspeech, pp. 2087–2091 (2020)
24. Zhang, Y., et al.: Pushing the limits of semi-supervised learning for automatic speech recognition. arXiv preprint arXiv:2010.10504 (2020)