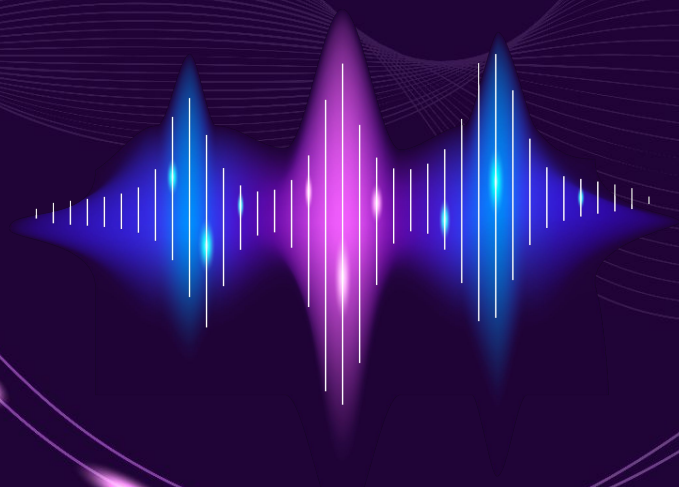
# EXPLORING SELF SUPERVISED SPEECH MODELS FOR LOW-RESOURCE ASR: A CASE STUDY ON BADAGA USING HuBERT AND Wav2Vec2

GROUP - 8

# PROBLEM STATEMENT

- Automatic Speech Recognition (ASR) converts spoken language into text using AI.
- Badaga is a Dravidian language with minimal digital speech resources.
- Developing ASR for Badaga is difficult due to limited transcribed data.
- Traditional models rely on handcrafted features and large datasets.
- HMM-based approaches perform poorly on low resourced languages.



# MOTIVATION AND RELEVANCE

- Badaga has very little research in speech recognition.
- The language is not well represented in digital tools.
- Using ASR helps in preserving and promoting the language.
- Self-supervised models work well even with small datasets.

# OBJECTIVE

- ➔ To build an efficient ASR model for the Badaga language using deep learning.
- ➔ To fine-tune self-supervised models like HuBERT and Wav2Vec2 for Badaga speech.
- ➔ To evaluate model performance using Word Error Rate (WER) as the main metric.
- ➔ To explore challenges in adapting ASR for low-resource language environments.

# LITERATURE REVIEW

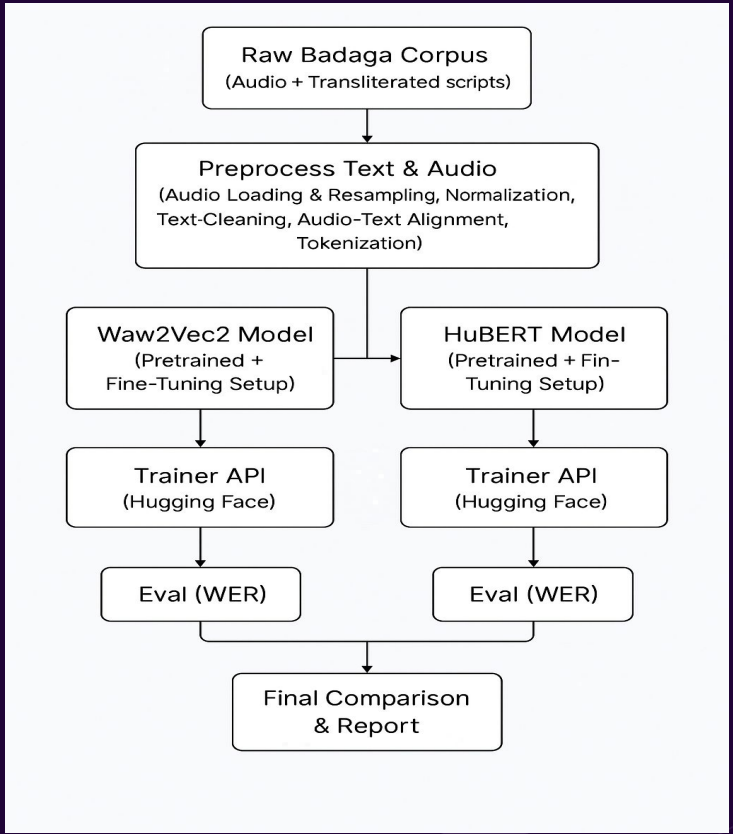
| Model Description | Dataset | WER (%) | Source |
|---|---|---|---|
| RNN combined with Connectionist Temporal Classification (CTC) | Wall Street Journal (WSJ) | 30.1 | https://www.mdpi.com/2073-8994/11/8/1018#B27-symmetry-11-01018 - An Overview of End-to-End Automatic Speech Recognition |
| CNNs and Transformer-based self-attention | Not specified | 2.1/4.3 | https://arxiv.org/abs/2005.08100 Convolution-augmented Transformer for Speech Recognition |
| Tamil ASR model based on Mozilla DeepSpeech architecture | Not specified | 55 | https://cdn.techscience.cn/ueditor/files/iasc/TSP_IASC-32-2/TSP_IASC_22021/TSP_IASC_22021.pdf |
| Direct Speech-to-Text Translation | Not specified | N/A | https://arxiv.org/pdf/2306.11646 - "Recent Advances in Direct Speech-to-Text Translation“ |
| CNN model | Low-resource Turkic languages with common alphabets | 17.4 | https://www.nature.com/articles/s41598-024-64848-1- Multilingual end-to-end ASR for low-resource Turkic languages with common alphabets |

| Model Description | Dataset | WER (%) | Source |
|---|---|---|---|
| ASR for Low-Resource Phonetic Languages Output Alphabet Reduction and Reconstruction Module; | Mvskoke | 7 | ttps://aclanthology.org/2024.acl-srw.16.pdf- ASR for Low-Resource Phonetic Languages Output Alphabet Reduction and Reconstruction Module Mvskoke language |
| Fine-Tuning ASR Models for Very Low-Resource Languages; HMM and E2E | Not specified | 24.7 | https://aclanthology.org/2024.acl-srw.16.pdf- Fine-Tuning ASR Models for Very Low-Resource Languages HMM and E2E |
| Whisper multilingual ASR model Cross-entropy for Low-Resource Languages in ASR | Low-Resource Languages | 3.29 | . https://www.arxiv.org/abs/2409.16954 - Whisper multilingual ASR model Cross-entropy for Low-Resource Languages in ASR. |
| Direct Speech-to-Text Translation | Not specified | N/A | https://arxiv.org/pdf/2306.11646 - "Recent Advances in Direct Speech-to-Text Translation“ |

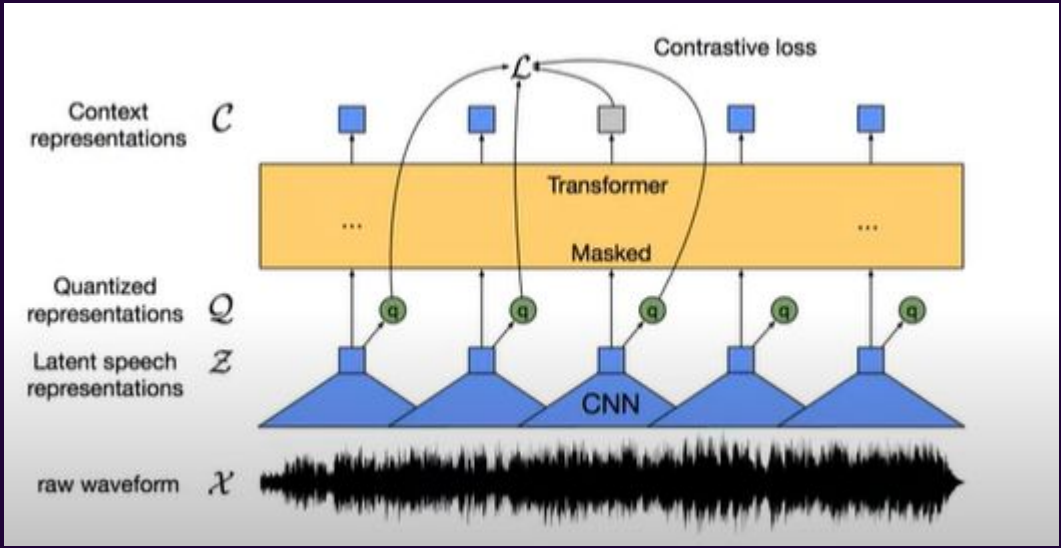
# DESCRIPTION OF DATASET

| | |
|---|---|
| Total Samples | 9,837 transcribed audio clips in the Badaga language. |
| Speaker Diversity | Data includes recordings from 11 unique users. |
| Gender Distribution | Balanced with 4,929 female and 4,908 male speakers. |
| Audio Duration | Average duration is 2.4s, with a max of 13.2s. |
| Split | Dataset is divided into 6,897 training and 1,470 testing ,1470 validation samples. |
| Structure | Contains 9 columns (translated,transliterated scripts,audio file name,gender,duartion |

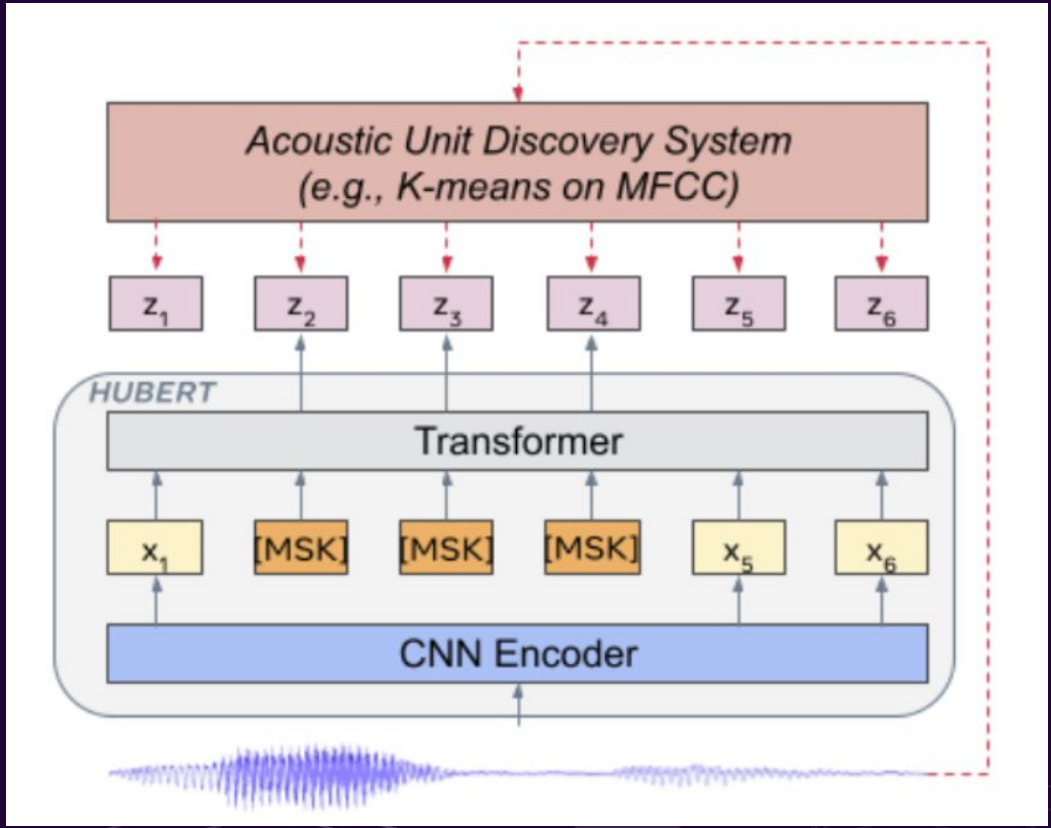
# WORKFLOW


Raw Badaga Corpus
(Audio + Transliterated scripts)
Preprocess Text & Audio
(Audio Loading & Resampling, Normalization, Text-Cleaning, Audio-Text Alignment, Tokenization)
Waw2Vec2 Model
(Pretrained + Fine-Tuning Setup)
HuBERT Model
(Pretrained + Fin-Tuning Setup)
Trainer API
(Hugging Face)
Trainer API
(Hugging Face)
Eval (WER)
Eval (WER)
Final Comparison & Report

# ARCHITECTURE - Wav2Vec2



# ARCHITECTURE - HuBERT



# IMPLEMENTATION - Wav2Vec2

1. Data Preparation - Collected Badaga audio-text pairs and cleaned transcriptions by removing special characters.

2. Vocabulary Creation - Built a JSON vocabulary mapping characters (A–Z, space, special tokens like `<pad>`, `<unk>`, `|`) to IDs.

3. Tokenizer Setup - Initialized `Wav2Vec2CTCTokenizer` using the custom vocabulary for converting text to token IDs.

4. Feature Extraction - Used `Wav2Vec2FeatureExtractor` to process raw audio into normalized, padded feature arrays.

5. Processor Integration - Combined tokenizer and feature extractor into a `Wav2Vec2Processor` to handle both audio and text.

6. Dataset Preprocessing - -Applied a `prepare_dataset()` function using `map()` to extract model-ready inputs and labels from the dataset.

# IMPLEMENTATION - Wav2Vec2

**7. Data Collator**: Pads sequences dynamically and replaces padding labels with -100 to ignore loss.

**8. Metric Calculation**: Uses Word Error Rate (WER) to evaluate model performance, ignoring padding tokens.

**9. Model Setup**:Loads the Wav2Vec2 model, configures dropout rates (`attention_dropout`, `hidden_dropout`), and moves it to GPU.

Main parameters:
`attention_dropout=0.1,hidden_dropout=0.1,feat_proj_dropout=0.0,mask_time_prob=0.05,layerdrop=0.1,`

**10.Training Configuration**: ets batch sizes, learning rate, number of epochs, and evaluation strategy using `TrainingArguments`.

Main parameters: `per_device_train_batch_size=4`, `learning_rate=3e-4`, `num_train_epochs=10`, `save_steps=100`, `logging_steps=10`.

# IMPLEMENTATION- Hubert

1.**Data Preparation**: The dataset consists of Badaga speech recordings along with transliterated transcripts.

Missing values were removed from the data.

Audio file paths were updated to point to the correct directory.

Data was split into training and testing sets based on a split_label.

2.**Audio Loading**: Used `librosa` to load audio at a sampling rate of 16kHz.

Audio files were converted into arrays and stored along with their sampling rates and paths.

Created structured dictionaries for both training and testing data.

3. **HuBERT Pretraining Approach: Step A: Discover Hidden Units (Unsupervised Learning)**

- **MFCC Features Extraction**:
    - Raw audio was converted into Mel-Frequency Cepstral Coefficients (MFCCs), which represent how sound is perceived by the human ear.

# IMPLEMENTATION

**K-means Clustering**:

- MFCC vectors from the entire dataset were pooled together.
- K-means was applied to group similar sound segments into clusters.
- Each audio frame was assigned a cluster ID, which acts as a "pseudo-label" or hidden unit.

**Step B: Predict Hidden Units (Self-Supervised Learning):**

- Audio is passed through a convolutional feature encoder to extract latent features.
- Random parts of the features are masked (similar to BERT).
- These masked features are passed through a Transformer network to generate contextual representations.
- The model learns to predict the hidden units (cluster IDs) for the masked sections.

## 4. Fine-Tuning on Badaga Dataset

- Used the pretrained `facebook/hubert-large-ls960-ft` model.
- Fine-tuned using the Badaga audio and transcript data.
- Loss function used: CTC (Connectionist Temporal Classification), ideal for speech-to-text without word alignment.

**5. Evaluation:** using **Word Error Rate (WER).** which measures the percentage of words incorrectly transcribed.

# RESULTS - Wav2Vec2

| EPOCH/STEP | TRAINING LOSS | VALIDATION LOSS | WER | NOTES |
|---|---|---|---|---|
| 100 | 3.78 | 3.35 | 1.000 | Initial high error,as expected |
| 1000 | 1.15 | 0.77 | 0.56 | Sharp improvement |
| 3000 | 0.50 | 0.47 | 0.31 | Continue to improve |
| 5000 | 0.43 | 0.39 | 0.24 | Good balance between training and validation performance |
| 7000 | 0.30 | 0.35 | 0.21 | Stabilizing phase |
| 9000 | 0.26 | 0.33 | 0.17 | Great performance |
| 10400 | 0.30 | 0.32 | 0.168 | Final step — well trained |

# RESULTS - HuBERT

| EPOCH/STEP | TRAINING LOSS | VALIDATION LOSS | WER | NOTES |
|---|---|---|---|---|
| 100 | 2.89 | 2.81 | 1.000 | Initial high error,as expected |
| 1000 | 1.00 | 0.70 | 0.46 | Sharp improvement |
| 3000 | 0.48 | 0.47 | 0.28 | Continue to improve |
| 5000 | 0.44 | 0.40 | 0.22 | Good balance between loss and WER |
| 7000 | 0.29 | 0.35 | 0.20 | Stabilizing phase |
| 9000 | 0.30 | 0.35 | 0.17 | Great performance |
| 10400 | 0.29 | 0.35 | 0.165 | Final step — well trained |

# INFERENCE - Wav2Vec2

- ➔ Effectively learns speech representations directly from raw waveform inputs without the need for handcrafted features.
- ➔ Performs well even with limited annotated data, making it suitable for low-resource language scenarios.
- ➔ Achieved a Word Error Rate (WER) of 0.168, demonstrating strong recognition capabilities.
- ➔ Offers efficient training and generalization across various speech tasks.

# INFERENCE - HuBERT

- Utilizes unsupervised clustering of acoustic units to guide the pretraining, enhancing feature learning.
- Demonstrates superior robustness in handling noisy and diverse audio data.
- Outperformed Wav2Vec2 with a slightly lower WER of 0.165, indicating better accuracy.
- Exhibits consistent and stable performance across different evaluation metrics and steps.

# CONCLUSION

- ➔ HuBERT achieved a lower WER (0.1608) than Wav2Vec2 (0.1687), indicating better performance.
- ➔ Self-supervised learning proved effective for ASR in low-resource languages.
- ➔ HuBERT's hidden unit prediction provided superior speech representation learning.
- ➔ Both models showed consistent improvement and reduced WER across epochs.
- ➔ Pre-trained models enable scalable ASR systems with minimal labeled data.

# REFERENCES


- R. Jain, A. Barcovschi, M. Y. Yiwere, D. Bigioi, P. Corcoran and H. Cucu, "A WAV2VEC2-Based Experimental Study on Self-Supervised Learning Methods to Improve Child Speech Recognition," in *IEEE Access*, vol. 11, pp. 46938-46948, 2023, doi: 10.1109/ACCESS.2023.3275106.
- N. Vaessen and D. A. Van Leeuwen, "Fine-Tuning Wav2Vec2 for Speaker Recognition," *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Singapore, Singapore, 2022, pp. 7967-7971, doi: 10.1109/ICASSP43922.2022.9746952
- R. Shankar, K. Tan, B. Xu and A. Kumar, "A Closer Look at Wav2vec2 Embeddings for On-Device Single-Channel Speech Enhancement," *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Seoul, Korea, Republic of, 2024, pp. 751-755, doi: 10.1109/ICASSP48485.2024.10447539
- A. Akhilesh, B. P, K. S, D. Gupta and S. Vekkot, "Tamil Speech Recognition Using XLSR Wav2Vec2.0 & CTC Algorithm," *2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Kharagpur, India, 2022, pp. 1-6, doi: 10.1109/ICCCNT54827.2022.9984422.

# REFERENCE


- O. Ludwig and T. Claes, "Compressing Wav2vec2 for Embedded Applications," *2023 IEEE 33rd International Workshop on Machine Learning for Signal Processing (MLSP)*, Rome, Italy, 2023, pp. 1-6, doi: 10.1109/MLSP55844.2023.10285964.
- F. Javanmardi, S. R. Kadiri and P. Alku, "Exploring the Impact of Fine-Tuning the Wav2vec2 Model in Database-Independent Detection of Dysarthric Speech," in *IEEE Journal of Biomedical and Health Informatics*, vol. 28, no. 8, pp. 4951-4962, Aug. 2024, doi: 10.1109/JBHI.2024.3392829

A WAV2VEC2-Based Experimental Study on Self-Supervised Learning Methods to Improve Child Speech Recognition

wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations

# Thanks!