Universitat Autònoma de Barcelona

Amplifying Voices: Self-Supervised Learning for Low-Resource and Minority Languages

Authors

Alex Roldan, Alex Sanchez, Jordi Longaron

GitHub:
https://github.com/alrocb/Amplifying-Voices-Advanced-Machine-Learning

Teacher
Andrey Barsky

Year
2024/2025

Contents

1 Introduction

1.1 Motivation and Problem Statement

Many languages worldwide face marginalization or extinction due to limited technological resources. As these languages fade, so too does their cultural heritage. Automatic Speech Recognition (ASR) systems can help preserve and support minority languages by improving accessibility and inclusion. However, developing ASR systems for low-resource languages like Catalan is challenging due to the scarcity of annotated datasets.

Self-supervised learning (SSL) models, such as Wav2Vec 2.0, address this challenge by leveraging unlabeled data to learn robust speech representations, significantly reducing the need for extensive labeled datasets. Catalan's linguistic similarity to Spanish offers an opportunity to investigate whether pretraining on a related language can aid adaptation. Conversely, pretraining on English, a less related language, provides a baseline to assess the impact of linguistic relatedness.

1.2 Research Questions and Objectives

This work centers around three key research questions:

1. How does a Wav2Vec 2.0 model trained from scratch in Catalan compare to models pretrained on Spanish or English?

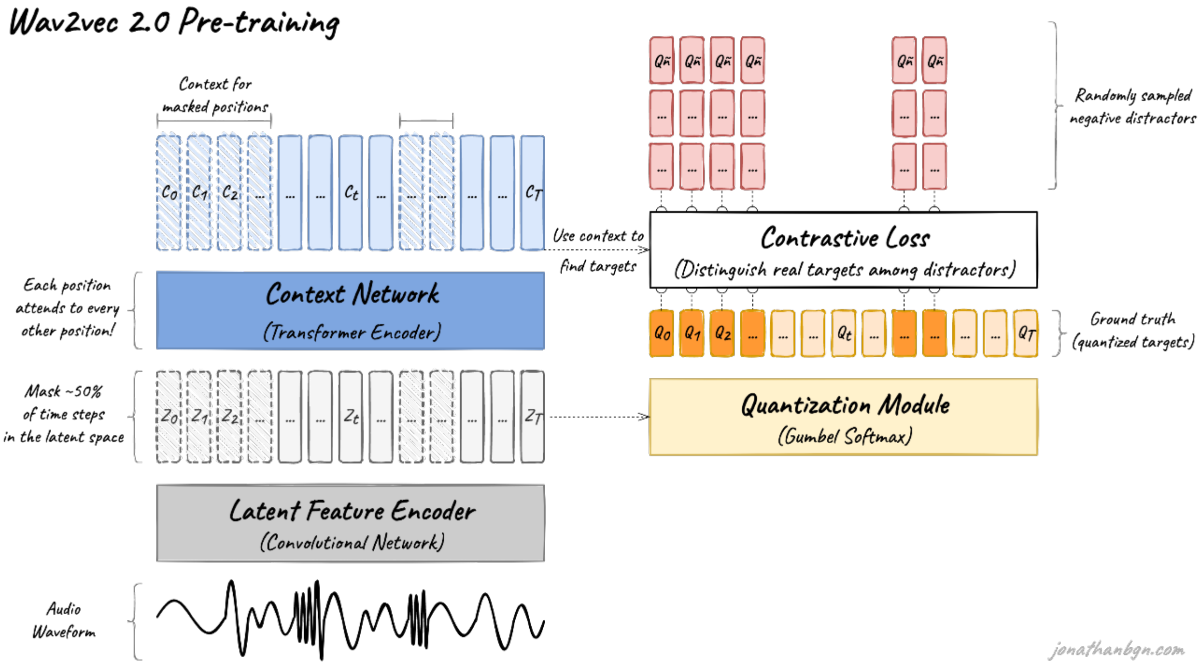
2. Does linguistic similarity enhance model adaptation, as seen between Spanish and Catalan?

3. Can self-supervised representations support additional tasks like accent and gender recognition?

1.3 Contributions

This work demonstrates the feasibility of training Wav2Vec 2.0 models for Catalan, evaluates the influence of linguistic similarity on model adaptation, and highlights the versatility of SSL embeddings for related tasks. Techniques like Grad-CAM provide interpretability, enhancing trust in these systems for low-resource languages. The findings advocate for leveraging multilingual training and SSL to support underrepresented languages, promoting cultural preservation and technological inclusivity.

2 Explaining the used model


Wav2vec 2.0 Pre-training
Context for masked positions
C_0 C_1 C_2 ... ... C_t ... ... C_T
Each position attends to every other position!
Context Network
(Transformer Encoder)
Mask ~50% of time steps in the latent space
Z_0 Z_1 Z_2 ... ... Z_t ... ... Z_T
Latent Feature Encoder
(Convolutional Network)
Audio Waveform
Use context to find targets
$Q_{\tilde{n}}$ $Q_{\tilde{n}}$ $Q_{\tilde{n}}$ $Q_{\tilde{n}}$ $Q_{\tilde{n}}$ $Q_{\tilde{n}}$
Randomly sampled negative distractors
Contrastive Loss
(Distinguish real targets among distractors)
Q_0 Q_1 Q_2 ... ... Q_t ... ... Q_T
Ground truth (quantized targets)
Quantization Module
(Gumbel Softmax)
jonathanbgn.com

1. **Input Audio**: The model begins with raw audio waveforms as input, which are passed through a Latent Feature Encoder. This convolutional network extracts low-level acoustic features, converting the raw audio into a sequence of latent representations.
2. **Masking**: Approximately 50-60% of the latent features are randomly masked, creating gaps that the model must learn to fill. This resembles the masking process used in natural language models like BERT.
3. **Context Network**: These masked latent representations are passed to a Context Network (Transformer Encoder). This component leverages self-attention mechanisms to understand the temporal relationships and context within the speech.
4. **Quantization Module**: The latent features are also passed through a Quantization Module that discretizes them into discrete codewords using Gumbel-Softmax. This creates fixed vocabulary-like representations from the continuous latent space.
5. **Contrastive Loss**: The model is trained with a contrastive loss, which requires it to distinguish the correct quantized target (from the masked positions) from a set of distractor samples. Simultaneously, a diversity loss ensures the quantization module uses a wide range of its available codewords.

This pretraining process enables the model to learn meaningful speech representations from unlabeled audio, which can later be fine-tuned for specific tasks like Automatic Speech Recognition (ASR).

For more details, consult annex A.1

3 Methodology and Baseline

3.1 Explaining Data, Labelled Data Preparation and Baseline model:

Common Voice Dataset and Its Relevance:

The Common Voice dataset by Mozilla is a community-driven project that collects diverse speech recordings from volunteers worldwide. Each language dataset includes audio clips, corresponding text transcriptions, and metadata, offering a valuable resource for developing ASR systems, especially for low-resource languages.

Practical constraints like computational resources and training time required us to work with a subset of the dataset. This subset, sourced from Delta Segment releases (versions 18.0 and 19.0), provides a representative sample of Catalan, encompassing diverse speakers.

Labeled Data for Baseline ASR and Fine-Tuning:
Before integrating unlabeled data pretraining, we first establish a baseline model trained exclusively on the labeled Catalan data segments. This baseline allows us to understand the performance using conventional supervised approaches without leveraging self-supervised learning. Specifically, we rely on the validated speech segments and their corresponding transcripts from the chosen Catalan subsets.

Data Preprocessing and Preparation Steps:
All audio is standardized to a uniform sampling rate (16 kHz) to align with the model's requirements. Advanced noise reduction or normalization are not strictly needed since the model robustness, so we maintain a consistent preprocessing pipeline for simplicity and reproducibility.

On the textual side, we extract the transcripts of each validated samples from the .tsv files. From these transcripts, we build a character-level vocabulary to support a Connectionist Temporal Classification (CTC) based ASR objective. The character set includes letters, punctuation, and special tokens for unknown characters, padding, and word boundaries. This vocabulary is saved for use with the tokenizer and processor, ensuring the model can tokenize and map raw text to integer IDs suitable for training.

Following vocabulary construction, we pair each audio clip with its transcript to create a unified dataset of speech-text pairs.

3.2 Baseline Training with Labeled Catalan Data

Model Initialization and Configuration: The chosen configuration reflects widely used settings from the literature, including the number of attention heads, hidden layer dimensions, and the vocabulary size corresponding to the character set derived from the Catalan transcripts. Since no pretraining is applied at this stage, the model's weights are randomly initialized.

Tokenizer and Feature Extraction Setup:
The previously constructed vocabulary and character-level tokenizer form the foundation for converting text to labels. Then, a feature extractor customized to 16 kHz audio input ensures that raw audio waveforms are consistently processed into the numerical representations required by the model.

Data Collation and Input Batching:
To manage variability in length, we implement a data collation strategy that pads audio and label sequences to the longest sample in each batch. Special care is taken to replace padding tokens in the labels with a neutral value (-100) so that these positions do not influence the model's loss calculation.

Training Procedure and Optimization:
With the training and validation sets prepared, we run multiple epochs of supervised training with a learning rate schedule with warm-up and gradual decay. Throughout training, we monitor performance on the validation set, using metrics like Word Error Rate (WER) to track the model's

progress.

3.3 Brief Results on Baseline Model

See loss graphs in annex: A.2

The baseline model was trained using a labeled dataset, following a 90/10 train-test split (configured with a hidden size of 768, 12 attention heads, and 12 transformer layers, using dropout)

Despite successfully learning from the available data, the model's performance was limited by the size and diversity of the dataset. While the training and validation losses were stable, the results revealed significant challenges in generalizing to unseen audio samples.

Qualitative evaluation showed that the model's transcriptions often lacked coherence, frequently reducing outputs to single characters or incomplete sequences. Transcriptions were mostly reduced to single characters or incoherent outputs (e.g., "E," "Q," "Es"), highlighting the model's inability to generalize effectively to new inputs. Overall, the baseline model demonstrated the need for more robust representation learning techniques, such as those enabled by self-supervised approaches, since the lack of labelled data.

4 Main Experiment and Evaluation

4.1 Wav2Vec 2.0 Pretraining from Scratch

This process involved extensive data preparation, constrained by the computational resources available, and the implementation of self-supervised learning techniques. The pretraining task played a crucial role in enabling the model to learn meaningful speech representations from unlabeled audio.

4.1.1 Data Sources and Preparation

For the pretraining phase, we utilized unlabeled audio from Common Voice Delta Segments 15.0, 16.0, and 17.0, amounting to approximately ***30 hours*** of validated speech. Labeled data, sourced from Common Voice Delta Segments 18.0 and 19.0, comprised ***10 hours*** of transcribed audio. While labeled data is critical for fine-tuning and evaluation, the bulk of the training process relied on the larger, unlabeled corpus to exploit self-supervised learning.

Dataset Chunking:
One of the major constraints in this project was the limited storage capacity of the computational cluster. Unlike images, audio files occupy significantly more disk space due to their larger file sizes. To address this challenge, the dataset was preprocessed and stored in smaller chunks. This chunking process allowed us to sequentially load and train on the dataset without exceeding the storage limits of the system. Random chunk splitting ensured that the audio distribution across chunks remained representative for pretraining.

4.1.2 Pretraining Task

Objective of Pretraining:
The self-supervised pretraining task for Wav2Vec2 is designed to learn general-purpose speech representations by predicting masked portions of the input audio. This approach is inspired by the masked language modeling paradigm in natural language processing.
During pretraining:

- A portion of the latent speech representations is masked (15% of time steps), and the model learns to reconstruct the masked regions based on the surrounding context.

- The model leverages contrastive learning to differentiate true quantized embeddings of masked regions from a set of negative distractors.

Quantization and Contrastive Loss:
The architecture includes a quantization module that converts continuous latent audio representations into discrete codewords. These codewords, sampled from a learned codebook, serve as the targets for the contrastive objective. The contrastive loss encourages the model to produce embeddings that are closer to the correct quantized targets while pushing away distractor embeddings. Additionally, a diversity loss is incorporated to ensure that the entire codebook is utilized effectively, avoiding collapse into a small subset of codewords.

4.1.3 Results on pretrain

The Wav2Vec2 pretraining phase was conducted in multiple stages, with the first phase using an initial learning rate of 8e-5 , a batch size of 8, and gradient accumulation steps set to 2. The model was configured with 768 hidden dimensions, 12 attention heads, and 12 transformer layers. A masking probability of 15% and mask length of 10 were used to optimize the self-supervised objective. Weight decay was applied at 0.010.010.01, and training proceeded for 5–6 epochs per phase.
First Phase Related loss graph in annexA.3.1
During the first pretraining phase, the training loss quickly dropped to -0.8, indicating somehow effective learning of representations. This value, is not strange in contrastive learning, suggests the model was distinguishing between true and distractor samples. However, the loss plateaued, signaling the need for adjustments in learning rate and setup in subsequent phases to refine the model further.

Evaluation on third phase:
By the third pretraining phase, the learning rate was reduced at the start of the phase. Despite this, the training loss did not change, having to take the decision to evaluate additional metrics to better understand the model's performance on the pretraining task. See related loss in annex A.3.2

- **Validation Contrastive Loss:**This fluctuation suggests that the model was struggling to consistently improve its ability to distinguish true quantized embeddings from negative distractors.
- **Validation Diversity Loss:** The diversity loss remained relatively stable, indicating that the model was effectively utilizing the codebook without collapsing into a narrow subset of codewords.
- **Mask Reconstruction Accuracy:** The accuracy for reconstructing masked audio segments showed minimal improvement during this phase, hovering around 1.0. This suggests that while the model learned basic patterns in the speech signal, further gains in accuracy were limited by either the data or the architectural constraints.

4.2 Wav2Vec 2.0 Finetuning from Scratch Pretrained Model

It involves optimizing the model on transcribed audio segments to map spoken language to text. A Connectionist Temporal Classification (CTC) layer is added on top of the pretrained architecture to align audio frames with character sequences. The model is trained using CTC loss, while its transcription accuracy is evaluated with Word Error Rate (WER).

4.2.1 Preparation

As explained before, this part uses the labelled data and tokenizer mapped text to integer IDs using the predefined vocabulary, while the processor ensured compatibility with the model, including handling variable-length sequences with padding tokens for consistency.

4.2.2 Model Initialization

Configurations on the model remained the same but for preparing the model to perform a specific we needed to:

1. **Layer Initialization:**
   The linear layer (lm_head) responsible for mapping the model's output embeddings to character-level predictions was resized and reinitialized to accommodate the Catalan vocabulary.

2. **Pretrained Weights Transfer:**
   The pretrained weights from the Wav2Vec2 model were copied into the fine-tuning model to retain the representations learned during the self-supervised pretraining phase.

4.2.3 Training Process

The training process followed the same methodology as the baseline, with the primary objective of aligning audio frames to character sequences. This involved fine-tuning the pretrained model using labeled data and optimizing it for transcription accuracy.

4.2.4 Quantitative Results

First Phase:
In the first phase of fine-tuning, the model was trained over 10 epochs with a learning rate of 1e-4 using a linear scheduler. Both training and validation losses decreased steadily, with the training loss converging toward 1.1. However, the Word Error Rate (WER) remained consistently high, above 0.998, indicating minimal improvement in transcription accuracy. This suggested that the learned representations were not yet sufficiently refined to yield meaningful performance gains.

To address this, adjustments were made for the next phase, including lowering the learning rate to 5e-5 to allow for finer parameter updates and further optimization.

See related loss in annexA.4.1

Second Phase:
In the second phase, the model showed marked improvements. Training loss dropped below 0.6, and validation loss stabilized around 0.8, reflecting better generalization compared to the first phase. Importantly, the WER showed significant progress, decreasing from above 0.98 to approximately 0.88 by the end of this phase. This indicates that the reduced learning rate and extended fine-tuning contributed to enhanced transcription accuracy.

See related loss in annexA.4.2

4.2.5 Qualitative Results

The qualitative analysis of the model's transcriptions reveals both strengths and areas requiring improvement. As shown in the examples, the model outputs include frequent [UNK] tokens, which indicate instances where the model could not confidently identify certain characters or sounds. Despite this limitation, the remaining decoded tokens align reasonably well with the ground truth (GT), capturing much of the phonetic structure and sound of Catalan speech, but if we do a deeper analysis, we can see that the model struggles with the hardest sounds in Catalan. (e.g ("ll" de "lliçó")).

If the [UNK] tokens are disregarded, the model's predictions often approximate the ground truth quite closely. For example:

- GT: Quina carrera va començar? - Prediction [UNK]uina [UNK]carrera[UNK]va[UNK]començar [UNK]
- GT: Què els va respondre l'oracle? - Predicción: [UNK]els[UNK]va[UNK]despon[UNK]breloracla [UNK]
- GT: A quina resolució es vol arribar amb aquest recorregut? - Predicción: [UNK]quina[UNK]resluci [UNK][UNK]vo[UNK]arribar[UNK]amb[UNK]aquesttre[UNK]correbut[UNK]
- GT: Enguany s'havia de fer una gran celebració? - Predicción: [UNK]nguany[UNK]sebia[UNK]de [UNK]fer[UNK]una [UNK]gra[UNK]n[UNK]sa[UNK]la[UNK]brasci[UNK][UNK]

Maybe if you are not native of the language, you could think that the predictions are quite bad, but they relate pretty accurately to how the words sound phonetically, not grammatically.

This contrasts with the baseline, which failed to generalize phonetic structures effectively. The fine-tuned model's ability to capture Catalan pronunciations, even if not perfectly transcribed, reflects its enhanced learning of linguistic patterns compared to the baseline's minimal capability. However, further improvements are still required to fully resolve [UNK] tokens and achieve higher accuracy.

4.2.6 Understanding the Model

GRAD-CAM

The Grad-CAM visualizations provide insights into which segments of the input waveform the model focused on during transcription. The blue lines represent the audio waveform, while the red Grad-CAM overlay indicates the model's attention or importance scores across the temporal dimension.

The Grad-CAM shows consistent attention across various sections of the waveform but fluctuates significantly. This suggests that the model is attempting to attend to relevant phonetic features. See more visualizations here:A.5

5 Finetuned Models

The primary aim of this section is to explore how linguistic similarity influences model adaptation. Our central question revolves around ***language relatedness***: does the shared Romance heritage between Spanish and Catalan lead to better adaptation and performance compared to English, which belongs to a more distant language family? To investigate this, we fine-tuned Wav2Vec2 models pretrained on Spanish and English speech data separately, using Catalan labeled data, and compared their performance.

5.1 From spanish pretrained model to catalan finetuned model

In this phase, we fine-tuned a Wav2Vec2 model that was pretrained on Spanish speech data, leveraging its linguistic similarity to Catalan. Spanish and Catalan share a common Romance language heritage, with overlapping phonetic and grammatical structures, making this transfer an ideal test of whether pretraining on a related language provides a performance advantage.
The fine-tuning process closely mirrors the earlier described methodology, using the same Catalan labeled dataset for training and validation. The pretrained Spanish model was fine-tuned over **15 epochs** with a learning rate of **3e-4**.

5.1.1 Qualitative and quantitative results

The quantitative graphs display a good curve in training and validation losses over the course of the fine-tuning process. However, the relatively high loss values suggest that the model has not fully captured all necessary linguistic patterns in Catalan.
Qualitative results in annex A.9

When analyzing the qualitative outputs, the fine-tuned model demonstrates notable progress in transcribing Catalan speech. Some examples are grammatically incorrect but phonetically accurate.
The fine-tuned model, although not perfect, has slightly adapted to Catalan the linguistic similarity allows the model to capture many phonetic and structural patterns, making it look like a Spanish speaker learning Catalan. While the predictions are not grammatically perfect, they show a strong resemblance to Catalan speech, especially in pronunciation and word structure.

For guidance in metrics, thus its not what we were looking for, computed the WER once finished the training, in a small subsample.
WER: 0.60 without normalizing, after normalizing and not taking into account uppercase and punctuation signs it dropped to almost 0.40.

5.2 From english pretrained to catalan finetuned

This part shows the pretrained model on English speech to Catalan, using the same methodology as in the Spanish-to-Catalan experiment.

5.2.1 Quantitative and Qualitative Results

Go to Annex A.10 for the corresponding loss graphs

When inspecting the qualitative transcriptions, it became evident that the English-pretrained model struggled significantly to adapt to Catalan. Transcriptions were often incoherent and lack of resemblance to the ground truth.

The qualitative results demonstrate that the English-pretrained model was unable to generalize to Catalan, as it lacks shared linguistic and phonetic roots. It instead defaulted to producing sequences resembling English pronunciation and structure.

The stark contrast in performance between models highlights the importance of linguistic similarity in transfer learning. The shared Romance roots between Catalan and Spanish allowed for a smoother adaptation, as seen in previous experiments, whereas English's distant Germanic origins led to minimal transferability. We computed the WER once finished the training, in a small subsample.
WER: **0.99** and normalized **0.96**
Here are some qualitative results:

- GT: Què els va respondre l'oracle? - Prediction: GETS MA AS PORM THAT I RACKLER
- GT: També ha criticat que el president de la Diputació atorgui subvencions directed. - Prediction: THAN ME APLIKICAT TON PRASIDET THAT ALE PUT AS OU A ORE GUITRU BUT SUGIETA
- GT: Què va provocar que el pastor matés a la seva enamorada? - Predicció: GI BAPURUCAC EL PASTUMATISARASIVA AND AMUDADA

5.3 Multilingual Fine-Tuned Model in Catalan

Model losses in annex A.11
The ***wav2vec2-large-xlsr-53 model*** is a multilingual model pretrained on 53 languages, making it a versatile tool for speech recognition across diverse linguistic contexts. In this experiment, we fine-tuned this model specifically on Catalan data to evaluate how its multilingual foundation impacts performance compared to monolingual and linguistically closer pretrained models. The results for the multilanguage fine-tuned model on Catalan show incredible improvements in its ability to generate coherent transcriptions, often aligning really close to the ground truth. Qualitative observations reveal that the model demonstrates a robust understanding of Catalan sentence structures and vocabulary. The predictions, in many cases, accurately capture the semantics and syntax of the language, with very few grammatical or phonetic errors compared to previous models.

Overall, these results highlight the strength of the multilingual fine-tuned model in adapting to Catalan. While not perfect, it clearly outperforms the Scratch, Spanish and English-pretrained models, making it a strong candidate for practical applications.
We computed the WER once finished the training, in a small subsample.
WER: 0.15 and normalized 0.10

5.4 Conclusion on finetuned models

The multilanguage fine-tuned model demonstrated the best performance among all approaches. Its success due to the extensive representation it has learned from numerous languages, making it richer in linguistic diversity, more data, and significantly more training hours. This extensive pretraining provided the model with a strong foundation to adapt effectively to Catalan, leveraging shared features across languages while being robust to language-specific nuances.

In conclusion, our hypothesis and main research question are validated: the linguistic root of a language indeed facilitates pretraining and fine-tuning for a related language. However, when sufficient data and resources are available, training on a diverse set of languages from various roots proves to be the most effective strategy, yielding superior results by enabling the model to learn broader and more versatile speech representations.

To see how the best model focuses on the audio, let's visualize it with GradCam.A.12

5.5 What features is the model in fact learning?

From what we've seen, the model can learn 3 main characteristics:

- Basic Acustic characteristics: The model has proved to be able to identify frequencies, durations, rhythms, tones... Basic patterns in the audio, essential to differentiate sounds, understand the flow of the language, and recognize patterns of natural language.
- Contextual Representations: The model has learnt temporal relations between sounds across the length of an audio, what phonemes belong to a letter, and some degree of semantic information. Useful for weaving and understanding full words and sentences, and making statistical correlations to predict sounds in each context.
- Language Representations: Multilingual models such as this learn universal representation, common across languages, along with many language characteristics, such as strange pronunciations or unique sounds, such as ñ. This is a key feature, especially for the universality of said multilingual models.

6 Going beyond the main task

After diving into the model's performance on ASR, we wanted to push its boundaries and see just how far we could stretch its capabilities. Using metadata from the Common Voice dataset, we decided to explore some exciting additional tasks. In this section, we'll take a look at how the model handles speaker gender identification, and accent detection, testing the limits of what these self-supervised representations can really do.

6.1 Accent Recognition

Accent recognition involves classifying speech samples based on regional variations in pronunciation. Using the metadata from the Common Voice dataset, we focused on five Catalan accent categories: Central, Valencian (La Vall d'Albaida), Northwestern (Tortosí), Balearic, and Septentrional. The dataset was preprocessed to balance the class distribution, ensuring fair representation. For training, we fine-tuned the facebook/wav2vec2-base model, adapting its configuration for a multi-class classification problem.

For a map of the different dialects of catalan, see annex A.13.1

Visit annex A.13 for a map that shows which accent corresponds to which region.

ResultsA.13

The model performed really well, reaching an impressive F1 score and recall of about 0.9768. Both training and evaluation losses dropped steadily, showing that the model was learning effectively without overfitting. It was able to pick up on the unique acoustic patterns that distinguish these accents, proving how flexible and capable Wav2Vec2 is for tasks beyond ASR.

- Accuracy 97.68
- F1 Score 0.9768
- Recall 0.9739

Important note: We only used some dialects for training due to not having enough examples of these to train, meaning that despite very good results, it might not perform so well in a real environment since some inferences might not pertain to any class it was trained on.

6.2 Gender Recognition

A.13.2

For gender recognition, we followed a similar methodology as the accent recognition task, adapting the dataset and model configuration to classify gender. Using the metadata available in the Common Voice dataset, we prepared a balanced dataset of male and female audio samples. The classification task utilized a fine-tuned Wav2Vec2 model configured for binary classification, with modifications to the final layers to align with the two-class structure.

Results

The model achieved impressive performance, as seen in the evaluation metrics:

- Accuracy: 97.25
- F1 Score: 0.9725
- Recall: 0.9725

These results indicate that the model successfully captured the distinguishing features between male and female voices. Gender recognition benefitted from the clear acoustic differences between classes, demonstrating the model's adaptability to speaker-related tasks.

7 Conclusions:

This project gave us a deeper understanding of how powerful self-supervised learning can be and how can help tackle challenges in low-resource languages like Catalan. By testing different approaches we saw how both linguistic similarity and diverse data impact performance in this kind of models.

One key takeaway is that shared linguistic roots, like those between Spanish and Catalan, do help. So our main question to solve was in fact True. However, the multilingual model outperformed all others, showing that exposure to a broader variety of languages creates a richer and more flexible representation, even if they aren't as closely related.

We also faced the challenges of working with limited data. Training from scratch was possible, but without the deeper linguistic knowledge offered by pretraining, the model struggled to perform as well.

A surprising highlight was the model's versatility. Beyond ASR, we successfully explored tasks like accent and gender recognition. This showed us that these models aren't just learning language—they're capturing speaker-specific features too, opening doors to even more applications.

On a personal note, this was our first time working with audio data, which made the experience both challenging and fun. It was fascinating to dive into the complexities of sound and speech, but it also came with a learning curve. Fun fact: we ended up spending over 270 hours training models in the cluster! Despite the long hours and technical hurdles, this project proved to be an incredibly rewarding journey.

In the end, this project reminded us of both the possibilities and the challenges in using SSL for smaller languages. While tools like multilingual pretraining are incredibly effective, there's still a need for more data and better ways to train on it. Despite that, the potential to preserve and support minority languages through these models is an exciting step forward in making technology more inclusive.
Thanks for reading.
The Lords of The Layers.

8 References

- Self-Supervised Learning and Wav2Vec2: Baevski, A., Zhou, H., Mohamed, A., & Auli, M. (2020). wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations. Advances in Neural Information Processing Systems, 33, 12449-12460. https://arxiv.org/abs/2006.11477

- Low-Resource Language Technologies: Bird, S. (2020). Decolonising Speech and Language Technology. Proceedings of the 28th International Conference on Computational Linguistics, 8-24. https://aclanthology.org/2020.coling-main.8/

- Mozilla Common Voice Dataset: Ardila, R., Branson, M., Davis, K., et al. (2020). Common Voice: A Massively-Multilingual Speech Corpus. Proceedings of the 12th Conference on Language Resources and Evaluation (LREC). https://arxiv.org/abs/1912.06670

- https://github.com/somosnlp/wav2vec2-spanish

- Romance Language Similarities and Transfer Learning: Choueiter, G., Anguera, X., & Lopez-Otero, P. (2021). Cross-Lingual ASR Adaptation: Using Shared Linguistic Traits in Multilingual Training. IEEE Transactions on Audio, Speech, and Language Processing, 29, 314-326.

- Evaluation Metrics in ASR: Chan, W., Jaitly, N., Le, Q. V., & Vinyals, O. (2016). Listen, Attend and Spell. Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 4960-4964.

- Multilingual Models in ASR: Conneau, A., Khandelwal, K., Goyal, N., et al. (2020). Unsupervised Cross-lingual Representation Learning at Scale. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL). https://arxiv.org/abs/1911.02116

- Grad-CAM for Interpretability: Selvaraju, R. R., Cogswell, M., Das, A., et al. (2017). Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. Proceedings of the IEEE International Conference on Computer Vision (ICCV), 618-626. https://arxiv.org/abs/1610.02391

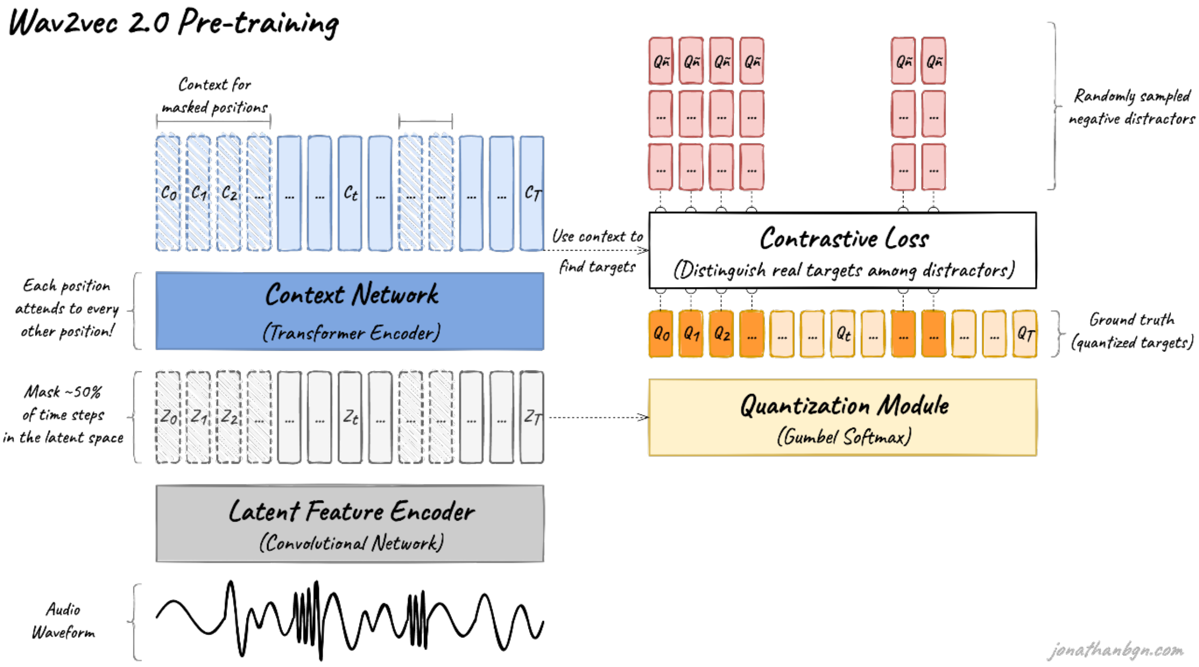
- Ethical Considerations in ASR for Low-Resource Languages: Hovy, D., & Spruit, S. L. (2016). The Social Impact of Natural Language Processing. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL), 591-598.

A Appendix

A.1 More Detailed model Explanation:

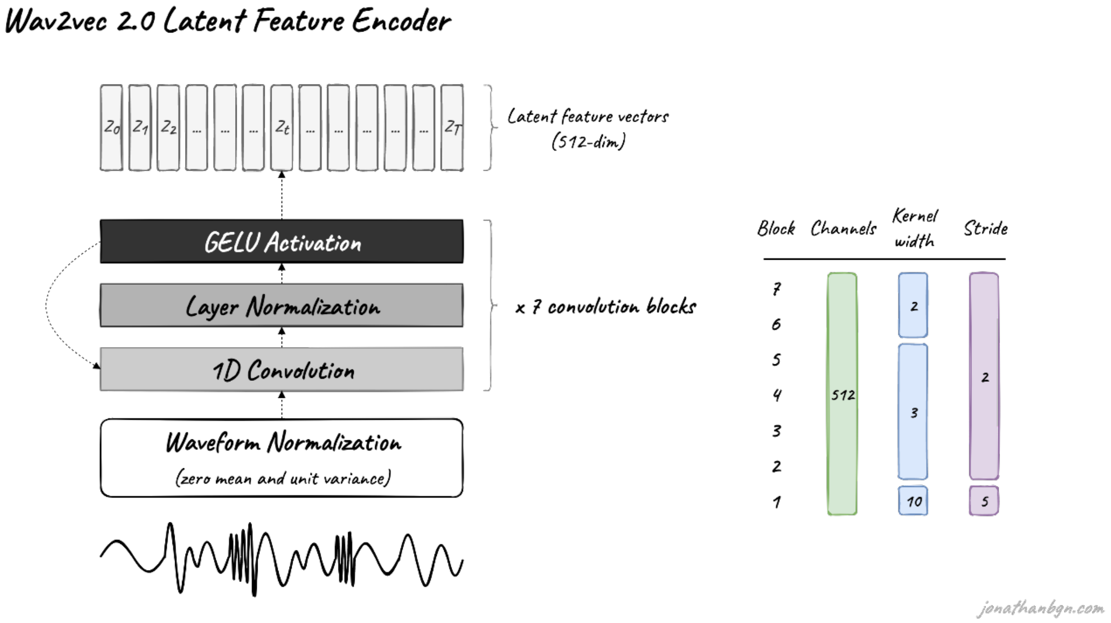
Wav2Vec2's architecture can be broadly divided into two main components: a **convolutional feature encoder** and a **transformer-based contextual encoder**. The feature encoder processes raw waveforms directly, extracting low-level acoustic patterns without the need for handcrafted features such as MFCCs. These latent acoustic representations are then passed to a transformer encoder, which provides contextual understanding through self-attention mechanisms. By modeling temporal dependencies, the transformer produces rich, context-aware speech embeddings.

A key element distinguishing Wav2Vec2 from earlier approaches is its use of a quantization module that maps continuous latent representations to discrete codewords. This discretization enables the model to adopt a contrastive learning objective during pretraining, encouraging the learned embeddings to be both discriminative and robust.


Wav2vec 2.0 Pre-training
Context for masked positions
c_0 c_1 c_2 ... ... c_t ... ... c_T
Use context to find targets
Each position attends to every other position!
Context Network
(Transformer Encoder)
Mask ~50% of time steps in the latent space
z_0 z_1 z_2 ... ... z_t ... ... z_T
Latent Feature Encoder
(Convolutional Network)
Audio Waveform
$Q_{\tilde{n}}$
Randomly sampled negative distractors
Contrastive Loss
(Distinguish real targets among distractors)
Q_0 Q_1 Q_2 ... ... Q_t ... ... Q_T
Ground truth (quantized targets)
Quantization Module
(Gumbel Softmax)
jonathanbgn.com

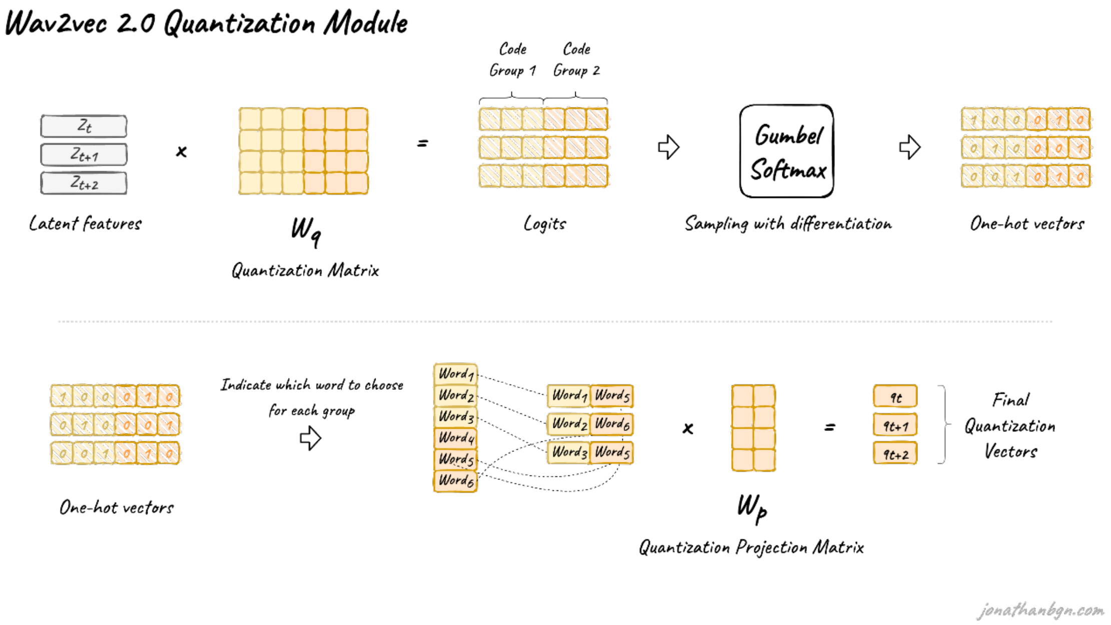
A.1.1 Input and Feature Extraction:

The input to Wav2Vec2 is raw audio, typically sampled at 16 kHz. The model does not require manual feature extraction; instead, it employs a stack of temporal convolutional layers to transform the raw waveform into latent feature vectors. Each convolutional layer often utilizes GELU activation and normalization techniques like LayerNorm or GroupNorm, ensuring stable training and effective representation learning. The output of this stage is a sequence of latent representations capturing local acoustic details of the input signal.


Wav2vec 2.0 Latent Feature Encoder
z_0 z_1 z_2 ... ... z_t ... ... z_T
Latent feature vectors (512-dim)
GELU Activation
Layer Normalization
1D Convolution
x 7 convolution blocks
Waveform Normalization
(zero mean and unit variance)
Block
Channels
Kernel width
Stride
7
6
5
4
3
2
1
512
2
3
10
2
5
jonathanbgn.com

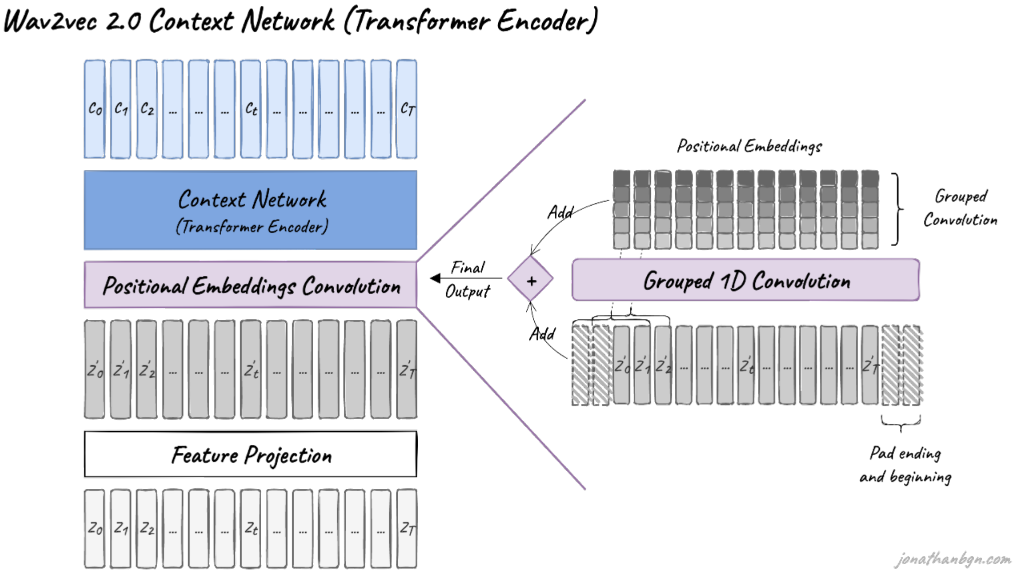
A.1.2 Quantization Module:

After obtaining these latent features, Wav2Vec2 discretizes them using a learned codebook. Techniques such as Gumbel-Softmax allow the model to select discrete codewords in a differentiable manner. The codebook is organized into multiple groups, and one codeword is chosen from each group to form a final quantized vector. A diversity loss term encourages the model to utilize the full range of available codewords, preventing it from collapsing to a small subset. This discrete representation forms the foundation of the model's contrastive objective, guiding the formation of meaningful and distinguishable speech units.



A.1.3 Transformer-Based Contextual Encoder:

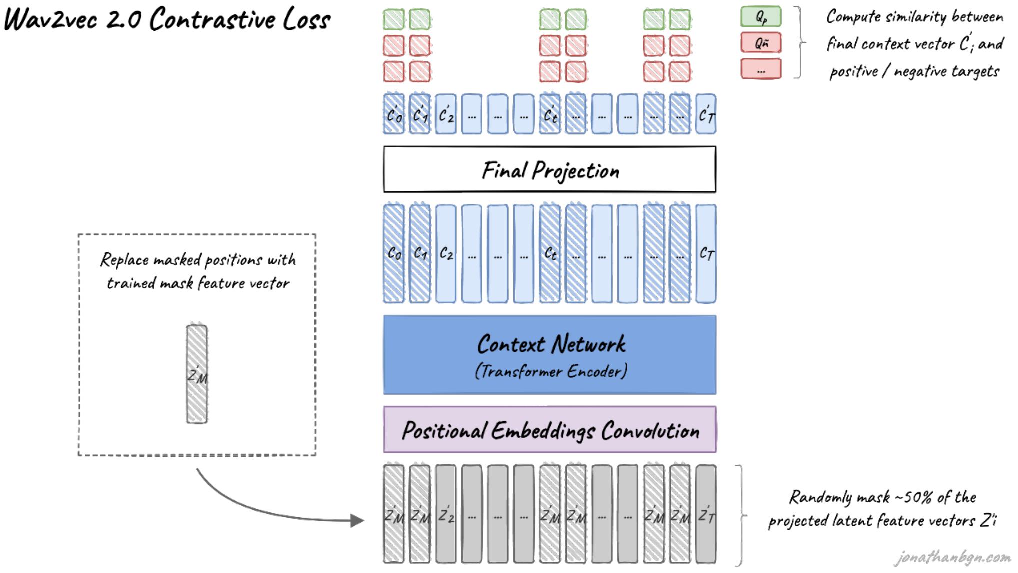
On top of the quantized representations, Wav2Vec2 employs a transformer encoder architecture. Multi-head self-attention layers in the transformer aggregate information across all time steps, allowing the model to capture both short- and long-term dependencies within the speech signal. Feed-forward layers (with GELU activations) and layer normalization further refine the representations. During pretraining, a portion of the latent features is masked, and the model must infer these masked frames from the surrounding context. This masked prediction objective is analogous to masked language modelling in NLP (as BERT) and encourages the model to learn generalizable, high-level features.


Wav2vec 2.0 Context Network (Transformer Encoder)
c_0 c_1 c_2 ... ... c_t ... ... c_T
Context Network
(Transformer Encoder)
Positional Embeddings Convolution
z'_0 z'_1 z'_2 ... ... z'_t ... ... z'_T
Feature Projection
z_0 z_1 z_2 ... ... z_t ... ... z_T
Final Output
+
Add
Add
Positional Embeddings
Grouped Convolution
Grouped 1D Convolution
z'_0 z'_1 z'_2 ... ... z'_t ... ... z'_T
Pad ending and beginning
jonathanbgn.com

A.1.4

sectionPretraining Objectives and Fine-Tuning: The pretraining phase of Wav2Vec2 is driven by a contrastive loss. For each masked time step, the model must identify the correct quantized codeword from a set of negative candidates. This objective forces the learned representations to be highly informative, making it easier for the model to distinguish correct acoustic units from incorrect ones. Additionally, the diversity loss ensures that the entire codebook is utilized, enhancing the expressiveness of the learned embedding space.

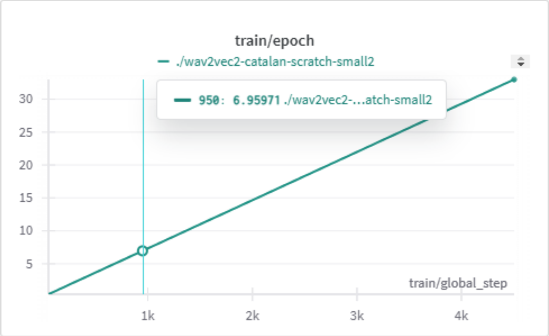
Once pretrained, Wav2Vec2 can be adapted to ASR tasks with relatively little labelled data. A common approach involves adding a simple classification head (lm_head) on top of the transformer outputs and training it with Connectionist Temporal Classification (CTC) loss. Because the model's representations are already robust and general-purpose, the fine-tuning stage can achieve strong ASR performance with far fewer labelled examples than a from-scratch training pipeline would require.

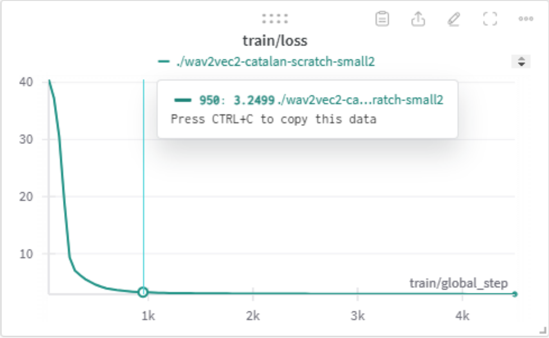

Wav2vec 2.0 Contrastive Loss
Q_p
$Q_{\tilde{n}}$
...
Compute similarity between final context vector C'_i and positive / negative targets
C'_0 C'_1 C'_2 ... C'_t ... C'_T
Final Projection
C_0 C_1 C_2 ... C_t ... C_T
Context Network
(Transformer Encoder)
Positional Embeddings Convolution
Replace masked positions with trained mask feature vector
Z_M
Z'_M Z'_M Z'_2 ... Z'_M Z'_M ... Z'_M Z'_M Z'_T
Randomly mask ~50% of the projected latent feature vectors Z'_i
jonathanbgn.com

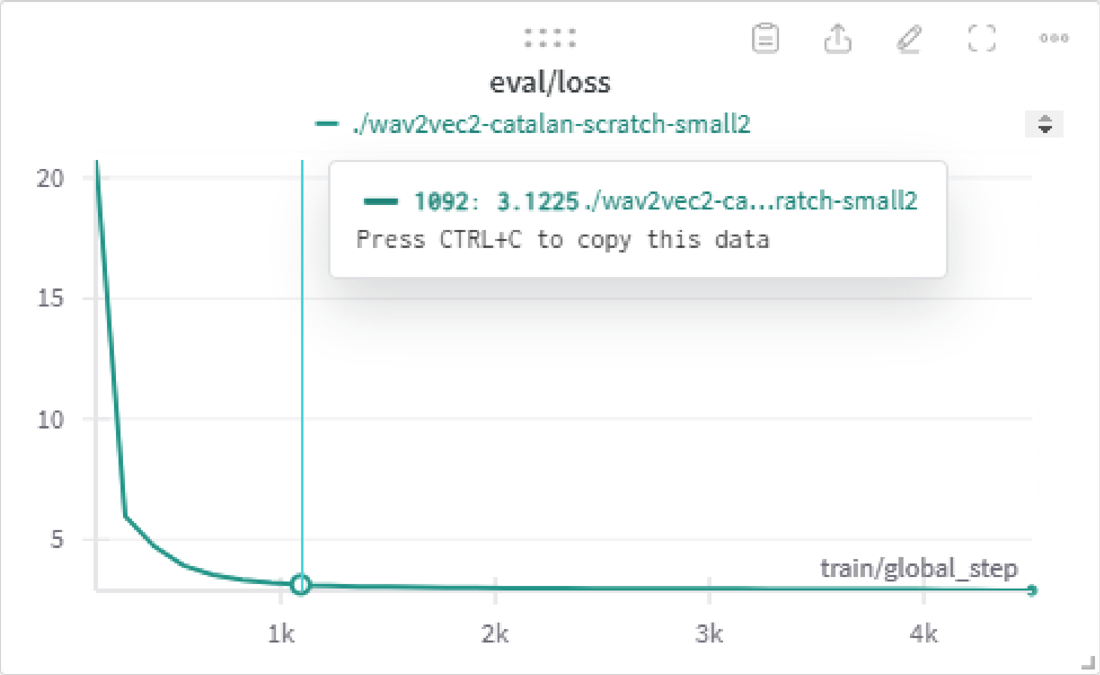
A.1.5 Evaluation Metrics and Performance

During pretraining, the model's progress is monitored via the contrastive and diversity losses. For downstream ASR evaluation, standard metrics such as Word Error Rate (WER) and Character Error Rate (CER) are employed. Numerous studies have demonstrated that Wav2Vec2 models consistently outperform traditional approaches, especially in low-resource environments, confirming that self-supervised pretraining produces robust and easily adaptable speech representations.

A.2 Results on Baseline Model

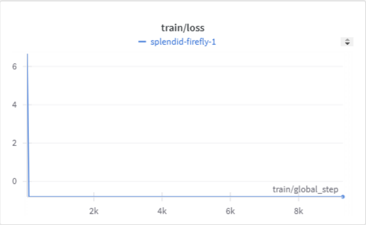


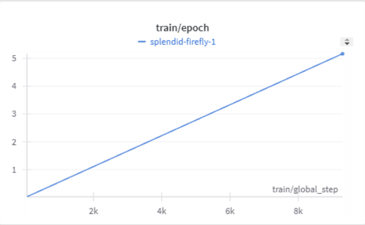


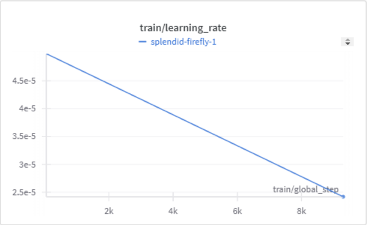


A.3 Results on Pretrain

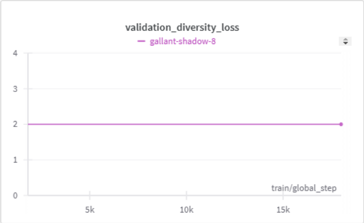
A.3.1 First Phase

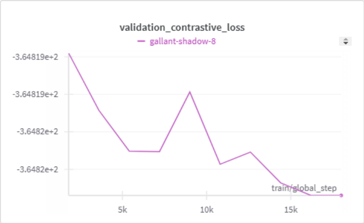


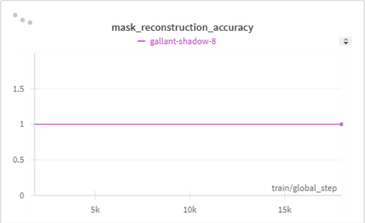




A.3.2 Third phase

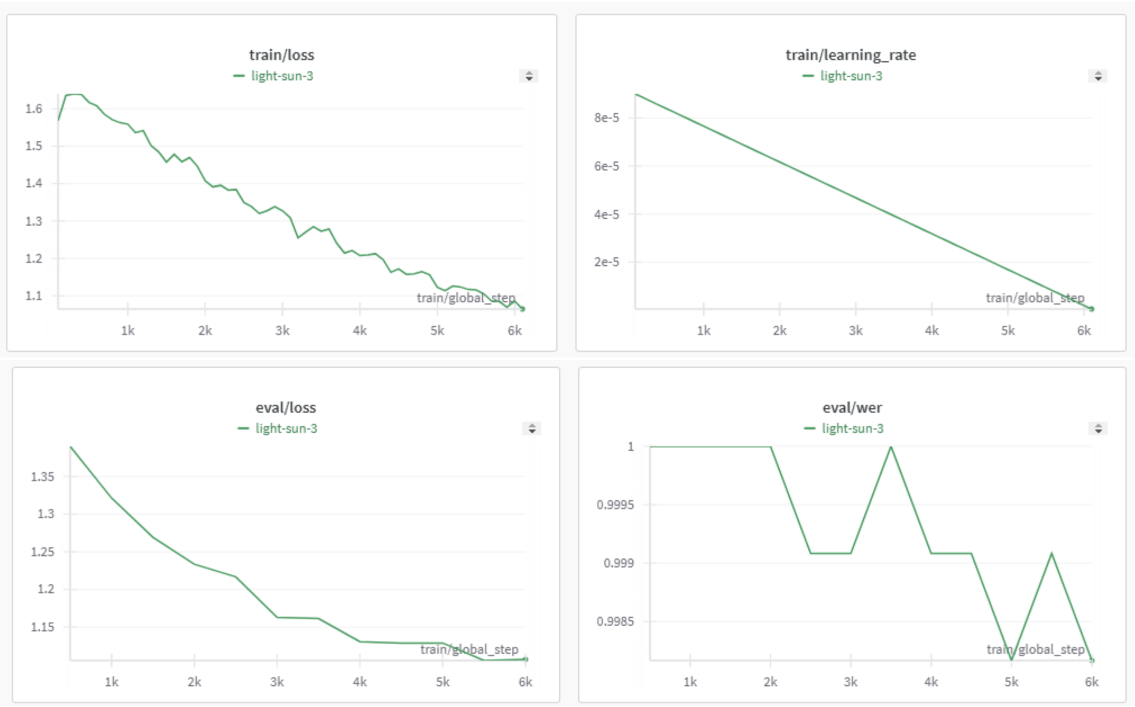




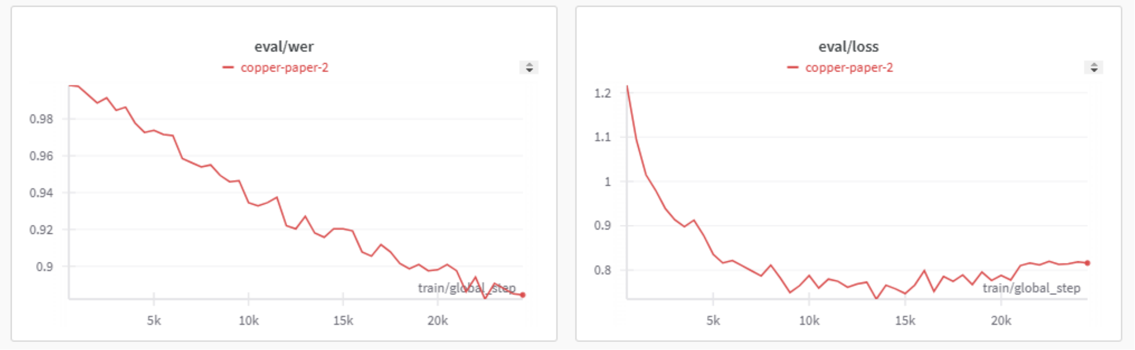


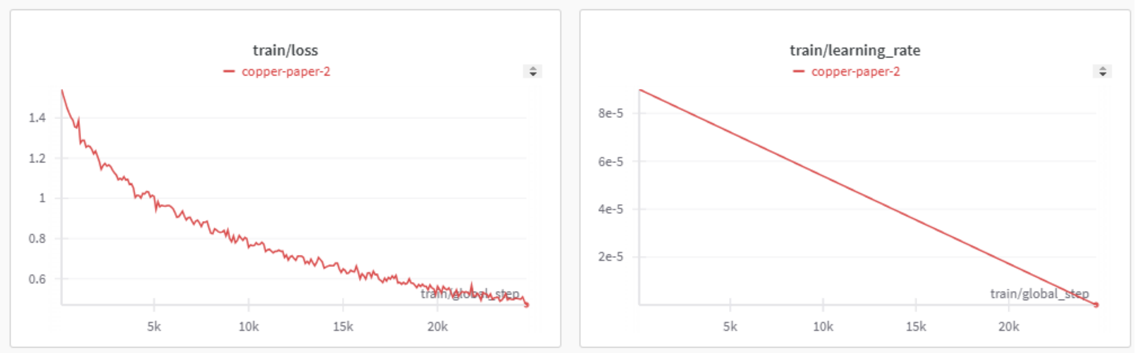
A.4 Quantitative Results

A.4.1 First Phase

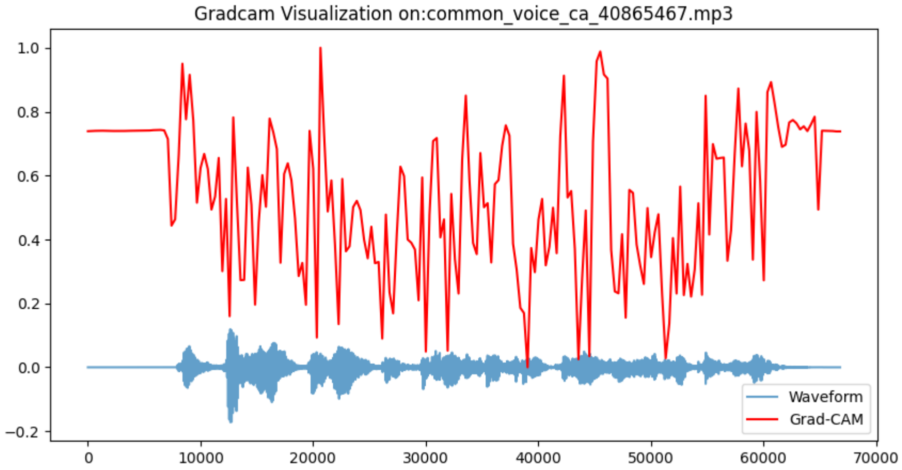


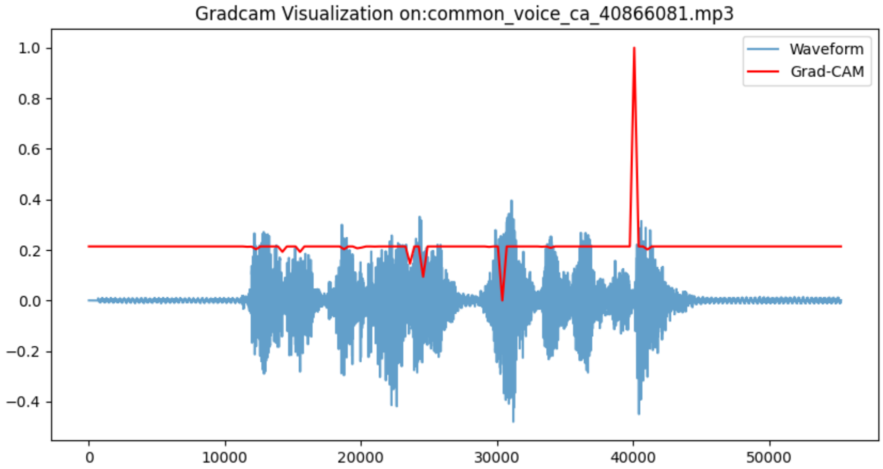
A.4.2 Second Phase

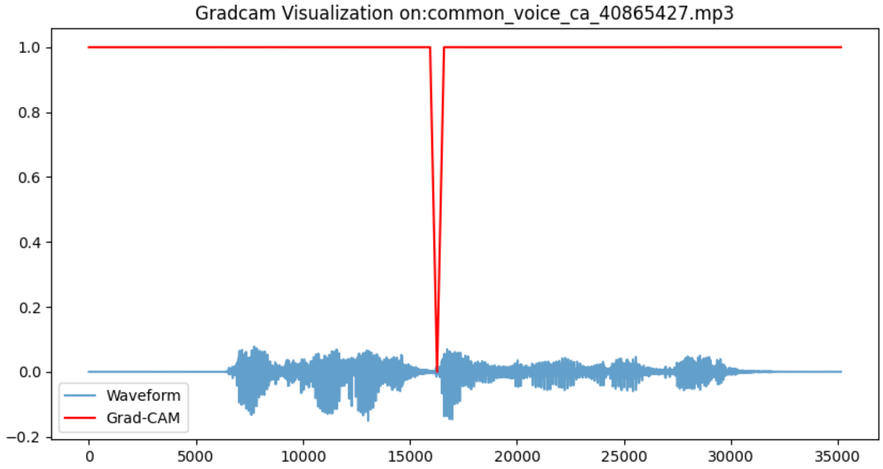




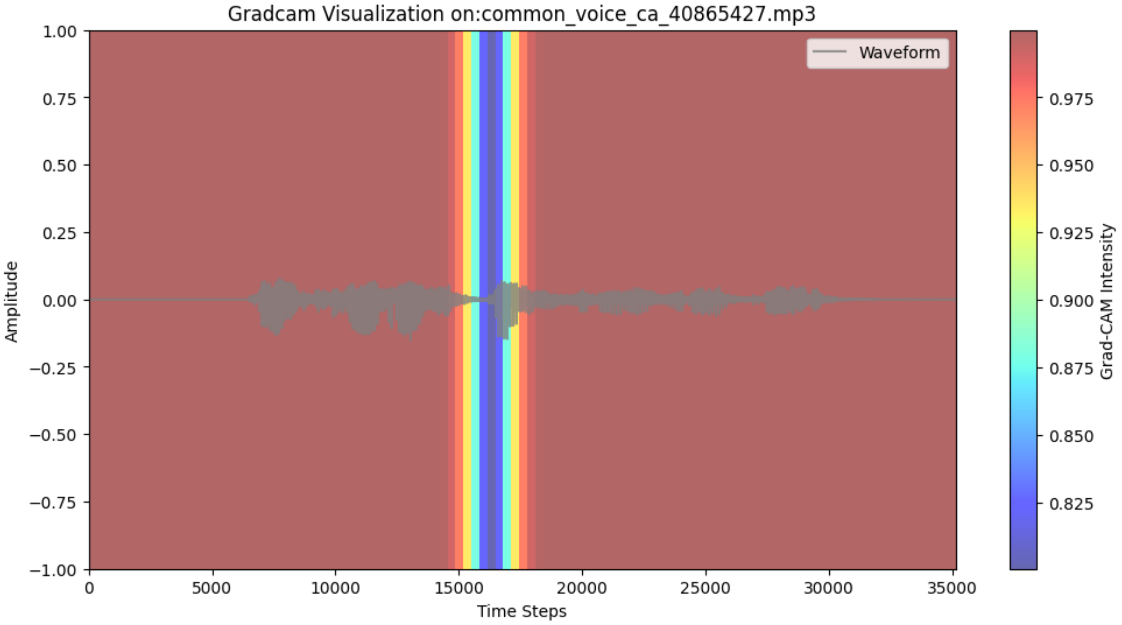
A.5 GRADCAM

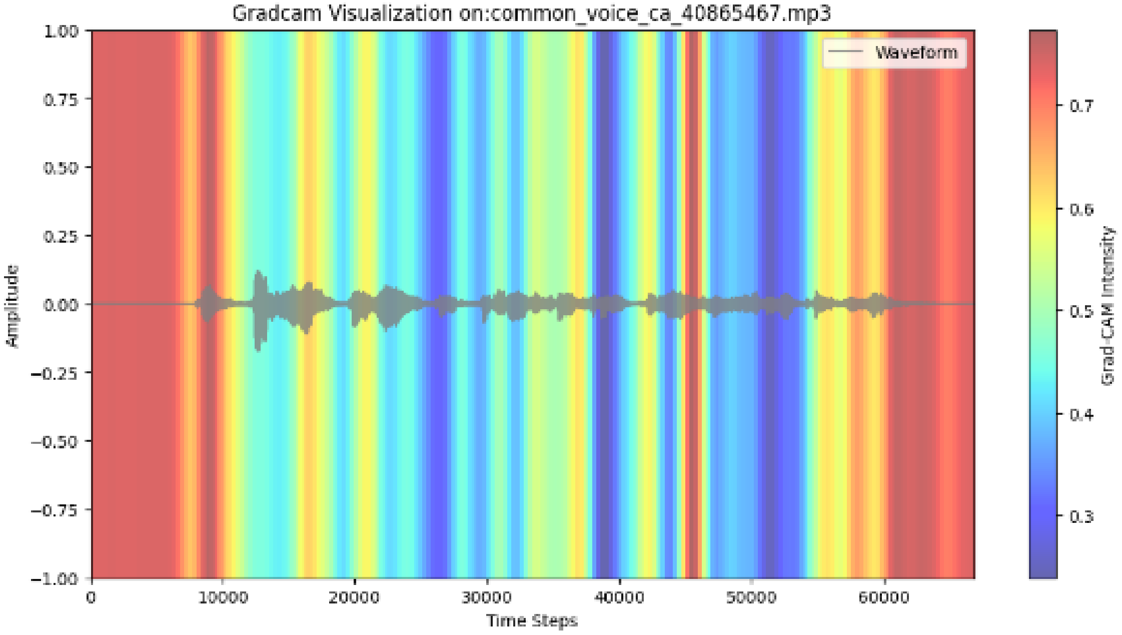

Gradcam Visualization on:common_voice_ca_40865467.mp3

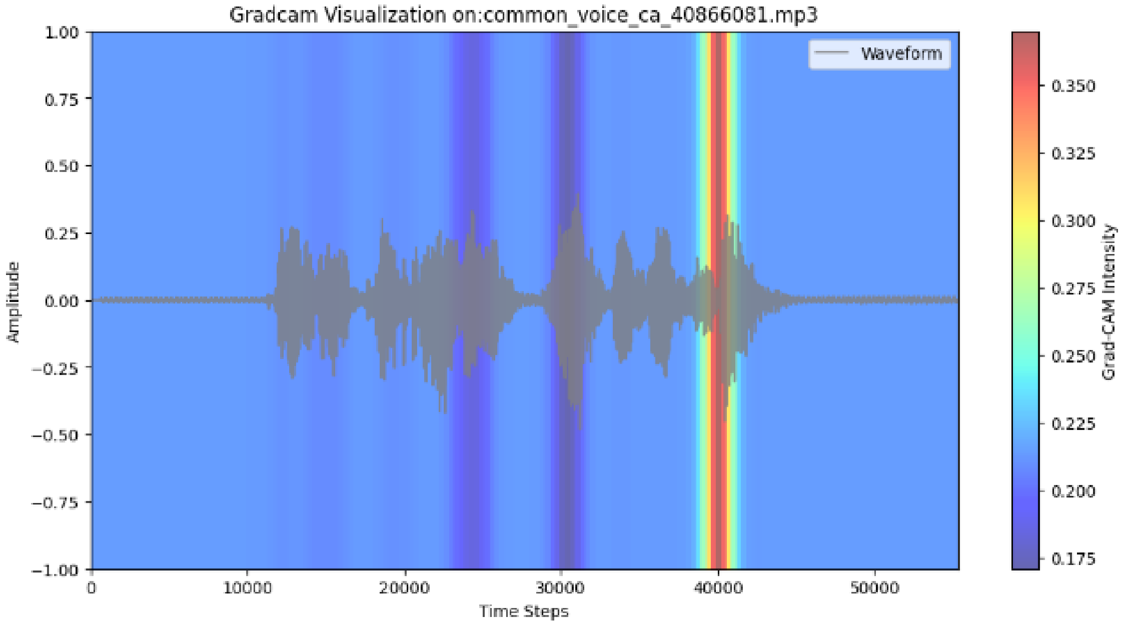

Gradcam Visualization on:common_voice_ca_40866081.mp3


Gradcam Visualization on:common_voice_ca_40865427.mp3

A.6 GRADCAM Heat

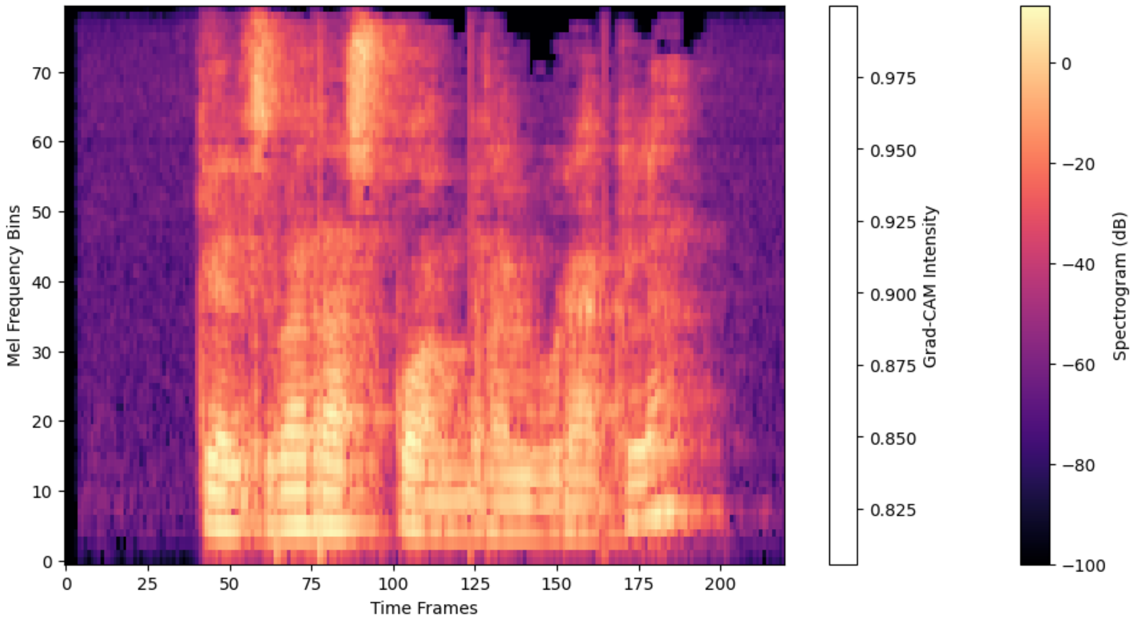


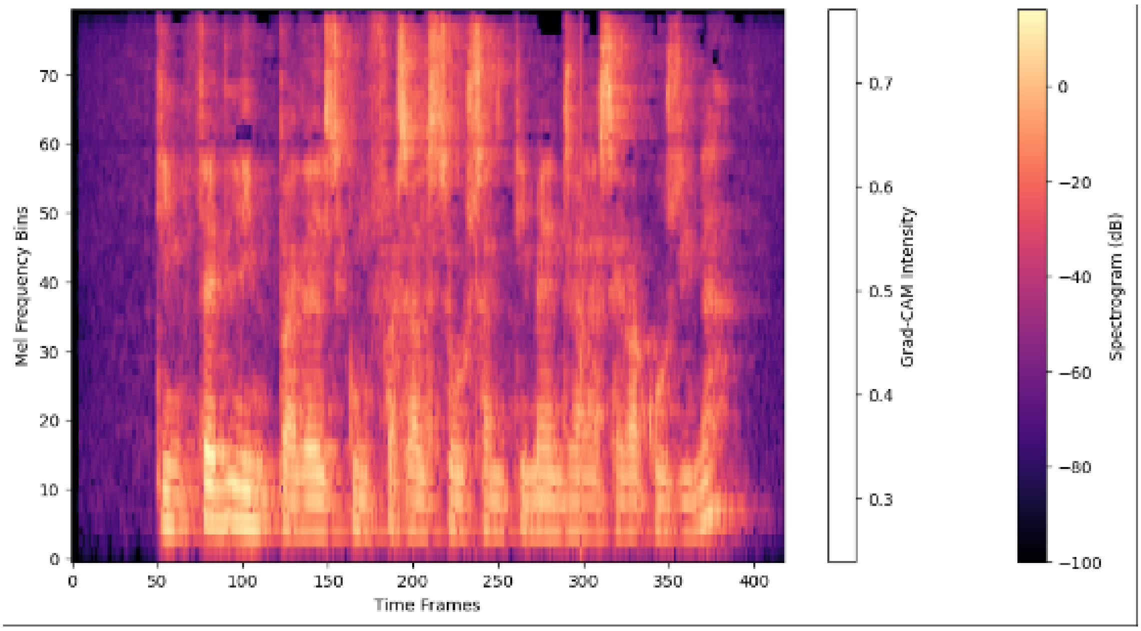


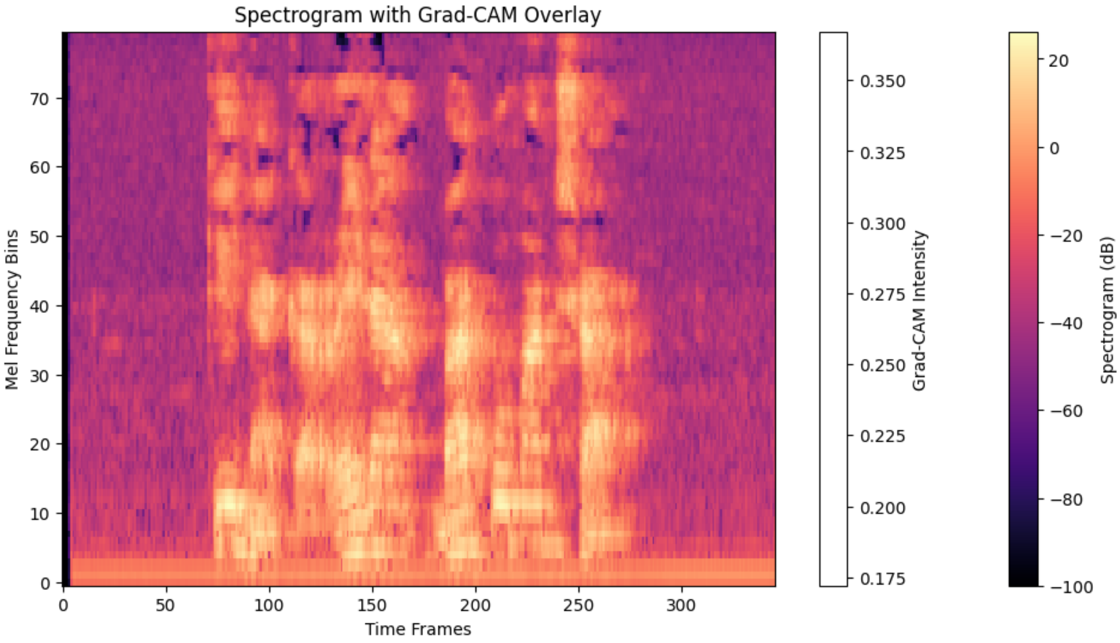


Gradcam Visualization on:common_voice_ca_40866081.mp3
Waveform
Amplitude
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
10000
20000
30000
40000
50000
Time Steps
Grad-CAM Intensity
0.350
0.325
0.300
0.275
0.250
0.225
0.200
0.175

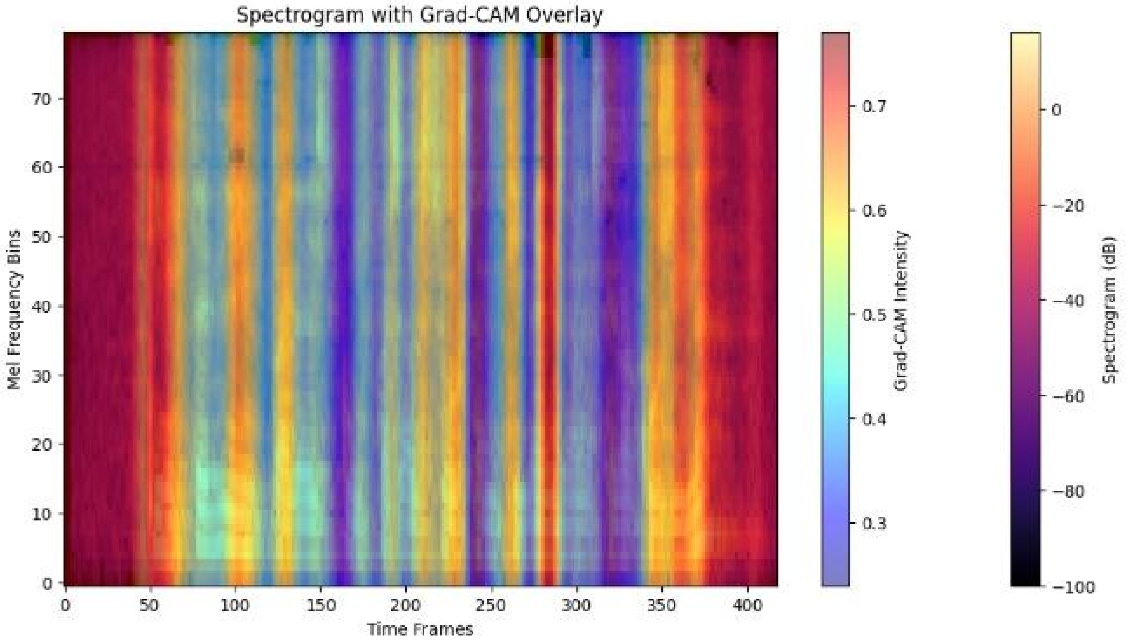
A.7 Spectrogram

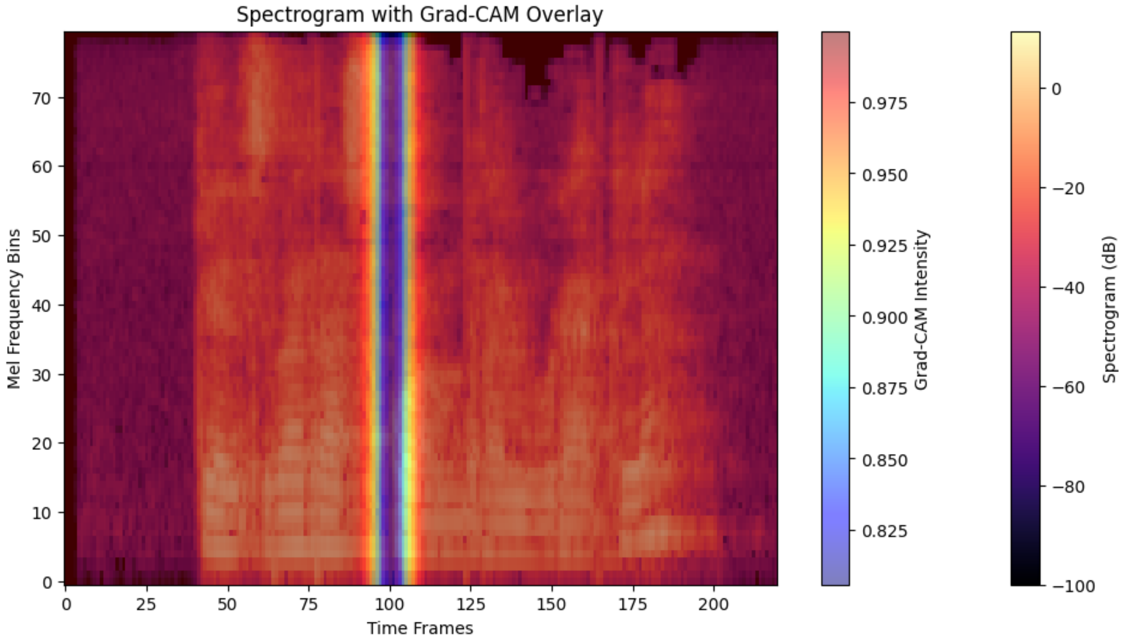


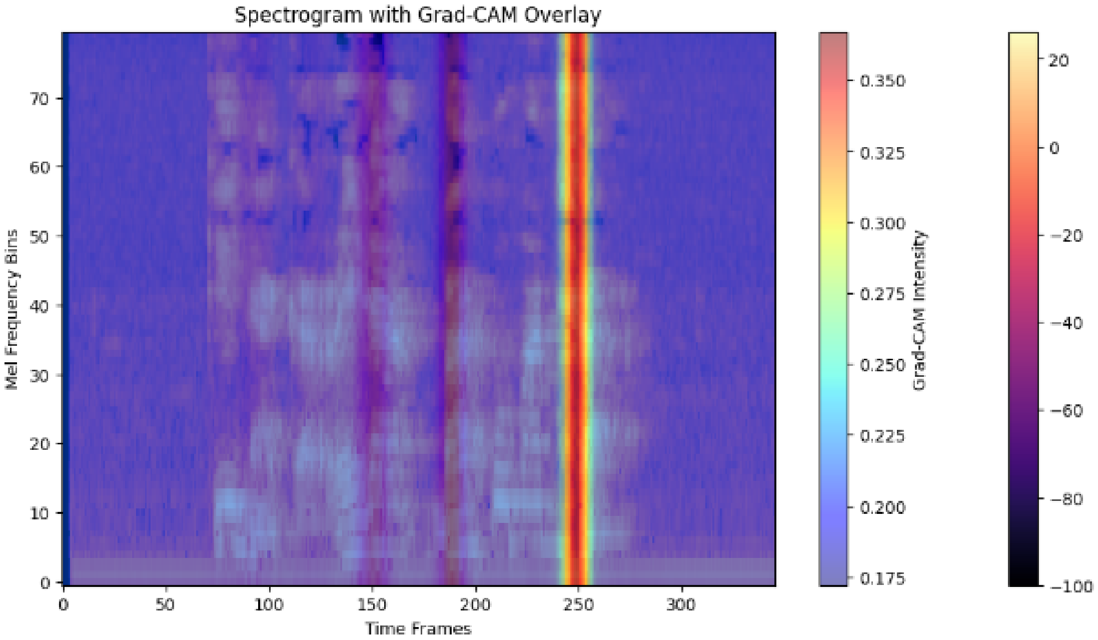



Spectrogram with Grad-CAM Overlay
Mel Frequency Bins
0
10
20
30
40
50
60
70
0
50
100
150
200
250
300
Time Frames
Grad-CAM Intensity
0.350
0.325
0.300
0.275
0.250
0.225
0.200
0.175
Spectrogram (dB)
20
0
−20
−40
−60
−80
−100

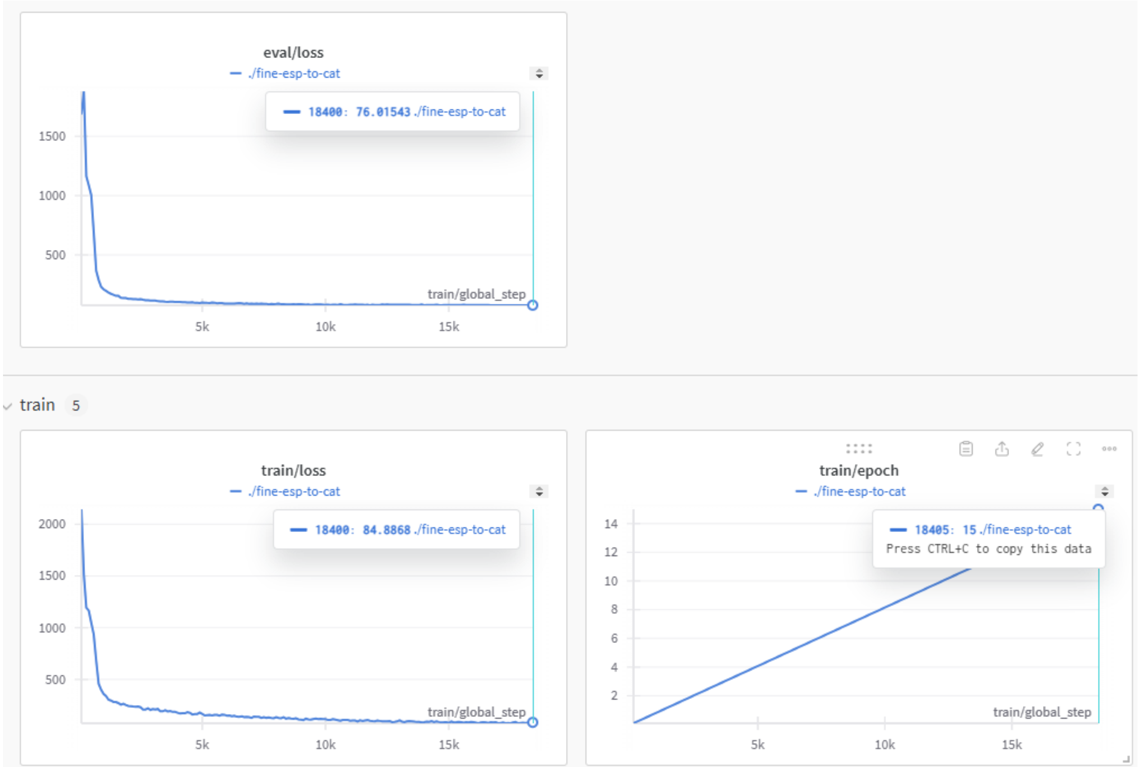
A.8 Overlay


Spectrogram with Grad-CAM Overlay
Mel Frequency Bins
0
10
20
30
40
50
60
70
Time Frames
0
50
100
150
200
250
300
350
400
Grad-CAM Intensity
0.3
0.4
0.5
0.6
0.7
Spectrogram (dB)
−100
−80
−60
−40
−20
0

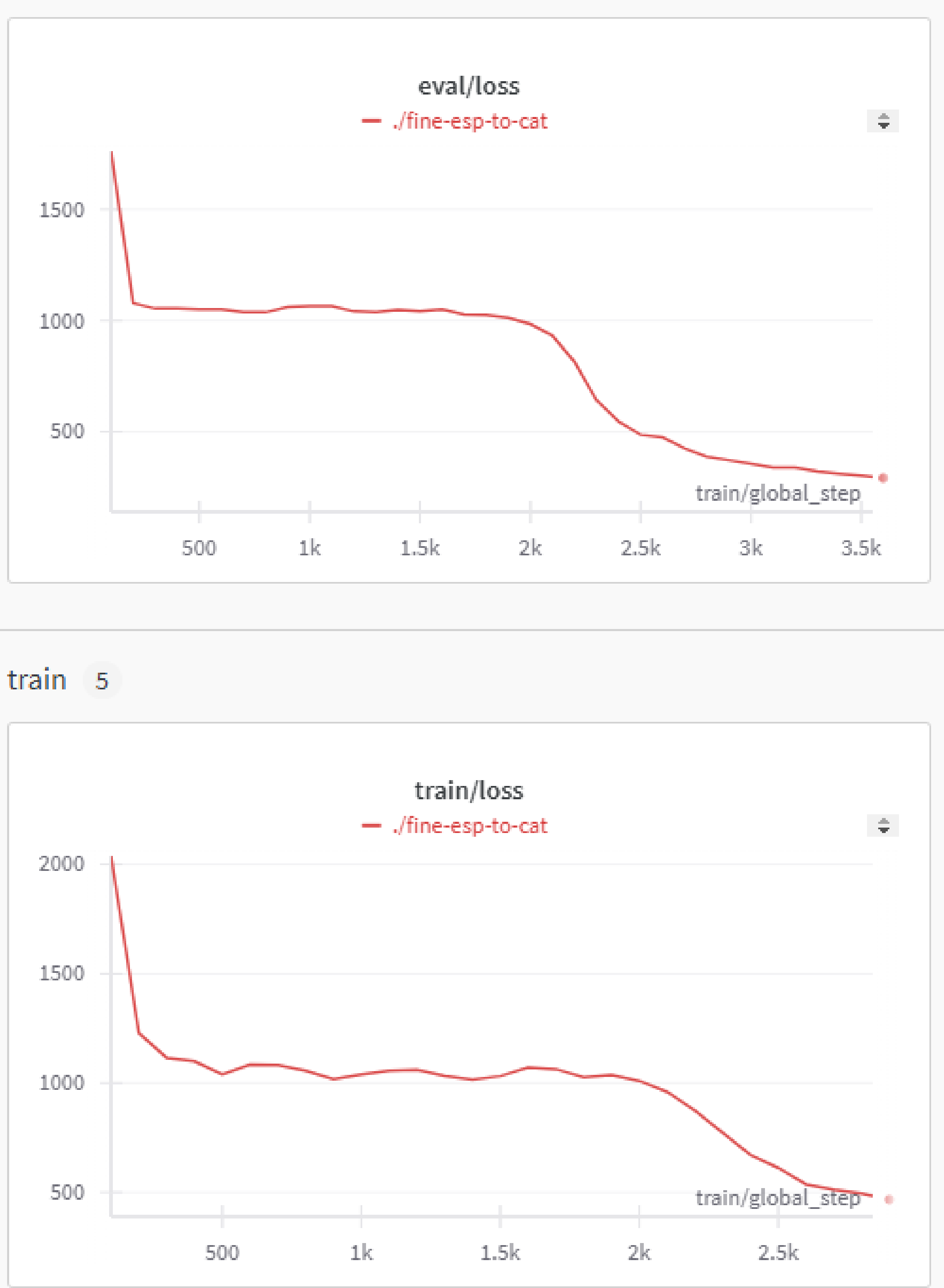

Spectrogram with Grad-CAM Overlay
Mel Frequency Bins
0
10
20
30
40
50
60
70
Time Frames
0
25
50
75
100
125
150
175
200
Grad-CAM Intensity
0.825
0.850
0.875
0.900
0.925
0.950
0.975
Spectrogram (dB)
−100
−80
−60
−40
−20
0



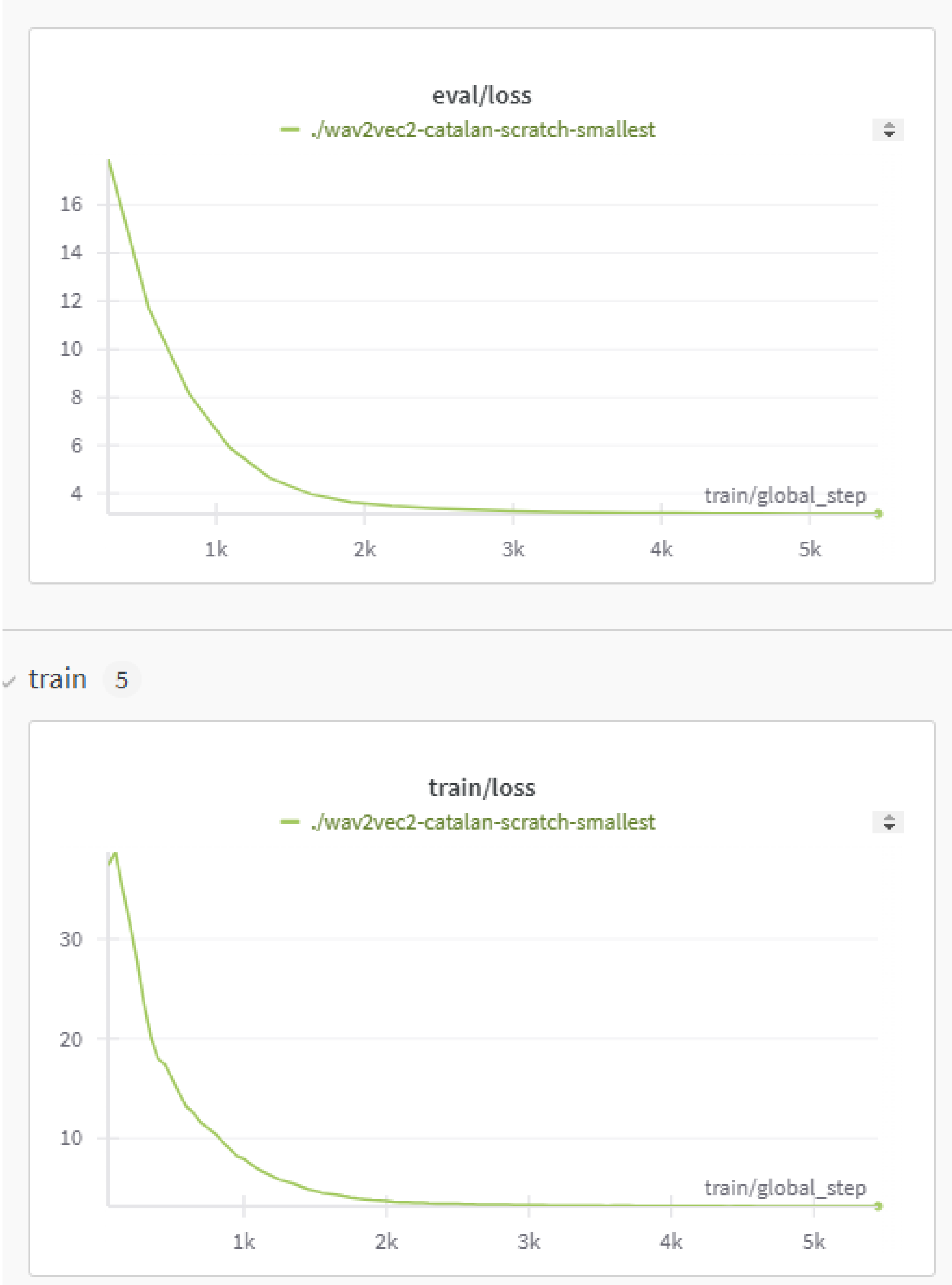
A.9 QUALITATIVE AND QUANTITATIVE RESULTS Spanish to catalan



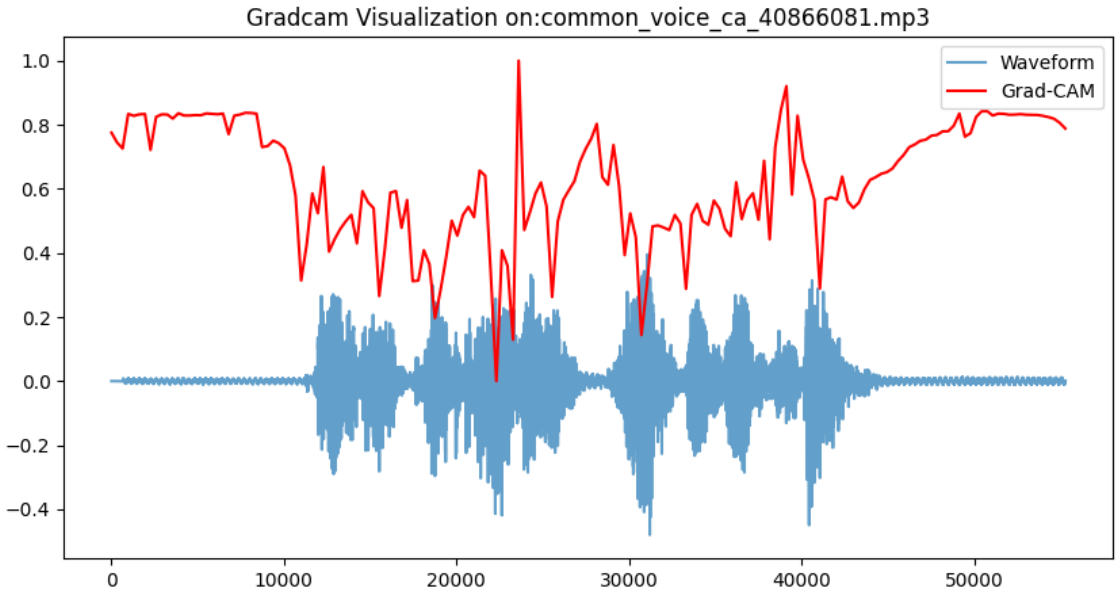
A.10 QUALITATIVE AND QUANTITATIVE RESULTS English to Catalan

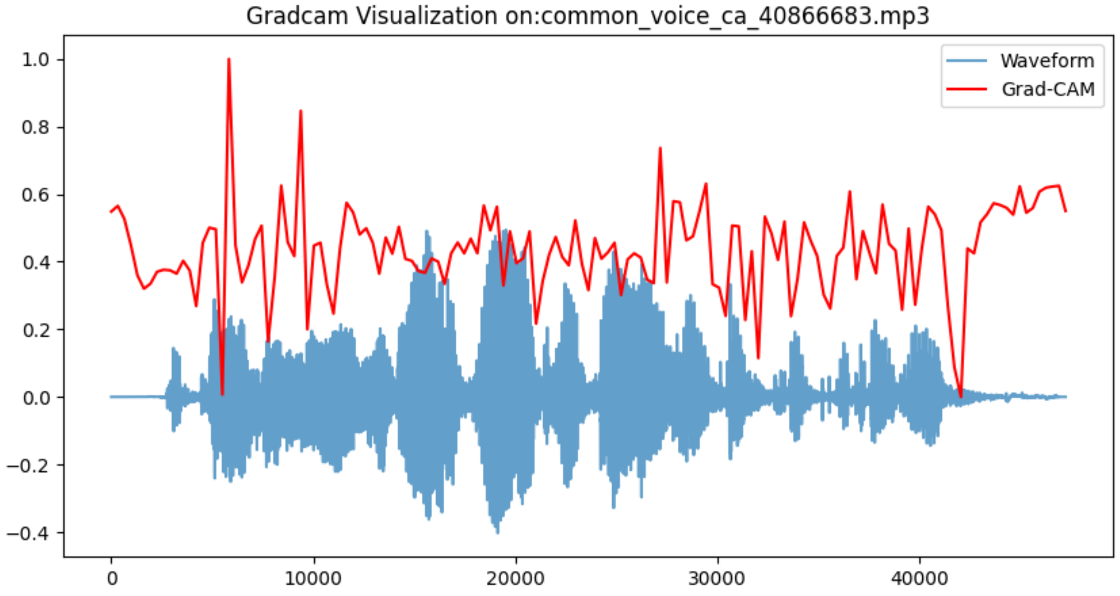


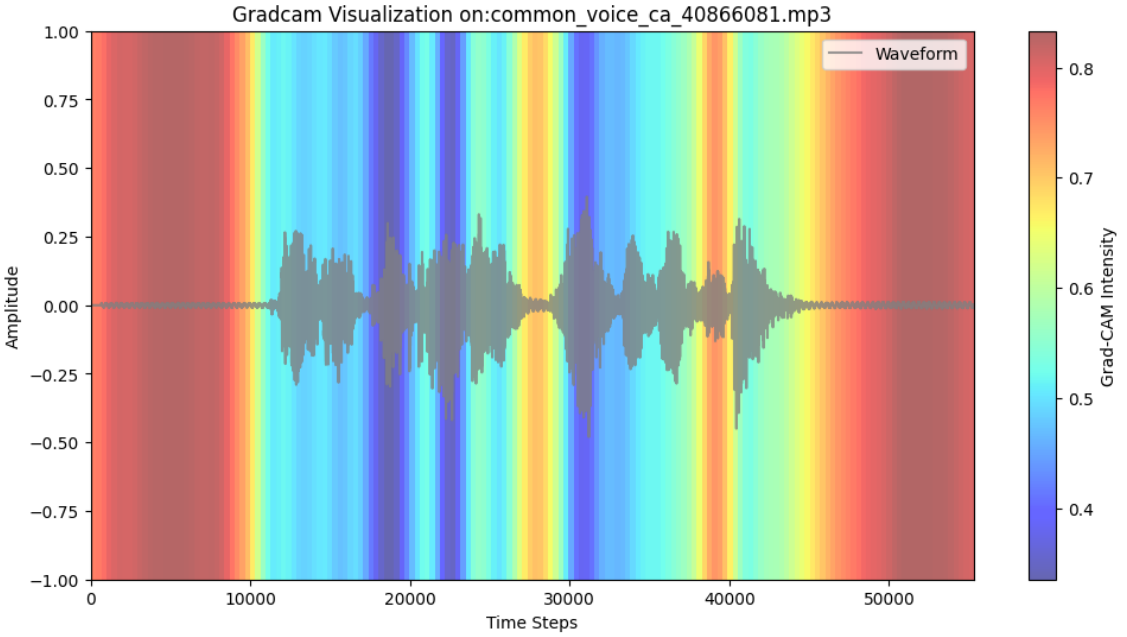
A.11 Catalan Multilingual Finetuned models in catalan

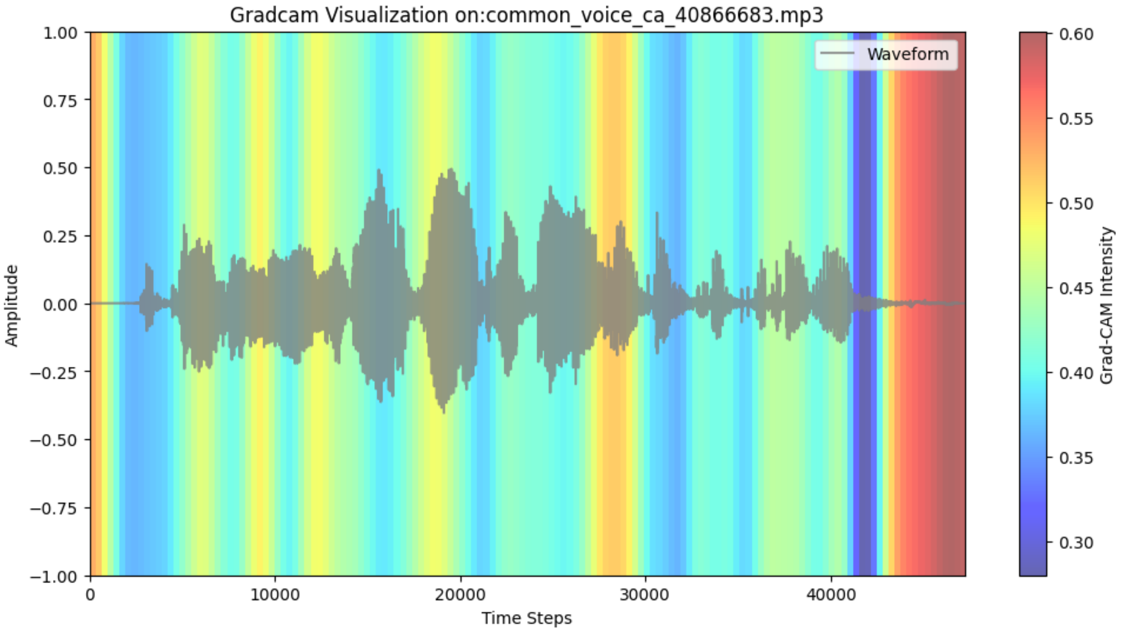


A.12 Making the Best model explain itself

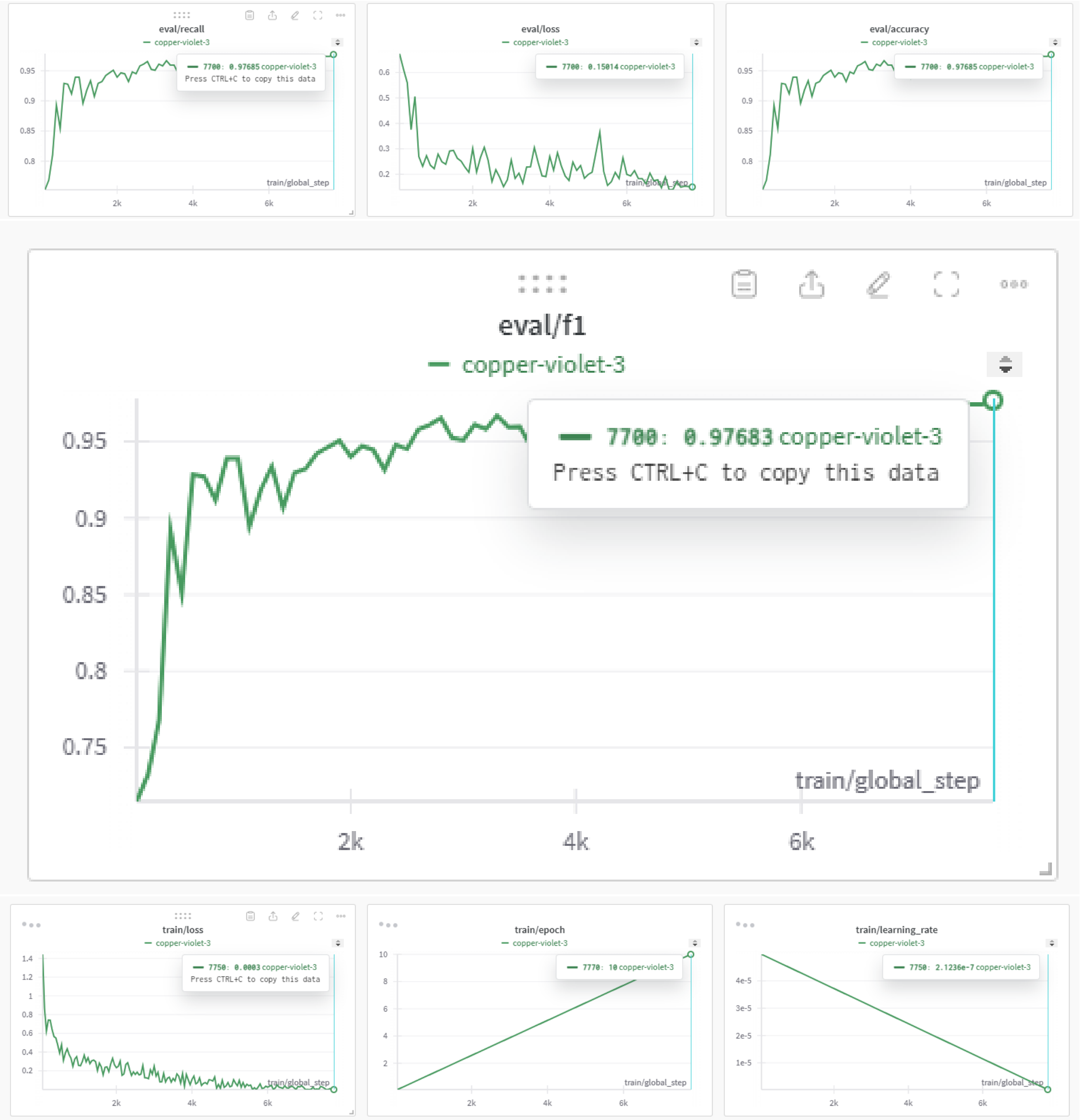




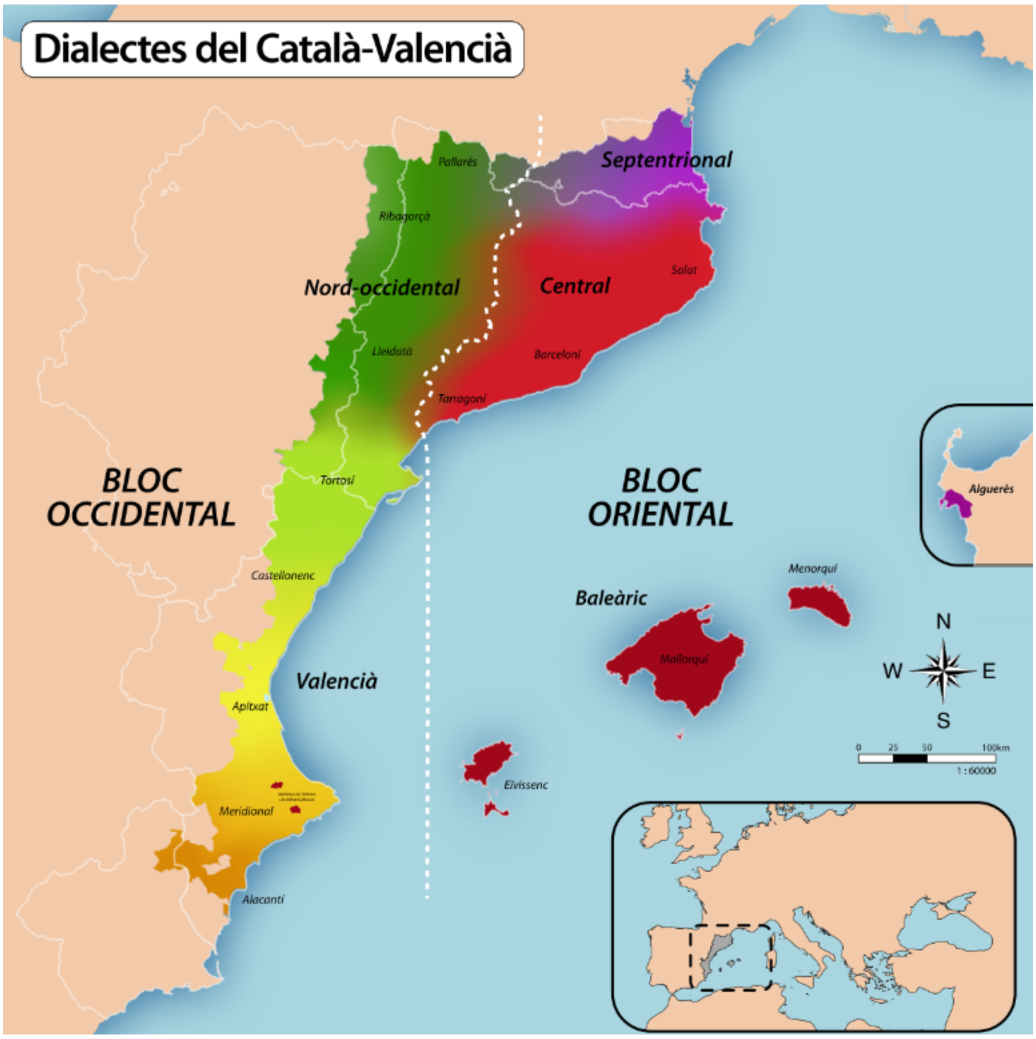

Gradcam Visualization on:common_voice_ca_40866081.mp3
Waveform
Amplitude
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
10000
20000
30000
40000
50000
Time Steps
Grad-CAM Intensity
0.8
0.7
0.6
0.5
0.4


Gradcam Visualization on:common_voice_ca_40866683.mp3
Waveform
Amplitude
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
10000
20000
30000
40000
Time Steps
Grad-CAM Intensity
0.60
0.55
0.50
0.45
0.40
0.35
0.30

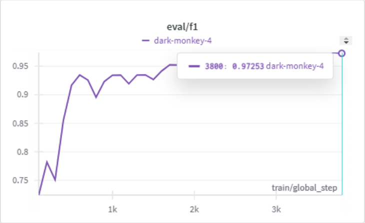
A.13 Accent Recognition

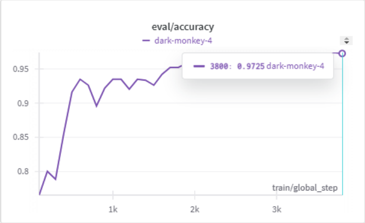


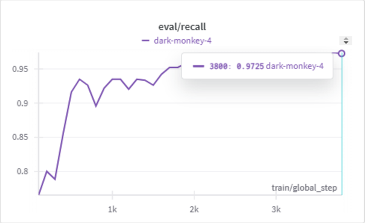
A.13.1 Dialect Map

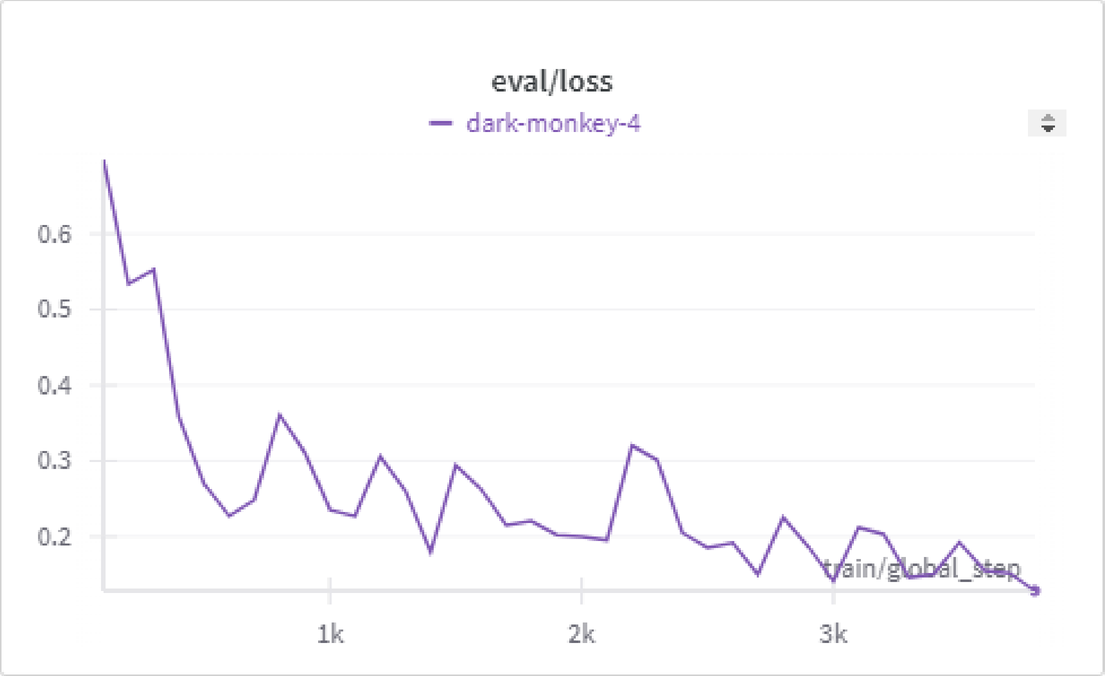


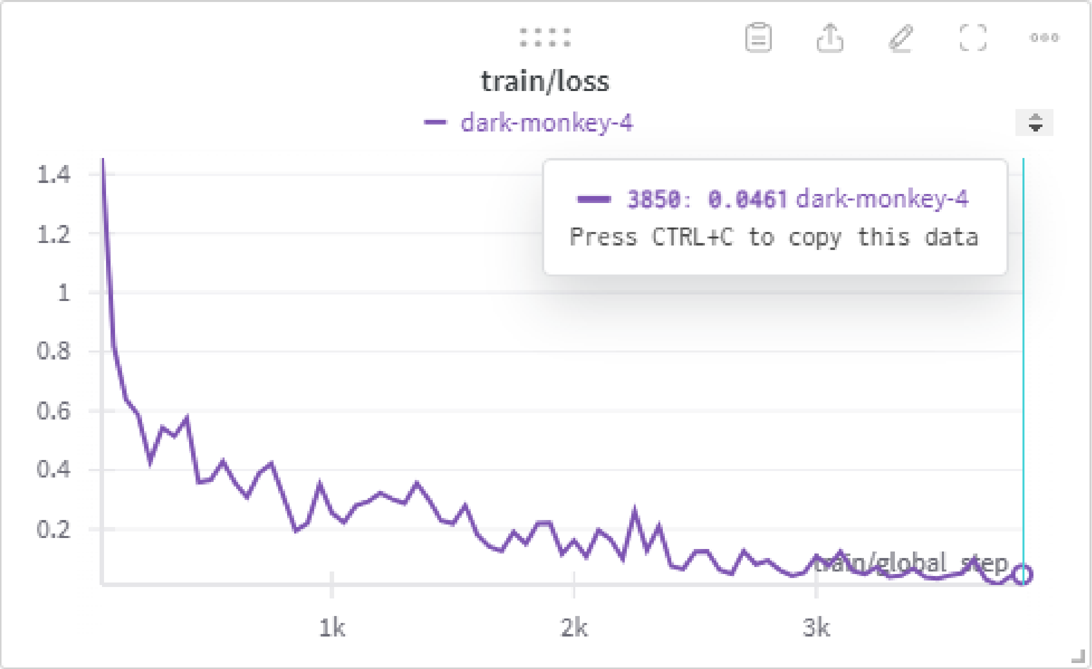
A.13.2 Gender Recognition graphs


eval/f1
dark-monkey-4
3800: 0.97253 dark-monkey-4
train/global_step
1k
2k
3k


eval/accuracy
dark-monkey-4
3800: 0.9725 dark-monkey-4
train/global_step
1k
2k
3k


eval/recall
dark-monkey-4
3800: 0.9725 dark-monkey-4
train/global_step
1k
2k
3k


eval/loss
dark-monkey-4
train/global_step
1k
2k
3k


train/loss
dark-monkey-4
3850: 0.0461 dark-monkey-4
Press CTRL+C to copy this data
train/global_step
1k
2k
3k