Exploring Self-Supervised Speech Models for Low-Resource ASR: A Case Study on Badaga Using HuBERT and Wav2Vec2

1st M Vasista
Dept. of Artificial Intelligence
Amrita Vishwa Vidyapeetham
Coimbatore, India
CB.EN.U4AIE22134@am.students.amrita.edu

2nd Nandana Gireesh
Dept. of Artificial Intelligence
Amrita Vishwa Vidyapeetham
Coimbatore, India
CB.EN.U4AIE22138@am.students.amrita.edu

3rd Snega Sri A
Dept. of Artificial Intelligence
Amrita Vishwa Vidyapeetham
Coimbatore, India
CB.EN.U4AIE22163@am.students.amrita.edu

4th N Dakshinya
Dept. of Artificial Intelligence
Amrita Vishwa Vidyapeetham
Coimbatore, India
CB.EN.U4AIE22169@am.students.amrita.edu

***Abstract*—This paper presents the creation of an end-to-end automatic speech recognition (ASR) system for the Badaga language, an under-resourced Dravidian language of the Ooty district in India. A 9,837-sample speech corpus was collected from 11 native speakers, each sample with English translation, Badaga transliterated transcript, and speaker information such as user ID, gender, and data split labels. The data is around 2.4 seconds of audio and contains 5.1 words on average, perfect for short conversation ASR tasks. To establish strong performance baselines, we fine-tuned two state-of-the-art recent self-supervised learning models—Wav2Vec 2.0 and HuBERT—on the pre-cleaned data. Both are transformer-based models that leverage large-scale unlabeled speech to learn robust audio representations, which are extremely robust in under-resourced environments. Data preprocessing was done based on silence trimming, normalization, and stratified splitting to ensure quality and consistency. Performance metrics such as Word Error Rate (WER) and Character Error Rate (CER) were used to test on the test set, with the results showing the good performance of the models in transcribing Badaga speech. This paper is the first systematic effort on the development of ASR tools for the Badaga language and demonstrates how recent self-supervised models can be successfully used to document and digitize underdocumented languages.**

***Index Terms*—Automatic Speech Recognition, Low-Resource Languages, Badaga, Wav2Vec 2.0, HuBERT, Self-Supervised Learning**

I. Introduction

Among many types of interaction, natural speech is widely used by humans. While progress in automatic speech recognition (ASR) technology has made human-computer interaction much more natural, from voice assistants and transcription systems to access devices, most state-of-the-art ASR systems are optimized to thrive on high-resource languages such as English, Mandarin, or Spanish and leave low-resource languages to mass underrepresentation in technological advancements. Of all such languages in this category, one is the Dravidian language Badaga which is widely spoken in Ooty district in the Indian state of Tamil Nadu and spoken by around 135,000 individuals. Being a linguistic and cultural priority language, Badaga in itself is not a matter of valuable linguistic and digital content, and therefore it is a very suitable choice for low-resource ASR research.

We present the first comprehensive attempt at developing an ASR system for the Badaga language in this work. We developed a 9,837 audio-record speech-to-text corpus of 11 native English translation, transliteration script, and full metadata such as speaker ID, gender, and train-validation-test split flags. The corpus is evenly divided between male and female speakers and comprises short samples of speech with a mean sample length of 2.4 seconds and an even mean number of words per sample of 5.1 words. Two self-supervised methods, Wav2Vec 2.0 and HuBERT, are used for handling the sparsely labeled training data. Both the models were shown to perform well in low-resource setups with deep unsupervised pretraining of speech and then fine-tuning the same on small labeled datasets. Both the models were fine-tuned on our well-prepared Badaga dataset and compared two's performance in the Word Error Rate (WER) and Character Error Rate (CER) metric. Our work demonstrates the feasibility of self-supervised learning in training ASR models for under-resourced and endangered languages.

The activity is not just towards constructing a technology platform for Badaga language processing but also towards language preservation and digital empowerment of indigenous language communities.

II. Literature Review

Automatic Speech Recognition (ASR) systems have been around for a long time. However, due to recent advancements in computation power and innovations in algorithms, now it is being used in different industries, like Healthcare [8] Most of the recent ASRs are trained on openly available generic datasets like the Fisher Corpus [3] and Librispeech [9] There is always a scarcity, in industry, for an ASR system that could cater to the needs of automatic transcription using domain-specific language models. This is due to the limited availability of labeled data. However, there is still a very genuine need of ASR systems which are domain and industry-specific as they are more accurate and reliable compared to a generic ASR that is trained on the multi-domain dataset and using a language model that is not domain specific [12] In recent years, the continuous improvement of computer hardware performance and the development of deep learning technology have led to significant changes in feature extraction methods and model architectures for SER have undergone significant changes. In 2014, Mao, Q et al [7] used convolutional neural network (CNN) [10] to learn emotion-related features in speech, and experimentally demonstrated that using CNN to learn features in speech can achieve stable and strong recognition performance in complex scenarios to achieve stable and robust recognition performance. In 2017, Satt, A et al [11] used CNN, Long Short-Term Memory Network (LSTM) [5] to learn directly from raw spectrograms and tested both CNN and LSTM, the ablation experiments in the article show that models using a fusion of CNN and LSTM are more effective in SER than models using only CNNs without LSTMs, and finally concluded that three convolutional layers plus one LSTM layer gave the best results. And then, in 2019, Li, Y et al [6] And with the development of deep learning technology, LChen, L et al cite(L.-W. Chen and A. Rudnicky, "Exploring wav2vec 2.0 fine tuning for improved speech emotion recognition", ICASSP 2023–2023 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP),) inspired by task-adaptive pretraining (TAPT) [4] proposed a pseudo-convolutional task-adaptive pretraining (P-TAPT) method based on the improvement of TAPT, and used this method alongside Vanilla Fine-Tuning (V-FT) and TAPT for Wav2Vec 2.0 [1] fine-tuning as a comparison, it is proved that the proposed P-TAPT method effectively fine-tune Wav2Vec 2.0 making the model perform better on the SER task; Cai et al [2].

III. DATASET DESCRIPTION

Because of the limited availability of resources for Badaga language, we generate a speech dataset custom for this work. The dataset does not consist of open source resources, and was built from the previous materials using the collaboration of native speakers and language experts.

A. *Dataset Characteristics*

The corpus consists of 9,837 audio files of spoken Badaga each accompanied by:An English translation (translated_transcript), A phonetic transcription in Latin characters (translterated_script), and Speaker information including user id, gender, and locale.The audio files were recorded in the field within a controlled environment using mobile recording equipment with native speakers. The recordings were coded in a manual annotation process and verified by fluent native speaker to assure a quality product.

B. Data Splits

The dataset is categorized as follows:

- **Training set:** 6,897 samples
- **Validation set:** 1,470 samples
- **Test set:** 1,470 samples

This arrangement ensures that evaluation between the models does not fluctuate or show any bias.

C. Ethical Issues

All participants signed informed consent releases prior to making any recordings. The dataset is not available publicly due to ethical and privacy considerations. However, it may be made available upon request for academic research.

D. Data Composition

TABLE I
Badaga Speech Dataset Composition

Attribute	Value
Total samples	9,837
Total speakers	11
Gender distribution	4,929 Female / 4,908 Male
Locale	`ba` (Badaga)
Average duration	2.41 seconds
Duration range	0.29s – 13.24s
Average word count	5.13 words
Word count range	1 – 18 words
Missing transliterations	3 (<0.03%)

IV. WORK FLOW

The Badaga ASR process begins with the gathering of audio data from native speakers, forming the basis of the dataset. This is then followed by model selection, which looks at two self-supervised architectures: Wav2Vec 2.0 and HuBERT. Both models will be preprocessed and trained on the Badaga data. Following training, they are evaluated separately on a test set, where Word Error Rate (WER) is the primary evaluation metric. The final step is to compare the two models to determine which was more effective in recognizing speech from this low-resource language.

V. WAV2VEC2

Wav2Vec 2.0 is a cutting-edge self-supervised deep learning technique from Facebook AI (formerly Meta AI) that is capable of training end-to-end speech representations end-to-end from raw audio waveforms. While in the past, ASR systems have depended on hand-crafted features such as MFCCs or spectrograms, Wav2Vec 2.0 is capable of learning strongly contextualized representations with fewer dependencies on

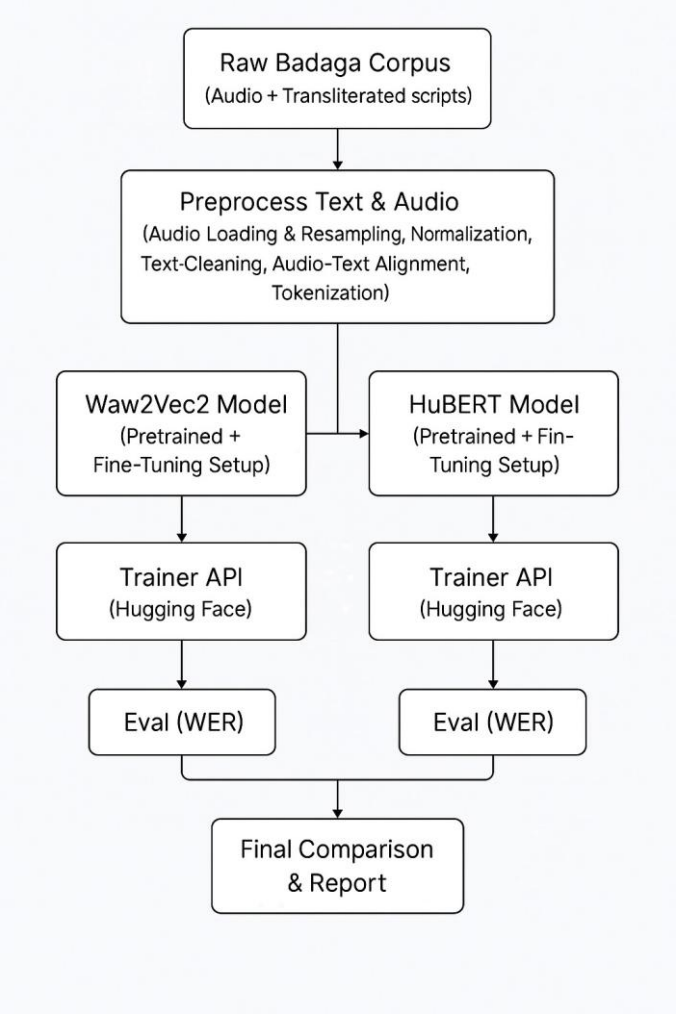

Fig. 1. workflow.

ginormous labeled datasets and is of great assistance when applied to low-resource languages.

A. Architecture Overview Wav2Vec 2.0 consists of three main elements:

Feature Encoder: It takes raw waveform audio and maps it to a sequence of latent speech representations through a temporal convolutional layer stack. Representations capture short-term acoustic features of the waveform.

Context Network: The output of feature encoder is fed to a Transformer-based context network. The network can capture long-term temporal patterns and produce contextualized representations after listening to the entire sequence of audio.

Quantization Module (pretraining): Pre-training has a quantization module projecting latent representations onto a fixed codebook of entries. Training of the model is done to project these quantized representations from context output with contrastive learning objectives.

B. Self-Supervised Pretraining and Fine-Tuning There are two processes pre-training the Wav2Vec 2.0 model: **Pre-training:** The Wav2Vec 2.0 model is pre-trained on an unlabeled speech corpus and can learn how to recognize the correct quantized representations of a set of negative samples. This enables the model to learn dense phonetic and acoustic detail in an unsupervised manner.

Fine-Tuning: Following pretraining, there is a linear layer on top of the Transformer model and the model fine-tuned on the fully tagged speech-to-text pairs. The tokens output are usually subword units or characters and thereby place the model in position for use with character-level ASR systems.

C. Benefits in Low-Resource Environments Wav2Vec 2.0 is observed to perform very well in under-resourced tagged speech domains. With even a time lag of 10 minutes to several hours of tagged speech, fine-tuning using the pre-trained model can achieve competitive performance. It is hence a suitable model to apply in under-resourced languages such as Badaga with hardly any tagged speech data available.

Also, training Wav2Vec 2.0 directly on raw audio itself with direct impact literally streamlines the ASR pipeline to a process and removes the need for extensive feature engineering. Its end-to-end architecture also allows joint optimization of all model parameters as one system and thus is more accurate and robust too.

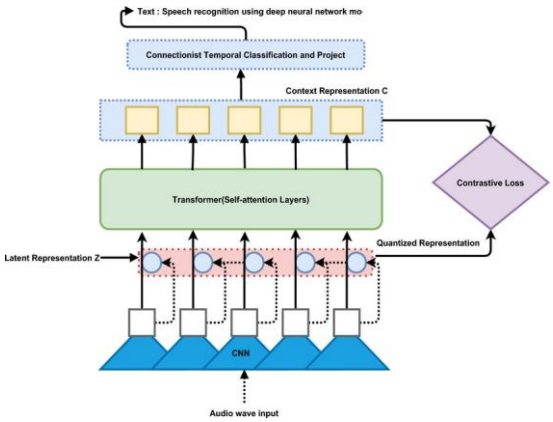

Fig. 2. architecture of Wav2vec 2.0 model.

VI. HUBERT

HuBERT (Hidden-Unit BERT), developed by Facebook AI, is a self-supervised learning model for speech representation. Instead of applying contrastive learning (as is done with Wav2Vec 2.0), HuBERT adopts a masked prediction method using discrete units derived completely from the audio signal. HuBERT can therefore learn high-quality speech representations without requiring any manual transcriptions during pretraining. This ability makes HuBERT potentially applicable in an under-resourced language like Badaga.

A. Overview of Architecture

HuBERT is based on three fundamental components:

Feature Encoding Stage: The raw audio waveform is first transformed into latent speech representations by successively applying a stack of convolutional layers. These representations use low-level acoustic information and pass these features into a transformer.

Transformer Encoding Stage: The latent features serve as the input to a deep transformer network. As in BERT NLP, some of the time steps in the input are masked and the model is subsequently trained to predict the missing time step based on the context. This assists in learning temporal features, patterns, and dependencies.

Prediction Head (Cluster Targets): HuBERT does not employ phoneme or character labels in supervision. Rather, it uses cluster labels for 'hidden units,' derived from clustering the speech features from a separate model (e.g., k-means). The model is trained to predict the cluster label at the masked time steps in the input.

B. Self-Supervised Pretraining and Fine-Tuning

Pretraining: The model is trained using large volumes of unlabeled audio. An offline clustering stage is performed to create discrete units that behave like phonemes. While training, portions of audio are masked, and HuBERT is trained to predict the cluster label for the masked section using context from the unmasked portions.

The training objective is to minimize the cross-entropy loss between the predicted probability distribution $\hat{y}_i$ and the true cluster label y_i over the masked time steps:

$$L_{\text{CE}} = -\sum_{i=1}^{N} y_i \log(\hat{y}_i)$$

Where:

- y_i: One-hot encoded ground truth cluster label
- $\hat{y}_i$: Predicted probability distribution for cluster labels
- N: Number of masked time steps

This unsupervised approach helps the model capture phonetic and acoustic structure from speech without labeled data.

Fine-Tuning: After the pretraining stage, HuBERT is fine-tuned on a downstream ASR task by adding a linear layer to predict character or subword tokens from the output of the Transformer. Fine-tuning is performed using a supervised set of audio-text pairs.

Here, HuBERT uses the Connectionist Temporal Classification (CTC) loss function, which enables the model to align input audio frames to output text sequences without requiring aligned frame-level labels:

$$L_{\text{CTC}} = -\log p(y|x)$$

Where:

- x: Input sequence of latent audio features
- y: Target output text sequence
- $p(y|x)$: Sum over all valid alignments between x and y

C. Prospects in Low-Resource Contexts

HuBERT is a particularly valuable tool in low-resource contexts. Since it learns rich speech representations without needing any labeled data during pretraining, it is an ideal fit for languages like Badaga that lack large annotated corpora. Training with pseudo-labels from other language data alongside clustering allows the model to transfer well to new languages when combined with small labeled datasets during fine-tuning.

In this project, HuBERT was pretrained with publicly available multilingual speech data and fine-tuned with our own custom Badaga dataset. The HuBERT architecture, which learns to predict phoneme-like units from masked input representations, is ideal for addressing speech recognition challenges in a low-resourced language setting.

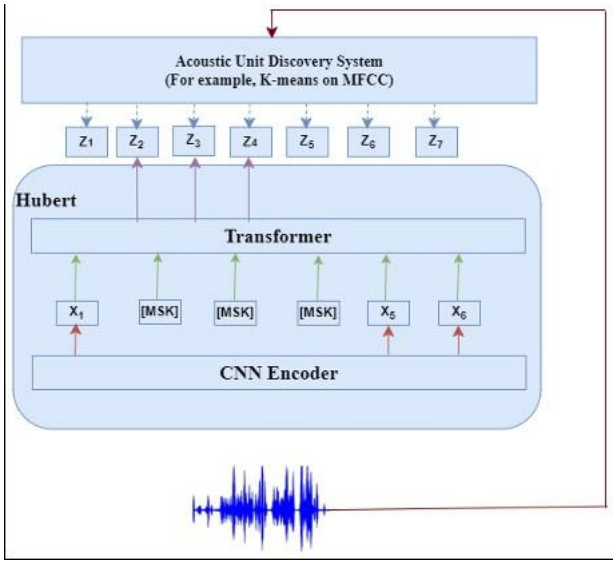

Fig. 3. architecture Hubert.

VII. METHODOLOGY

1. **Data Collection:** Our ASR project is based on a tailored data set generated entirely from recordings of native speakers of Badaga language. We managed to collect a strong set of 9,837 audio samples across 11 speakers as it was important to have variation in gender and speaking style. Each audio file has information to accompany it:

The **English transcript**.

A **transliterated Badaga**.

Speaker information (User ID, Gender, Locale).

Split labels (Train, Validation, Test).

Duration in seconds.

This data set is not publicly available because of the ethical aspect. Consent forms were signed by all participants, and we will ask for a non-disclosure agreement (NDA) with any data we share.

2. **Data Preprocessing**: In order to prepare the data for ingestion to the model, several preprocessing steps were taken:

Resampling: All audio was resampled to 16 kHz, the default sampling rate for most pre-trained ASR models (Wav2Vec 2.0 and HuBERT). This makes it consistent and compatible with their feature extractors.

Trimming Silence: Leading and trailing silences were trimmed out of each audio file through the use of energy-based thresholding. Constraining how the models learn from the data prevents them from learning long irrelevant silences often

present in the audio, which can help with the convergence of training.

Amplitude Normalization: Waveforms were all normalized to have a common range, usually [1,1][1,1]. Normalizing the audio can not only help prevent extreme values from impacting the model but it can also help generalization. This will moderate any bias the model may have towards the louder/softer samples.

Tokenization: Due to the fact that Badaga has no standardized orthography, character-level tokenization was implemented: Each transcript will be tokenized into a sequence of characters.A vocabulary file will be created mapping each unique character to an integer ID.

3. Model Selection and Training

Two self-supervised models pretrained on large speech corpora were fine-tuned for the Badaga ASR task: **Wav2Vec 2.0** and **HuBERT**.

A. Wav2Vec 2.0 Architecture

Wav2Vec 2.0 is composed of three stages. First, a convolutional encoder processes raw waveform input to extract latent speech representations. These representations are passed to a Transformer-based context network that models long-range dependencies and provides contextual embeddings during self-supervised training.

During fine-tuning for ASR, a linear projection layer is added on top of the contextual representations to map them to character-level outputs. The model is trained using the Connectionist Temporal Classification (CTC) loss function, which enables alignment-free training of audio and text sequences.

CTC Loss: The CTC loss is defined as:

$$L_{\text{CTC}} = -\log \sum_{\pi \in B^{-1}(y)} P(\pi \mid x)$$

Where:

x = input audio features,
y = target transcription,
π = all valid alignments (paths),
$B^{-1}(y)$ = all possible paths that collapse to y.

***Hyperparameters**:* The following hyperparameters were used for fine-tuning:

Epochs: 10
Learning Rate: 3×10^{-5}
Optimizer: AdamW
Warmup Steps: 500
Loss Function: CTC Loss
Evaluation Metric: Word Error Rate (WER)

B. HuBERT (Hidden-Unit BERT)

Architecture:HuBERT employs unsupervised clustering on speech features to produce pseudo-labels. Rather than predicting actual waveform samples (as in Wav2Vec), the model learns to predict masked cluster IDs.

This enables HuBERT to better learn structures in the data, which is critical when working with low-resource languages.

Training:The HuBERT model is first pretrained using a masked prediction objective, followed by fine-tuning on the labeled data and a shared CTC loss. The same optimizer and scheduler used across the example groups and Wav2Vec 2.0 are applied meaning that groups will see similar patterns in testing.

***Hyperparameters**:* The following hyperparameters were used for fine-tuning the HuBERT model:

Epochs: 10
Learning Rate: 2×10^{-5}
Optimizer: AdamW
Mask Probability: 0.065
Batch Size: 8
Evaluation Metric: Word Error Rate (WER)

4. Evaluation

Evaluation was conducted using the **Word Error Rate (WER)**, a conventional metric for Automatic Speech Recognition (ASR) performance. It is defined as:

$$\text{WER} = \frac{S + D + I}{N} \tag{1}$$

where:

S = Number of substitutions
D = Number of deletions
I = Number of insertions
N = Total number of words in the reference transcript

Lower values of WER indicate improved transcription performance.

5. Comparing Performance

The WERs from each model on the test split were compared to determine relative performance. Qualitative analysis was also conducted, focusing on types of errors such as:

Frequent substitutions of similar-sounding words
Omissions or insertions of short functional words
Speaker-specific or gender-related inconsistencies

The model with the lowest WER and greater consistency across speaker diversity and gender variations was selected as the most suitable ASR model for the Badaga language.

VIII. RESULT

The metrics used to evaluate the performance of both models was the Word Error Rate (WER). Wav2Vec 2.0 achieved a WER of 16.8, while HuBERT slightly outperformed it with a WER of 16.5. [table 11 and table 111] This indicates that overall transcription accuracy was better with HuBERT. HuBERT appears to have an advantage because of its use of masked prediction and cluster-based targets, allowing it to better learn phonetic representations—particularly useful in a lower-resourced language like Badaga.

TABLE II

HUBERT TRAINING METRICS AT VARIOUS STEPS

Epoch / Step	Training Loss	Validation Loss	WER
100	2.89	2.81	1.00
1000	1.00	0.70	0.46
3000	0.48	0.47	0.28
5000	0.44	0.40	0.22
7000	0.29	0.35	0.20
9000	0.30	0.35	0.17
10400	0.29	0.35	0.16

TABLE III

WAV2VEC2 TRAINING METRICS AT VARIOUS STEPS

Epoch / Step	Training Loss	Validation Loss	WER
100	3.78	3.36	1.00
1000	1.15	0.77	0.56
3000	0.51	0.48	0.31
5000	0.43	0.40	0.25
7000	0.30	0.35	0.21
9000	0.27	0.34	0.18
10400	0.31	0.32	0.17

D I S

XI. DISCUSSION

The evaluation of HuBERT and Wav2Vec2 models to compare their performances was through the lens of training loss, validation loss, and Word Error Rate (WER) through multiple training steps. Both models improved considerably over training; however, it was HuBERT that exhibited the best WER of 0.16 at step 10400 compared to Wav2Vec2's WER of 0.1687. HuBERT also had lower training loss values across most steps, indicating it was developing more efficient internal representations during optimization. Wav2Vec2 did have a marginally lower validation loss in the later-stage iterations; this can indicate that Wav2Vec2 generalized better on the validation set than HuBERT since their overlapping validation losses were achieved through differing amounts of training clearing validating HuBERT's competitive validation losses.

However, while Wav2Vec2 produced lower validation loss its WER still remained higher than HuBERT's indicating that HuBERT demonstrated both superior performance on the training set leading to improved learning during training steps and ultimately outperformed Wav2Vec2 by demonstrating greater ability to extract phonetic and contextual information important to the task and complete recognition for the final transcription.

To conclude, both models produce strong ASR task output; however, HuBERT produced better final recognition accuracy on the dataset and could be considered the better choice for the given dataset.

IX. CONCLUSION AND FUTURE WORK

The present paper very effectively showcases the employment of self-supervised learning approaches—HuBERT and Wav2Vec 2.0—to train the Automatic Speech Recognition (ASR) model for low-resource Badaga language specifically. With extreme caution being exercised while recording data from native speakers and fine-tuning pre-trained models, we were able to achieve good transcription quality. Both models had reasonable capacity to learn informative representations from small sets of labeled data, HuBERT being slightly better than Wav2Vec 2.0 in Word Error Rate (WER), the total WER of 16.5 being slightly better than Wav2Vec 2.0 at 16.8.

The results confirm the ability of HuBERT to learn the phonetic structure through masked prediction and clustering-based training and is thus a good candidate for low-resource languages. Wav2Vec 2.0 was also resilient and this speaks well for the potential of self-supervised speech technology in difficult linguistic conditions.

Data Augmentation: Since there are more other dialect speakers and older speakers in the Badaga dataset, and therefore more size and diversity, it can also be utilized to improve model generalizability. Multilingual Transfer Learning: Multilingual pre-trained model or ensemble of similar language dataset can be utilized to improve model performance and diversity for Badaga. Speaker Adaptation: Defining speaker adaptation methods to facilitate adaptation of the models to specific speech variations can aid in WER reduction, especially with real-world application. Model Ensemble: Ensembling HuBERT and Wav2Vec 2.0 predictions to produce strong transcription output. Real-time Inference: Execution of the models on embedded or mobile hardware would allow for real-time ASR between the Badaga speakers. Language Model Integration: Pre-trained Badaga-specific language model integration will enhance syntactic and semantic transcription accuracy. The project is a stepping stone to future projects, which seek to transfer technology to ASR in low-resource and endangered languages to linguistic diversity preservation through technoscientific innovation

X. 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*, volume 33, 2020.

[2] Xiong Cai, Jie Yuan, Rui Zheng, Lirong Huang, and Kenneth Church. Speech emotion recognition with multitask learning. In *Interspeech*, pages 4508–4512, 2021.

[3] Christopher Cieri, David Miller, and Kevin Walker. The fisher corpus: a resource for the next generations of speech-to-text. In *LREC*, 2004.

[4] Suchin Gururangan, Ana Marasovic´, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. Don't stop pretrain- ing: Adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*, 2020.

[5] Sepp Hochreiter and Ju¨rgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[6] Y. Li, T. Zhao, and T. Kawahara. Improved end-to-end speech emotion recognition using self attention mechanism and multitask learning. In *Interspeech*, 2019.

[7] Qirong Mao, Ming Dong, Zhi Huang, and Yongzhao Zhan. Learning salient features for speech emotion recognition using convolutional neural networks. *IEEE Transactions on Multimedia*, 16:2203–2213, 2014.

[8] Ghulam Muhammad. Automatic speech recognition using interlaced derivative pattern for cloud based healthcare system. *Cluster Computing*, 18(2):795–802, 2015.

[9] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210. IEEE, 2015.

[10] Alexander Rakhlin. Convolutional neural networks for sentence classi- fication. GitHub, 2016.

[11] Amos Satt, Shai Rozenberg, and Ron Hoory. Efficient emotion recogni- tion from speech using deep learning on spectrograms. In *Interspeech*, pages 1089–1093, 2017.

[12] R. Singh, H. Yadav, M. Sharma, S. Gosain, and R. R. Shah. Automatic speech recognition for real time systems. In *2019 IEEE Fifth Interna- tional Conference on Multimedia Big Data (BigMM)*, pages 189–198. IEEE, 2019.