INVESTIGATION OF ROBUSTNESS OF HUBERT FEATURES FROM DIFFERENT LAYERS TO DOMAIN, ACCENT AND LANGUAGE VARIATIONS

Pratik Kumar, Vrunda N. Sukhadia, S. Umesh

Speech Lab, Dept. of Electrical Engineering, IIT Madras, Chennai, India

ABSTRACT

In this paper, we investigate the use of pre-trained *HuBERT* model to build downstream Automatic Speech Recognition (ASR) models using data that have differences in domain, accent and even language. We use the standard ESPnet recipe with *HuBERT* as pre-trained models whose output is fed as input features to a downstream Conformer model built from target domain data. We compare the performance of *HuBERT* pre-trained features with the baseline Conformer model built with Mel-filterbank features. We observe that as the domain, accent and bandwidth (as in the case of *Switchboard* data) vary, the relative improvements in performance over baseline decrease significantly. Further, with more labelled data in the target domain, the relative improvement narrows down, and both systems become comparable. We also investigate the effect on ASR performance when output from intermediate layers of *HuBERT* are used as features and show that these are more suitable for data in a different language, since they capture more of the acoustic representation. Finally, we compare the output from Convolutional Neural Network (CNN) Feature encoder used in pre-trained models with the Mel-filterbank features and show that Mel-filterbanks are often better features for modelling data from different domains.

Index Terms— Self-supervised, *HuBERT*, Automatic Speech Recognition (ASR), Pre-training, Low resource speech recognition

1. INTRODUCTION

Recently, self-supervised learning has received a lot of attention in the speech recognition community. Manually transcribed text is expensive and limits the amount of supervised data available to build large end-to-end speech recognition systems that give good performance. Self-supervised methods, on the other hand, use audio-only data which are often easily available in large quantities. Using the audio-only data, these networks learn contextual representations that capture the sequence structure of acoustic units in a language. These large self-supervised networks often use tens of thousands of hours of speech-only data and use hundreds of GPUs running over several days to train the network. *Wav2vec2.0* [1] and *HuBERT* [2] are couple of such pre-trained models that are publicly available for download. Both these models have been trained on a large 60,000 hour corpus from *Libri-Light*. Using the encoder output embeddings from these models, large improvements in recognition performance in the low-resource setting have been obtained on the matched *LibriSpeech* data.

In this paper, we investigate the robustness of these pre-trained models when the target data is from a different domain, accent or even language. In [3, 4] a degradation in performance is observed when there is a shift in domain or across languages, in the case of cross-lingual model. The focus of these papers are to improve the models by adding target domain data in the pre-training step and pre-training the model again. This is of course expensive, and it is not clear how much improvement would be obtained by adding low-resource target domain data. The focus of this paper is to investigate these issues. In particular, we investigate:

- The effect of domain difference on the performance.
- The effect of embeddings from different layers on the performance. This is motivated by the idea that lower layers capture more of the acoustic information while the higher layers capture language and domain-specific information. [5]
- The use of latent representation obtained at the output of the feature encoder as features instead of conventional Mel-filterbank features.

The last point is motivated by previous studies that have tried to learn front-end representations [6, 7] instead of handcrafted features such as Mel-filterbanks. Since the final objective functions are different, these latent representations learnt in pre-trained models seem to be inferior to Mel-filterbank features when modelling data from a different domain.

In the next section, we first describe the different datasets used, followed by the set-up for fine-tuning with the *HuBERT* pre-trained model using the standard recipe from ESPnet. We then describe the experiments to investigate the points mentioned above and analyse the results obtained.

2. DATASETS

The different datasets used in the experiments for analysing the efficacy of the pre-trained *HuBERT* model's features for Automatic Speech Recognition (ASR) across domains, accents and languages are mentioned in Table 1.

Dataset	Train(hr)	Valid(hr)	Test(hr)
LibriSpeech-360	360.0	5.4	5.4
LibriSpeech-100	100.0	5.4	5.4
Wall Street Journal	80.0	1.1	0.4
Switchboard-300	313.4	5.1	5.1
Switchboard-30	35.5	5.1	5.1
Indian English-180	180.0	5.4	5.4
Indian English-40	40.0	5.4	5.4
Hindi-180	180.0	5.0	5.0
Hindi-40	40.0	5.0	5.0
Gujarati-40	40.0	5.0	5.0

Table 1: Details of the Datasets

HuBERT is trained on *Libri-Light* data. *LibriSpeech-100* and *LibriSpeech-360* datasets match *HuBERT*'s training data in domain,

accent and language. All three datasets are of US English data from read audio books. *WSJ* consists of US English speech of articles read from *Wall Street Journal*, and hence is different in domain from *Libri-Light*. *Switchboard* data consists of telephone conversations in US English and is different in many aspects like bandwidth, domain and speaking style. (The telephone bandwidth speech is upsampled to 16 KHz for our experiments).

Indian English and *Hindi* datasets comprise of both conversational speech and read speech. It contains speech data from different domains like news, articles, sports, movies and recorded conversations. These two datasets were released as a part of Automatic Speech Recognition challenge conducted by IITM Speech Lab. All of these datasets are publicly available. *Gujarati* data was released by Microsoft as part of the Low Resource Automatic Speech Recognition Challenge for Indian Languages conducted in Interspeech 2018. *Indian English* has differences in domain and accent, while *Hindi* and *Gujarati* are *Indo-Aryan* languages and are different in domain, accent and language from *Libri-Light*. The differences in datasets with respect to *Libri-Light* is summarised in Table 2.

Dataset	Domain	Accent	Language
WSJ *Switchboard*	Different	Same	Same
Indian English	Different	Different	Same
Hindi *Gujarati*	Different	Different	Different

Table 2: Comparison of Datasets with *Libri-Light*

3. EXPERIMENTAL SETUP

All models discussed in this paper were built in ESPnet2 [8, 9]. We have used pre-trained *HuBERT* (hubert_large_ll60k) [2] as the frontend for ASR in our proposed approach to analyse the efficacy of *HuBERT* features. *HuBERT* consists of a feature extractor built on Convolutional Neural Network (CNN) Encoder inspired by Van den Oord in [10] and 24 layers of Transformer encoders which have been trained to predict feature cluster labels. The number of parameters in this pre-trained model is approx 317M. This frontend is utilised for the downstream task of building Encoder-Decoder ASR models in Conformer framework [11]. ESPnet2 has a standard recipe for downloading and implementing *HuBERT* as the frontend for downstream ASR task. *HuBERT* weights were frozen for all the experiments, and therefore, the *HuBERT* embeddings are treated as features for subsequent model building.

The downstream ASR Conformer models use 12 encoder layers and 6 decoder layers, and are randomly initialised. Details of the hyperparameters of the Conformer models used in our experiments are in Table 3. The inputs to these models are either hand-crafted Mel-filterbank features to get the baselines, or embeddings from the *HuBERT* model for the proposed investigation. For Mel filterbank features, we have used 80-dimensional Mel-filterbank features with extra 3 pitch features which are extracted every 10 msec with 25 msec window length. In our experiments, we also analyse the performance of *HuBERT* features tapped from different layers of the pre-trained model. In such cases, the embeddings from the desired layer of *HuBERT* are tapped and fed to the Conformer model, instead of regular Mel-filterbank features, as shown in the Figure 1.

For training the downstream ASR task, characters or byte-pair encodings (BPE) are used as targets. ESPnet2 uses a default BPE

Hyper-parameter	Values
Kernel Size	15
Feature vector dimension	256
Encoder layers	12
Encoder units	1024
Decoder layers	6
Decoder units	1024
Attention heads	4
Beam width	20

Table 3: Conformer Configurations

count of 80 for *WSJ*. For *Switchboard-300*, *Indian English-180* and *Hindi-180*, 1000 BPE units gave optimal ASR performance. For all the other smaller datasets like *Switchboard-30*, *Indian English-40*, *Hindi-40* and *Gujarati-40*, 'characters' were used as targets.

All the Conformer experiments in this work were trained in a joint Connectionist Temporal Classification (CTC) / Attention framework[12] with CTC weight 0.3 and attention weight of 0.7. All the models were trained for 100 epochs. External language model [13] was not considered for the experiments in this work. We have done our evaluations based on the Word Error Rate (WER).

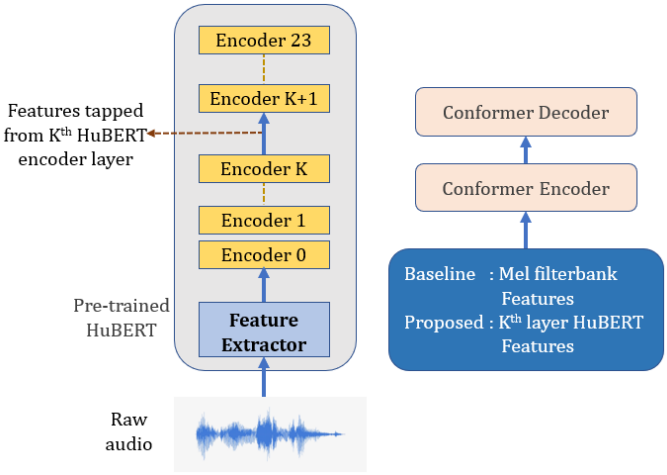

Fig. 1: Pre-trained *HuBERT* Model as Feature Encoder

4. PERFORMANCE OF *HUBERT* PRE-TRAINED MODEL ACROSS DOMAIN, ACCENTS, LANGUAGES AND DATA SIZE

In the original paper, *HuBERT* pre-trained with *Libri-Light* has shown impressive results on *LibriSpeech* [2]. It is to be noted that *LibriSpeech* is from the same domain, accent and language. In this paper, we analyse the generalisation of *HuBERT* features for different datasets which vary in domain, accent and language.

4.1. Generalisation Ability of the Conventional *HuBERT* Pre-trained Features for Different US English and Indian English Datasets

To study the generalisation ability of *HuBERT* features, we analyze its performance on different US English datasets that are from different domains and Indian English dataset that is from a completely

different accent. The conventional *HuBERT* features obtained from its final layer are used as features to train Conformer ASR models for each of the datasets. The performance of these models are compared with the conventional Mel-filterbank features. The results in terms of WER, with no external Language Model, are presented in Table 4. We have not used any external language model to better understand the acoustic modelling capability of the pre-trained *HuBERT* model. From Table 4 it can be observed and inferred that :

- *HuBERT* features perform better than Mel-filterbank features for all these English datasets.
- Performance gain in *LibriSpeech-100* and *LibriSpeech-360* is high due to a complete match in domain, accent and language with *Libri-Light*.
- For *WSJ*, which has a similar data size like *LibriSpeech-100*, the performance gain is less compared to *LibriSpeech-100*. This could be due to the mismatch in domain compared to *Libri-Light*.
- *Switchboard* gives the least improvement with *HuBERT* features among the US English datasets. This could be due to the several differences in this dataset from *Libri-Light* data. Apart from a mismatch in domain, *Switchboard* is also a conversational and a narrow band telephone speech dataset. Therefore a mismatch in bandwidth could have affected the performance improvement as well (even though we upsampled the data to 16KHz).
- *Indian English* gives the least improvement in performance compared to the US English datasets. The reason could be due to the additional effect of accent mismatch along with the domain mismatch.
- Overall, we can conclude that conventional *HuBERT* features are sensitive to both domain and accent mismatches in the datasets.

To overcome the problem of the domain and accent mismatch, one work-around is to use the target data along with original pre-trained data and repeat the entire pre-training process [3]. This is, however, very expensive and may not be useful in practice.

HuBERT is trained to predict cluster-labels from masked inputs. Therefore, it learns a strong context by capturing the sequential information of the acoustic units for that language. While the lower layers may capture low-level acoustic units, as we progress to higher layers, more and more domain and accent specific information is captured. So we hypothesise that tapping *HuBERT* features from intermediate layers might be beneficial when there is a mismatch in domain and accent, compared to the conventional *HuBERT* features, which are tapped from the final layer. Also, this approach is not as expensive as the approach proposed in [3]. So, in the forthcoming subsections we explore the effect of features from different HuBERT layers on the downstream ASR performance for different datasets.

4.2. Investigation of Optimal *HuBERT* Layer for Different US English and Indian English Datasets

In this section, we investigate *HuBERT* features from different layers on US English and Indian English datasets. The ASR performance for the *HuBERT* features from its different layers is presented in Table 5. From the Table it can be deduced that the features from the final layer of *HuBERT* are not optimal for different domain US English and Indian English datasets. 22^{nd} layer gives optimal WER for all the different Domain US English and Indian English datasets compared to the conventional 23^{rd} layer. As we go down the *HuBERT* layers, the performance degrades for all the mentioned datasets. So we can infer that:

- Features from the last layer of *HuBERT* are not optimal for different domain US English and Indian English datasets, even though the language is the same. This is because of differences in domain and accent.
- The performance gain from the 22^{nd} layer for mismatched datasets is not as significant as that of the 23^{rd} layer performance gain of *LibriSpeech*. Therefore, it can be concluded that there will be a significant gap between the performance gain of matched and mismatched datasets, even after the optimal layer search.

Dataset	Baseline Mel-filterbank	*HuBERT* Features	Relative Improvement WER (%)
LibriSpeech-100	8.6	3.0	65.0%
LibriSpeech-360	5.1	2.6	49.0%
WSJ	9.0	4.6	48.8%
Switchboard-300	10.2	9.8	4.0%
Switchboard-30	20.7	13.8	33.0%
Indian English-180	11.9	11.6	2.5%
Indian English-40	31.5	27.7	12.0%

Table 4: Relative Improvement with *HuBERT* Pre-trained Features over the Conventional Mel-Filterbank Features for Different US English Datasets. *(Note: No external Language models were used.)*

HuBERT Layer Tapped	*WSJ*	*Switchboard -30*	*Indian English -180*	*Indian English -40*
Baseline with Mel-filterbank features	9.0	20.7	11.9	31.5
23^{rd} Layer	4.6	13.9	11.6	27.7
22^{nd} Layer	4.3	13.8	11.5	27.1
21^{st} Layer	4.4	14.0	11.6	27.2
0^{th} Layer	10.3	22.2	14.8	35.8

Table 5: *HuBERT* Layer-wise Performance for US English and Indian English datasets. *(Note: No external Language Models were used.)*

4.3. Investigation of Optimal *HuBERT* Layer for Indo-Aryan Languages

We have taken two languages from the Indo-Aryan family for this investigation: *Hindi* and *Gujarati*. These languages are completely different from that of *Libri-Light*. The rationale behind this investigation is to check whether *HuBERT* learns low-level acoustic information about the speech which may be language agnostic. As shown in Table 6, for *Hindi* and *Gujarati*, optimal performance is obtained in layers much lower than in the previous analyses. For *Hindi*, 8^{th} layer gives an optimal performance with a relative improvement of 9.5% over the baseline and the conventional 23^{rd} layer *HuBERT* features. For *Gujarati*, the optimal *HuBERT* layer is the 9^{th} layer. The respective relative improvements over the baseline is 9.5% and 23^{rd} layer features is 4.7%. So we conclude that *HuBERT* does

learn language-agnostic general speech representations in the middle layers that may be useful for other languages as well. As we move up the *HuBERT* network, it becomes more specific to the source dataset in terms of domain, accent and language.

# of layers from which the output is taken	***Hindi-40***	***Gujarati-40***
Baseline with Mel-filterbank features	24.4	24.3
23^{rd} Layer	24.4	23.1
17^{th} Layer	23.7	22.7
11^{th} Layer	22.3	22.5
10^{th} Layer	-	22.4
9^{th} Layer	22.2	22.0
8^{th} Layer	22.1	22.1
7^{th} Layer	22.5	-
5^{th} Layer	24.0	23.0
0^{th} Layer	25.5	24.7

Table 6: *HuBERT* Layer-wise Performance for the Indo-Aryan datasets: *Hindi-40* and *Gujarati-40*. *(Note: No external Language models were used.)*

4.4. Conventional Mel-filterbank Features vs. 0^{th} layer *HuBERT* Features for Different Datasets

In both wav2vec2.0 and *HuBERT*, the input speech waveform goes through a sequence of convolutional layers to generate a feature sequence. This step is inspired by the original work of Van den Oord on representation learning using contrastive predictive coding [14]. Once the pre-training step is over, these feature encoder layers are frozen for subsequent downstream tasks. Previously, there have been multiple studies to learn feature representations directly from speech waveform instead of using hand-crafted features such as Mel-filterbank features [6, 7].

In this section, we investigate whether the output from feature-encoder layer (which is learnt directly from speech waveform) is competitive to the hand-crafted Mel-filterbank features. In the Table 7, we have compared the features from the 0^{th} layer of *HuBERT* with Mel-filter bank features. We observe that when the domain matches completely, in the case of *LibriSpeech-100*, *HuBERT*'s 0^{th} layer features are better than Mel-filterbank. For all the other datasets mentioned in Table 7, 0^{th} layer features of *HuBERT* are worse than the Mel-filterbank features. This indicates that even the lowest *HuBERT* layer is influenced by the data used in pre-training, and results in performance degradation when there are mismatches in the target domain data.

4.5. Investigation of the Effect of Dataset Size on the Performance Improvement with *HuBERT*'s Features

From Table 8, we observe that with the increase in data size, the effect of *HuBERT*'s features on the performance improvement reduces. Also, in case of mismatched data, with more data available in the target domain, the model will learn better to a point that the contribution from *HuBERT* features is not so significant.

Dataset	**Test set**	**Mel filterbank**	***HuBERT*** 0^{th} **layer**
LibriSpeech-100	test-clean	8.6	8.4
WSJ	eval93	9.0	10.3
Switchboard-300	swbd_dev	10.2	11.2
Switchboard-30	swbd_dev	20.7	22.2
Indian-English-180	eval_eng	11.9	14.8
Indian-English-40	eval_eng	31.5	35.8
Gujarati-40	eval_guj	24.3	24.7
Hindi-40	eval_hin	24.4	25.5

Table 7: Mel-filterbank vs *HuBERT* 0^{th} Layer Performance for Different Datasets *(Note : No external Language models were used.)*

Dataset	**Mel Filterbank**	**Best *HuBERT* Result**	**Relative Gain(%)**
LibriSpeech-100	8.6	3.0	65%
LibriSpeech-360	5.1	2.6	49%
Switchboard -30	20.7	13.8	33%
Switchboard-300	10.2	9.8	4%
Indian English-40	31.5	27.1	14%
Indian English-180	11.9	11.5	3.3%
Hindi-40	24.4	22.1	9.4%
Hindi-180	7.5	7.3	2.6%

Table 8: Performance Comparison of the Best *HuBERT* Features on Datasets of Different Sizes

5. CONCLUSIONS

In this work, we have explored the robustness of *HuBERT* pre-trained model for out of domain downstream ASR tasks. Our experiments indicate that as domain, accent, bandwidth and language deviates from the source domain, the relative improvement over baseline decreases. We have also investigated the effect of embeddings from different layers of *HuBERT* on the final performance. The last layer of *HuBERT* is very specific to the dataset on which it is trained. The second last layer seems to be better when there is domain and accent differences. As expected, the middle layers are more suited when data is from a different language, since they mostly capture the general acoustic characteristics of speech. Further, as more and more supervised data is available in the target domain, the relative improvement over baseline model decreases significantly. Finally, we have also compared the CNN feature encoder used in Conformer models with the Mel-filterbank features, and found that the Mel-filterbank features are better for modelling data from different domains. This could be because the pre-training objective function is different which also affects the CNN feature encoder during training.

6. REFERENCES

[1] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, Hugo Larochelle, Marc'Aurelio

Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, Eds., 2020.

[2] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *CoRR*, vol. abs/2106.07447, 2021.

[3] Wei-Ning Hsu, Anuroop Sriram, Alexei Baevski, Tatiana Likhomanenko, Qiantong Xu, Vineel Pratap, Jacob Kahn, Ann Lee, Ronan Collobert, Gabriel Synnaeve, and Michael Auli, "Robust wav2vec 2.0: Analyzing domain shift in self-supervised pre-training," *CoRR*, vol. abs/2104.01027, 2021.

[4] Alexis Conneau, Alexei Baevski, Ronan Collobert, Abdelrahman Mohamed, and Michael Auli, "Unsupervised cross-lingual representation learning for speech recognition," *CoRR*, vol. abs/2006.13979, 2020.

[5] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson, "How transferable are features in deep neural networks?," in *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger, Eds., 2014, pp. 3320–3328.

[6] Tara N. Sainath, Ron J. Weiss, Andrew W. Senior, Kevin W. Wilson, and Oriol Vinyals, "Learning the speech front-end with raw waveform cldnns," in *INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, September 6-10, 2015*. 2015, pp. 1–5, ISCA.

[7] Neil Zeghidour, Nicolas Usunier, Gabriel Synnaeve, Ronan Collobert, and Emmanuel Dupoux, "End-to-end speech recognition from the raw waveform," in *Interspeech 2018, 19th Annual Conference of the International Speech Communication Association, Hyderabad, India, 2-6 September 2018*, B. Yegnanarayana, Ed. 2018, pp. 781–785, ISCA.

[8] Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya Renduchintala, and Tsubasa Ochiai, "Espnet: End-to-end speech processing toolkit," in *Interspeech 2018, 19th Annual Conference of the International Speech Communication Association, Hyderabad, India, 2-6 September 2018*, B. Yegnanarayana, Ed. 2018, pp. 2207–2211, ISCA.

[9] Shinji Watanabe, Florian Boyer, Xuankai Chang, Pengcheng Guo, Tomoki Hayashi, Yosuke Higuchi, Takaaki Hori, Wen-Chin Huang, Hirofumi Inaguma, Naoyuki Kamo, Shigeki Karita, Chenda Li, Jing Shi, Aswin Shanmugam Subramanian, and Wangyou Zhang, "The 2020 espnet update: new features, broadened applications, performance improvements, and future plans," *CoRR*, vol. abs/2012.13006, 2020.

[10] Aäron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu, "Neural discrete representation learning," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, Eds., 2017, pp. 6306–6315.

[11] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang, "Conformer: Convolution-augmented transformer for speech recognition," in *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*, Helen Meng, Bo Xu, and Thomas Fang Zheng, Eds. 2020, pp. 5036–5040, ISCA.

[12] Jian Luo, Jianzong Wang, Ning Cheng, and Jing Xiao, "Loss prediction: End-to-end active learning approach for speech recognition," in *International Joint Conference on Neural Networks, IJCNN 2021, Shenzhen, China, July 18-22, 2021*. 2021, pp. 1–7, IEEE.

[13] Anjuli Kannan, Yonghui Wu, Patrick Nguyen, Tara N. Sainath, Zhifeng Chen, and Rohit Prabhavalkar, "An analysis of incorporating an external language model into a sequence-to-sequence model," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018, Calgary, AB, Canada, April 15-20, 2018*. 2018, pp. 5824–5828, IEEE.

[14] Aäron van den Oord, Yazhe Li, and Oriol Vinyals, "Representation learning with contrastive predictive coding," *CoRR*, vol. abs/1807.03748, 2018.