Feature Representations for Automatic Meerkat Vocalization Classification

Imen Ben Mahmoud[1], Eklavya Sarkar[1,2], Marta Manser[3], Mathew Magimai.-Doss[1]

[1]Idiap Research Institute, Martigny, Switzerland
[2]École polytechnique fédérale de Lausanne (EPFL), Switzerland
[3]University of Zurich (UZH), Switzerland

{ibmahmoud, esarkar, mathew}@idiap.ch, marta.manser@ieu.uzh.ch

Abstract

Understanding evolution of vocal communication in social animals is an important research problem. In that context, beyond humans, there is an interest in analyzing vocalizations of other social animals such as, meerkats, marmosets, apes. While existing approaches address vocalizations of certain species, a reliable method tailored for meerkat calls is lacking. To that extent, this paper investigates feature representations for automatic meerkat vocalization analysis. Both traditional signal processing-based representations and data-driven representations facilitated by advances in deep learning are explored. Call type classification studies conducted on two data sets reveal that feature extraction methods developed for human speech processing can be effectively employed for automatic meerkat call analysis.

Index Terms: bioacoustics, feature representations, self-supervised learning, call type classification

1. Introduction

Meerkats are highly social animals with a complex social structure [1]. Featuring a dominant breeding pair and cooperative behaviors, they dig safe places through their foraging areas. Communication among a clan occurs through various vocalizations including barks, chirps, trills, and growls. They are essential in coordinating group activities, warning of potential dangers, and maintaining social cohesion. Researchers have identified and classified around 30 types of vocalizations in meerkats [2]. These vocalizations can be categorized into alarm calls emitted when a potential predator is encountered [3], contact calls used to maintain group cohesion [4], and dominance calls employed during a conflict to assert social hierarchy. Additional vocalizations serve to express various other emotions. These vocalizations are part of a complex communication system, influenced by the group's social organization and ecology [5].

Over the past two decades, there has been a notable improvement in understanding this communication system, particularly in decoding the context of calls. For example, in [6], it is demonstrated that meerkat alarm calls encode information about both predator type and the signaler's perception of urgency simultaneously. Additionally, in [7], it was found that close calls are used to adjust movement direction and maintain group cohesion, especially in low-visibility environments and during continuous movement. However, understanding the context precedes contextual analysis. The process of categorizing calls is mainly conducted by human listeners, who rely on their expertise. Nonetheless, even among these experts, varying interpretations may arise, highlighting the complexity inherent in the classification task [8].

Although previous research has provided insights into the social and contextual aspects of meerkat vocalizations, there remains a lack of computational methods for the automatic analysis of this language. Specifically, to the best of our knowledge, there has not been a formal study on the automatic classification of meerkat vocalizations. One of the main reasons being that biological level and linguistic level analysis of meerkat vocalizations has evolved more recently, leading to the availability of reliable data sets for automatic analysis. As a first step, the present paper aims to investigate feature representations for automatic meerkat vocalization analysis. The motivation for this arises from the important role feature representation plays in pattern analysis and classification systems. In the past, in the field of speech and audio processing, these representations were largely obtained by combining prior knowledge with signal processing. Even though meerkat vocalizations have been analyzed using signal processing, there is still a lack of reliable prior knowledge to extract feature representations for automatic analysis. In recent years, with advances in deep learning, data-driven feature representations have become more prominent and have been demonstrated useful for bioacoustic analysis. In this paper, we investigate both types of feature representations.

The remainder of the paper is organized as follows: Section 2 introduces the two types of feature representations, providing a detailed overview of the methods used. Section 3 delineates the experimental setup and workflow, including the dataset used during the study, the classification setup, and the evaluation metric. Section 4 presents the classification results with a comprehensive analysis of the findings. Finally, Section 5 concludes our study.

2. Feature representations

This section motivates and presents the different feature representations investigated in this paper. These representations are grouped as (a) knowledge-based/hand-crafted feature representations and (b) neural-based data-driven feature representations.

2.1. Knowledge-based/hand-crafted feature representations

Catch22: Highly Comparable Time-Series Analysis (HCTSA) is an interpretable signal processing-based framework, where a set of 7700 features are extracted by characterizing the signal by different time series analysis methods, such as linear correlation, modeling fitting (e.g., autoregressive moving average analysis, GARCH), wavelet analysis, and extraction of information theoretic measures. It is then combined with feature selection to build statistical models for the end task [9]. The efficacy of this framework has been demonstrated for bioacoustic analysis. For instance, these features have been investigated for behavioral birdsong discrimination [10], automated acoustic

monitoring of ecosystems [11], as well as marmoset caller identification [12]. One of the limitations of the HCTSA approach is computational complexity, as it involves the evaluation of many similar features. In recent work, CAnonical Time-series CHaracteristics (Catch22) features, a subset of 22 HCTSA features that are minimally redundant has been proposed, and its utility has been demonstrated across 93 real-world time-series classification problems [13]. These features fall into different conceptual grouping such as distribution shape, linear autocorrelation, incremental differences, and self-affine scaling. The dimension of the feature set is 24 including the mean and the standard deviation.

COMPARE: COMPARE features have been developed for paralinguistic speech processing. The COMPARE feature set of length 6373 consist of functionals of (a) energy related low level descriptors (LLDs), (b) spectral LLDs, and (c) voicing related LLDs estimated over an utterance [14].

eGeMAPS: extended Geneva Minimalistic Acoustic Parameter Set (eGeMAPS) is yet another feature set developed for paralinguistic speech processing [15]. The feature set consists of 88 different features. They are obtained by extracting (a) LLDs, namely, frequency-related parameters, energy/amplitude related parameters, and spectral (balance) parameters, and (b) temporal features consisting of the rate of loudness peaks, mean length and standard deviation of voiced and unvoiced regions, and number of continuous voices regions per second from the acoustic signal.

2.2. Neural-based data-driven feature representations

Self-supervised learning-based: In traditional supervised learning, models rely on labeled data, which is expensive and time-consuming to obtain. Thus, the emergence of self-supervised learning (SSL) techniques offers a powerful alternative to these learning methods by leveraging unlabeled data and designing pretext tasks involving human speech. By doing so, it allows models to learn meaningful representations without relying on explicit human annotations. In [16], the authors explored leveraging embedding spaces focusing on the Marmoset caller discrimination. The study demonstrated that representations pre-trained on human speech could be effectively applied to the bio-acoustics domain. Motivated by that study, we chose three popular SSL models, namely, WavLM [17], wav2vec2 [18] and HuBERT [19], pre-trained with 960 hours of audio from Librispeech corpus [20]. We extract embeddings from one of the layers or all layers of the SSL model and model it for call classification.

Supervised-learning based (denoted as CNN-crafted): In this part, we focus on the feature extraction phase within a classification framework. This involves directly inputting waveform data into a neural network using an end-to-end Convolutional Neural Network (CNN) architecture. The architecture is inspired by [21] and is presented in Table 4. The model is trained to perform call type classification. After training, we derive a feature set of dimension 80 from each call by extracting the output of the penultimate layer of the model, referred to as CNN handcrafted features throughout the study.

3. Experiments

This section presents the dataset of our study, consisting of two Sets (A and B) of meerkat calls used during the study, followed by a detailed breakdown of the study's workflow.

3.1. Meerkat calls dataset

Set A consists of 90 audio recordings of 9 different meerkat call types collected and labeled by Prof. Marta Manser, University of Zurich, following ethical approval: Aggression (agg), Sentinel (sen), Alarm (al), Chatter (ch), Grooming (gr), Close-call (cc), Submission (sub), Lead (ld) and Sunning (su). Every file was manually segmented using Koe [22]; an open-source software to visualize, segment, and classify acoustic units in animal vocalizations, amounting to a total of 1795 vocalization segments at a sampling rate of 44.1 kHz, with a mean and median length of 161 ± 118 ms and 102 ms respectively. Table 1 shows the distribution of the different call types of Set A. It is crucial to emphasize that this table reveals a significant imbalance within the dataset, mirroring the real-world scenario.

Table 1: *Distribution of the different call types present in Set A.*

agg	sen	al	ch	gr	cc	sub	ld	su
125	411	609	108	12	81	99	28	322

Set B is a public dataset [23]. The corpus consists of 6428 individual files, categorized into 7 call types, sampled at 48 kHz with a mean of 148 ± 96 ms and a median of 124 ms. Four classes seen previously in Set A are also present in Set B, with three additional ones: Short note (sn), Social call (sc), and Move (mv). Table 2 displays the distribution of the different call types in Set B.

Table 2: *Distribution of the different call types present in Set B.*

agg	cc	al	ld	sn	soc	mo
375	1477	645	164	1854	1154	759

3.2. Experimental set-up

As a preprocessig step, we downsampled all waveforms to 16 kHz and vocalizations shorter than 100 ms were systematically replicated until they reached the desired minimum duration of 100 ms. To compare the feature representations, we adopted a 5-fold cross-validation strategy by employing 80:20 train-test split. Figure 1 shows the call classification framework. As illustrated in the figure, a call-level fixed length representation is obtained for each feature type and fed as input to a support vector machine (SVM) based classifier. We compare the feature representations by evaluating the respective call classifiers in terms of unweighted average recall (UAR). We chose UAR as metric due to class imbalance in the datasets. Unlike weighted average accuracy (classification accuracy), UAR measure gives importance to recognition of all classes. Higher UAR means higher recall across classes. When training the SVM classifier, we applied a grid search methodology on the training set of each fold with the Unweighted Average Recall (UAR) as the optimization criterion to search space of the hyperparameters (presented in Table 3). In the reminder of the section, we explain the call-level fixed length representation obtained for each feature representation type.

In the case of knowledge-based feature representation, (a) *pycatch22* toolkit was employed for extracting 24 dimensional call-level Catch22 features and (b) openSMILE [24] tool is used to extract 6373 dimensional call-level COMPARE feature representation and to extract 88 dimensional call-level eGeMAPS feature representations.
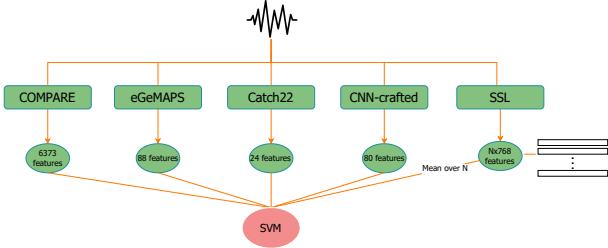
Figure 1: *Diagram of the workflow of the study. N denotes number of frames.*

Table 3: *SVM hyperparameters grid*

Parameter	Values
C	1e[-1, 0, 1, 2]
Gamma	1e[-3,-2,-1,0]
Kernel	['Linear', 'RBF', 'Polynomial', 'Sigmoid']

In the case of SSL feature representations, the call-level 768 dimensional feature representation is obtained as follows: (a) 768 dimension output of CNN encoder, 1st, 2nd, 6th or the last transformer layer is obtained per frame and averaged over frames, (b) the 768 dimension output of each of the 12 transformer layers are averaged per frame and then the resulting per frame representation is averaged over frames. The S3PRL toolkit [25] was used to extract the embeddings.

In the case of CNN-crafted feature representation, there is a need to train a CNN-based call classifier for feature extraction. As the data sets were small in size with severe class imbalance, as opposed to training a CNN feature extractor per fold, we employed stratified k-folds cross-validation strategy to get a single CNN feature extractor. This method constructs folds while maintaining class proportion integrity, i.e., ensuring consistent class proportions in both training and test sets, mirroring those of the original dataset. We set the number of folds to 5 and trained CNNs for each fold using the architecture presented in Table 4 using PyTorch. The adaptive average layer target size was set to one. This allows the network to handle variable length waveform inputs and yield fixed-length (80-dimensional) call level feature representation. We employed the cross-entropy error criterion to train the CNN. The CNN of the best performing fold was selected to extract 80 dimensional call-level CNN-crafted feature representation (from the output of the fully connected hidden layer).

4. Results and discussion

Table 5 presents an analysis of SSL neural embeddings. It can be observed the that lower layer transformer layer embeddings and CNN encoder representations yield better systems than higher layer transformer layer embeddings. Averaging the embeddings across the transformer layers, although yields better system than layer 6 and last layer embeddings, is not helpful when compared to layer 1 embedding, layer 2 embedding or CNN encoder output alone. Taken together, this indicates that lower transformer layer embeddings of SSLs pre-trained on human speech are more informative than higher transformer layer embeddings for meerkat call classification.

Table 6 compares the systems across different feature representations. For SSL feature representation wav2vec2, WavLM and HuBERT, we have reported the best system performance

Table 4: *CNN architecture for CNN-crafted feature extraction. n_f denotes number of filters. HU denotes number of hidden units.*

Block	Operation	Kernel	Stride	Padding	n_f/HU
1	Convolution	40	30	0	40
	Max Pooling	2	2	0	-
	ReLU Activation	-	-	-	-
2	Convolution	7	1	0	40
	Max Pooling	2	2	0	-
	ReLU Activation	-	-	-	-
3	Convolution	3	1	0	80
	Max Pooling	2	2	0	-
	ReLU Activation	-	-	-	-
4	Adaptive Avg Pooling	-	-	-	-
	Flatten	-	-	-	-
	Fully Connected	-	-	-	80

Table 5: *UAR scores of chosen representations using wav2vec2 (W2), WavLM (WL) and HuBERT (HT) models on Test set of Set A and B*

	Set A			Set B		
Model	W2	WL	HT	W2	WL	HT
CNN	0.71	0.68	0.74	0.78	0.77	0.78
1st Transformer	0.71	0.72	0.73	0.79	0.82	0.78
2nd Transformer	0.73	0.71	0.72	0.79	0.82	0.79
6th Transformer	0.54	0.50	0.64	0.69	0.70	0.76
Last Transformer	0.35	0.38	0.55	0.52	0.53	0.67
Average of Transformers	0.63	0.59	0.61	0.75	0.72	0.76

from Table 5. In the case of hand-crafted features, it is observed that eGeMAPS and COMPARE feature based systems yield better system than Catch22 feature representation. In the case of SSL feature representations, the systems are comparable. The CNN-crafted feature representation yields the best systems. When comparing hand-crafted features and neural embeddings, COMPARE feature outperforms SSL features on Set A and performs slightly worse when compared to wav2vec2 and HuBERT. It is worth pointing out that the COMPARE feature largely outperforms higher transformer layer embedding based systems (layer 6 and last layer in Table 6). This indicates that, similar to neural embeddings from networks pre-trained on human speech, hand-crafted representations developed for speech processing applications can be useful for meerkat call classification.

Table 6: *UAR scores on Test set of Set A and B with 5-fold CV for call types classification*

Model	Set A	Set B
eGeMAPS	0.61	0.66
COMPARE	0.80	0.75
Catch22	0.61	0.56
wav2vec2	0.73	0.79
WavLM	0.72	0.82
HuBERT	0.74	0.79
CNN-crafted	**0.84**	**0.84**

The main distinction between Set A and Set B lies in the number of classes, the number of samples, and the class dis-
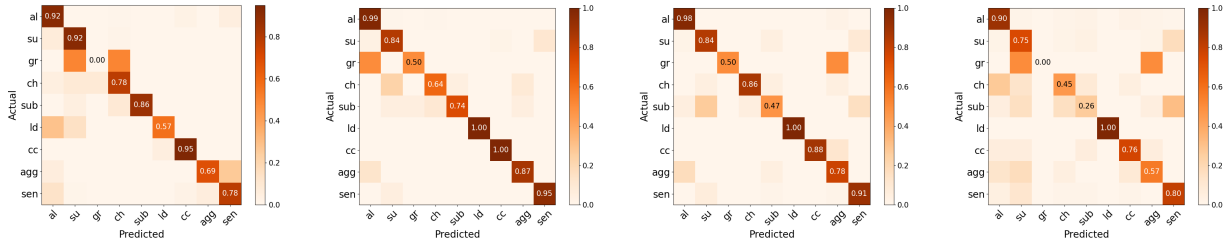
Figure 2: *Confusion matrices for SVM classifier using, from left to right, WavLM, CNN-crafted, COMPARE and Catch22 embeddings on the test set of Set A.*
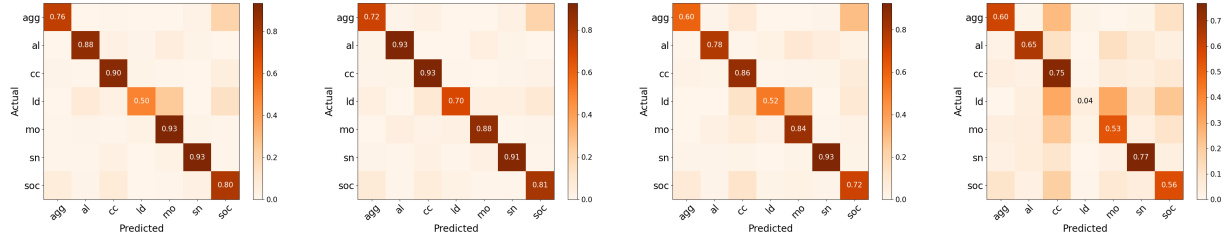

Figure 3: *Confusion matrices for SVM classifier using, from left to right, WavLM, CNN-crafted, COMPARE and Catch22 embeddings on the test set of Set B.*

tribution within the datasets. As discussed previously, Set B comprises more samples, fewer number of classes and exhibits better class balance than Set A. Therefore, our initial expectation was that Set B would yield superior performance. This hypothesis is confirmed with the SSL models, eGeMAPS, and the CNN model, where results with Set B perform better than Set A. Confusion matrices for WavLM, CNN-crafted, COMPARE and Catch22 are presented for Set A and Set B in Figure 2 and Figure 3. It can be observed that all the call types are mostly classified well except for "gr" in Set A which has the lowest amount of data.

For the case of CNN-crafted, Figure 4 shows the cumulative frequency response of the 40 first layer convolution filters. This is estimated by applying a DFT of 1024 points on filters of length 40 samples and taking logarithm of the summed magnitude responses. Although Set A and Set B have been collected independently and labeled, it can be observed that the cumulative filter responses of the CNNs of Set A and Set B are similar with a major emphasis between 0-2 kHz. This indicates that the CNNs are capturing information systematically for class classification across the two data sets. In our future work, we will investigate what kind of acoustic information does that frequency range carries in meerkat vocalizations for call analysis.

5. Conclusions

Meerkats with their highly social nature and diverse vocal repertoire, provide an intriguing model system for investigating animal communication and, as an extension could help us better understand the evolution of human communication. One of the challenges in that direction is the lack of methods for automatic meerkat call analysis. In that direction, this paper explored feature representations for automatic analysis of meerkat vocalizations. We compared time-series analysis-based hand-crafted feature representation, hand-crafted feature representations developed for human speech processing, SSL-based feature representations obtained from neural networks trained on human speech, and feature representations automatically learned in a
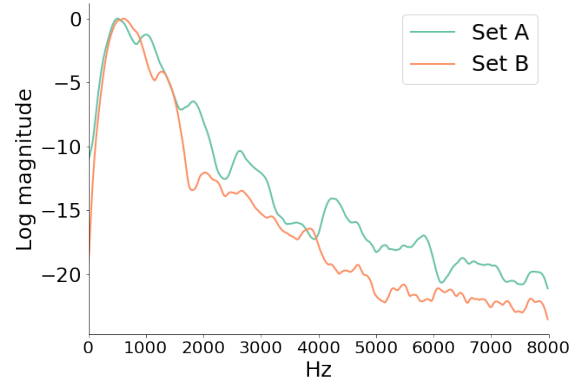

Figure 4: *Cumulative frequency responses of first layer filters of CNN*

task-dependent manner from meerkat calls using CNNs. Our studies show that hand-crafted feature extractors and SSL feature extractors developed for human speech processing can be used for meerkat call classification. Similarly, we observe that the CNN-based method developed for automatic feature learning in a task-dependent manner for human speech processing can be scaled for meerkat call classification task (CNN-crafted). Our future work will focus on analyzing these diverse feature representations to tease out and explain the acoustic information that is relevant for meerkat call analysis.

6. Acknowledgement

7. References

[1] J. R. Madden, J. A. Drewe, G. P. Pearce, and T. H. Clutton-Brock, "The social network structure of a wild meerkat population: 2. intragroup interactions," *Behavioral Ecology and Sociobiology*, vol. 64, no. 1, pp. 81–95, Nov 2009. [Online]. Available: https://doi.org/10.1007/s00265-009-0820-8

[2] S. W. Townsend, B. D. Charlton, and M. B. Manser, "Acoustic cues to identity and predator context in meerkat barks," *Animal Behaviour*, vol. 94, pp. 143–149, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0003347214002413

[3] G. Moran, "Vigilance behaviour and alarm calls in a captive group of meerkats, suricata suricatta," *Zeitschrift für Tierpsychologie*, vol. 65, no. 3, pp. 228–240, 1984. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1439-0310.1984.tb00101.x

[4] S. Engesser and M. B. Manser, "Collective close calling mediates group cohesion in foraging meerkats via spatially determined differences in call rates," *Animal Behaviour*, vol. 185, pp. 73–82, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0003347221003997

[5] M. B. Manser, D. A. Jansen, B. Graw, L. I. Hollén, C. A. Bousquet, R. D. Furrer, and A. le Roux, "Chapter six - vocal complexity in meerkats and other mongoose species," ser. Advances in the Study of Behavior. Academic Press, 2014, vol. 46, pp. 281–310. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780128002865000067

[6] M. B. Manser, R. M. Seyfarth, and D. L. Cheney, "Suricate alarm calls signal predator class and urgency," *Trends in Cognitive Sciences*, vol. 6, no. 2, pp. 55–57, 2002. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364661300018404

[7] G. E. C. Gall and M. B. Manser, "Group cohesion in foraging meerkats: follow the moving 'vocal hot spot'," *R. Soc. Open Sci.*, vol. 4, p. 170004, 2017.

[8] A. Kershenbaum, D. T. Blumstein, M. A. Roch, Çağlar Akçay, G. Backus, and M. A. B. et al., "Acoustic sequences in non-human animals: a tutorial review and prospectus," *Biological Reviews*, vol. 91, no. 1, pp. 13–52, 2016. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/brv.12160

[9] B. D. Fulcher, M. A. Little, and N. S. Jones, "Highly comparative time-series analysis: the empirical structure of time series and their methods," *Journal of The Royal Society Interface*, vol. 10, no. 83, p. 20130048, 2013. [Online]. Available: https://royalsocietypublishing.org/doi/abs/10.1098/rsif.2013.0048

[10] P. Avishek, H. McLendon, V. Rally, J. T. Sakata, and S. C. Woolley, "Behavioral discrimination and time-series phenotyping of birdsong performance," *PLoS Comput. Biol.*, vol. 17, no. 4, p. e1008820, Mar. 2021.

[11] S. S. Sethi, R. M. Ewers, N. S. Jones, A. Signorelli, L. Picinali, and C. D. L. Orme, "Safe acoustics: An open-source, real-time eco-acoustic monitoring network in the tropical rainforests of borneo," *Methods in Ecology and Evolution*, vol. 11, no. 10, pp. 1182–1185, 2020. [Online]. Available: https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.13438

[12] N. Phaniraj, K. Wierucka, Y. Zürcher, and J. M. Burkart, "Who is calling? optimizing source identification from marmoset vocalizations with hierarchical machine learning classifiers," *Journal of The Royal Society Interface*, vol. 20, no. 207, p. 20230399, 2023. [Online]. Available: https://royalsocietypublishing.org/doi/abs/10.1098/rsif.2023.0399

[13] C. H. L. S. S. Sethi, P. Knaute, S. R. Schultz, B. D. Fulcher, and N. S. Jones, "catch22: Canonical time-series characteristics," *Data Mining and Knowledge Discovery*, vol. 33, no. 6, pp. 1821–1852, Nov 2019. [Online]. Available: https://doi.org/10.1007/s10618-019-00647-x

[14] B. Schuller, S. Steidl, A. Batliner, J. Hirschberg, J. K. Burgoon, A. Baird, A. Elkins, Y. Zhang, E. Coutinho, and K. Evanini, "The interspeech 2016 computational paralinguistics challenge: Deception, sincerity & native language," in *Proc. Interspeech 2016*, 2016, pp. 2001–2005.

[15] F. Eyben, K. R. Scherer, B. W. Schuller, J. Sundberg, E. André, and C. B. et al., "The geneva minimalistic acoustic parameter set (gemaps) for voice research and affective computing," *IEEE Transactions on Affective Computing*, vol. 7, no. 2, pp. 190–202, 2016.

[16] E. Sarkar and M. Magimai.-Doss, "Can Self-Supervised Neural Representations Pre-Trained on Human Speech distinguish Animal Callers?" in *Proc. INTERSPEECH 2023*, 2023, pp. 1189–1193.

[17] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, and Z. e. a. Chen, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.

[18] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," 2020. [Online]. Available: https://arxiv.org/abs/2006.11477

[19] 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," 2021. [Online]. Available: https://arxiv.org/abs/2106.07447

[20] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.

[21] D. Palaz, M. Magimai-Doss, and R. Collobert, "End-to-end acoustic modeling using convolutional neural networks for hmm-based automatic speech recognition," *Speech Communication*, vol. 108, pp. 15–32, 2019.

[22] Y. Fukuzawa, W. H. Webb, M. D. Pawley, M. M. Roper, S. Marsland, D. H. Brunton, and A. Gilman, "Koe: Web-based software to classify acoustic units and analyse sequence structure in animal vocalizations," *Methods in Ecology and Evolution*, vol. 11, no. 3, pp. 431–441, 2020. [Online]. Available: https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.13336

[23] M. Thomas, F. H. J. B. Averly, V. Demartsev, M. B. Manser, T. Sainburg, M. A. Roch, and A. Strandburg-Peshkin, "A practical guide for generating unsupervised, spectrogram-based latent space representations of animal vocalizations," *Journal of Animal Ecology*, vol. 91, no. 8, pp. 1567–1581, 2022. [Online]. Available: https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/1365-2656.13754

[24] F. Eyben, M. Wöllmer, and B. Schuller, "Opensmile: the munich versatile and fast open-source audio feature extractor," in *Proceedings of the 18th ACM International Conference on Multimedia*, ser. MM '10. New York, NY, USA: Association for Computing Machinery, 2010, p. 1459–1462. [Online]. Available: https://doi.org/10.1145/1873951.1874246

[25] S. wen Yang, P.-H. Chi, Y.-S. Chuang, C.-I. J. Lai, K. Lakhotia, and Y. Y. L. et al., "SUPERB: Speech Processing Universal PERformance Benchmark," in *Proc. Interspeech 2021*, 2021, pp. 1194–1198.