CS 7357 Project Report

Matthew McNair, Pari Mendpara, Kunal Shenoi

{mmcnair7, pmendpa1, kshenoi1} @students.kennesaw.edu

Objective:

Our objective in this project is primarily to improve model accuracy. In our initial analysis, we found that there were issues in the quality, amount, and processing of the training data. In its current state, DementiaNet is incomplete. Through modifications to data augmentation pipelines and the network as a whole, we intend to improve predictive accuracy as well as make efforts to maintain results while improving computational efficiency.

Problem Statement:

The existing DementiaNet framework presents several limitations. Training on a limited dataset increases the risk of overfitting and reduces generalization capability. Additionally, the current architecture does not account for computational constraints, as the transformer encoder lacks efficiency optimizations such as selective layer freezing. While the model demonstrates competent classification performance, accuracy remains suboptimal due to deficiencies in the preprocessing and training pipeline. Though we use the same dataset, the proposed modifications aim to address these gaps by improving both computational efficiency and classification accuracy.

Status of the Research:

Recent advances in speech recognition have demonstrated the importance of data augmentation in improving model robustness and generalization, particularly for low-resource tasks. Park et al. [1] introduced SpecAugment, a simple yet effective augmentation method that applies time warping, frequency masking, and time masking directly to log mel spectrograms. Their approach converted speech recognition from an overfitting to an underfitting problem, enabling state-of-the-art results on LibriSpeech and Switchboard benchmarks even without language models. For multimodal systems, Oneață and Cucu [2] demonstrated that audio-level augmentations encourage models to leverage visual information more effectively, showing consistent improvements across multiple datasets when perturbing the audio signal during training. Additionally, Bartelds et al. [3] explored data augmentation in severely constrained settings with only 24 to 192 minutes of training data, demonstrating that self-training approaches can yield relative WER reductions up to 20.5% when fine-tuning pre-trained models on limited data.

DementiaNet currently employs limited augmentation strategies, using only basic padding and random subsampling of 32-second clips from a collection of YouTube videos. With merely 227 training samples, DementiaNet operates in a environment even more resource-constrained than the low-resource scenarios examined by Bartelds et al. The absence of sophisticated masking techniques, such as SpecAugment's frequency and time masking, represents a significant missed opportunity. Park et al. showed that these techniques are most effective when they transform an overfitting problem into an under-fitting problem, precisely the situation facing DementiaNet given its aggressive regularization needs (dual dropout layers) and small dataset. Furthermore, the current approach of random subsampling may introduce inconsistency in temporal context, whereas structured masking approaches could provide more systematic robustness to partial signal loss.

While Oneață and Cucu [2] demonstrate that multimodal approaches can improve speech recognition performance by 8 to 31% when visual context is semantically aligned with audio, dementia detection tasks have yet to explore this potential systematically. The grounding between spoken content and visual stimuli in interview settings, such as picture description tasks commonly used in cognitive assessments, suggests that multimodal fusion could provide complementary signals for classification. The trade-off, as noted in [2], is that multimodal gains diminish when the

unimodal baseline is already strong. Thus, augmentation improvements to the audio pipeline must precede any multimodal extensions to maximize their effectiveness. Additionally, Bartelds et al. [3] found that gains from data augmentation are most pronounced when starting with very limited data (24 to 48 minutes), after which performance improvements plateau, suggesting that DementiaNet's current data environment is precisely where augmentation techniques would be most beneficial. However, their experiments with traditional augmentation techniques such as adding noise, pitch shifting, and simulating far-field speech did not improve performance and were discarded in favor of self-training and TTS-based approaches.

Design:

There are several changes that we want to add to this project in order to achieve our objectives. Firstly, we intend to implement block-wise time masking on the transformer input. This is inspired by SpecAugment's time masking but adapted to the alternate data type and processing flow. We will also implement frequency masking, noise injection, and speed perturbation during training to simulate real world variations. Some of the augmentation types may not provide effective improvements as identified by Bartelds et al [3], but testing each will be necessary given the different goals in usage. Additionally, we want to focus on the audio preprocessing and clean unimportant background noise to maximize the effective impact of our data augmentation. We intend to apply a dropout on latent features to the output of the frozen CNN prior to passing to the transformer. Rather than using random subsamples of the audio files, we intend to use sliding windows and energy-based selection to generate multiple, higher quality chunks for each audio file. Adjustments to the pooling will be made, where we believe the current implementation can cause unwanted dilution of important moments by averaging them with irrelevant segments. We will test alternatives such as the inclusion of std in mean pooling and use of attention pooling by using learnable weighted averages. Optimization will be performed using AdamW with learning rate scheduling, and performance is evaluated using accuracy, precision, recall, and F1-score. In further attempts to improve computational efficiency, we will reduce the size of the feature representations to lower memory and processing requirements. Then we check whether the model can still perform almost as well while using fewer resources.

Division of Work:

Our team's plan is to split up the work, but we will keep in contact and meet on Teams periodically to discuss our ideas and progress. Our goal is to have the next part done by March 14. By that date, we want to be done with time masking and data augmentation. We will split up the work as a group and plan to have our individual parts done by March 7th so that we can come together and finalize everything by March 14th. After that date, we will split up everything we have left that we need to get done by May 5th.

Methodologies:

(i) Dataset:

The dataset consists of 227 labeled audio recordings from the DementiaNet framework, each paired with a binary label indicating dementia or control status. All recordings are resampled to 16,000 Hz mono-channel audio using torchaudio's resample() function. Because the dataset is on the smaller side, there is a high risk of overfitting, so our pipeline aims to curb this risk by implementing improved preprocessing and data augmentation strategies. Rather than randomly subsampling a single 32-second clip per recording, we apply a sliding window over each file with window size W = 10 seconds and hop size H = 5 seconds, yielding an overlap of 50%. For a recording of duration D seconds, this produces approximately floor((D - W) / H) + 1 segments. On a typical 2-minute interview recording,

this generates roughly 21 segments per file, increasing the effective training set size by an order of magnitude. All segments inherit the label of their parent recording. To discard silence-dominated windows, each segment is evaluated by its root mean square (RMS) energy, computed as the square root of the mean of squared sample values across the window. This allows the model to learn from a broader set of variations in speech and better capture the small changes associated with dementia. Augmentation is applied stochastically at training time and never at validation or test time, so the model sees a different view of each segment each epoch.

(ii) Implementation:

We follow the pipeline described in the DementiaNet architecture that processes speech recordings through a feature extractor then passes them through a classifier for dementia prediction. Raw waveform segments are passed through the convolutional encoder of a pre-trained wav2vec 2.0 base model, loaded via facebook/wav2vec2-base from Hugging Face Transformers. We expand upon the original architecture by adding additional data augmentation techniques such as time masking, frequency masking, noise injection, and speed perturbation that are meant to simulate real world speech variations [1]. In addition, inspired by SpecAugment's time masking [1] but adapted to the embedding domain, we apply block-wise masking directly to the CNN output features before the Transformer processes them, where a contiguous block of up to 50 frames (~1 second) is zeroed out to force the model to infer context from surrounding frames rather than relying on any single contiguous speech region. We also apply dropout to the latent features extracted by the frozen CNN to improve regularization before passing them to the Transformer. We also plan to improve pooling by experimenting with mean and standard deviation-based pooling as well as attention-based pooling. We will train the model using the AdamW optimizer with learning rate scheduling. The primary deep learning framework used is PyTorch which is responsible for the Neural Network, tensors and backproprogation architecture. The hugging face transformers library is also used to load the wav2vec model for extracting features from the audio signals. Numpy and Pandas are used for data manipulation and skikit-learn is used for the evaluation metrics. The torchaudio library will also be used to handle waveform processing and the audio transformations.

(iii) Evaluation Metrics:

Four evaluation metrics are used to assess model performance: accuracy, precision, recall, and F1-score. Accuracy measures the proportion of correct predictions across all classes, providing a general indicator of overall model performance. Precision quantifies the ratio of true positive predictions to all positive predictions made by the model, reflecting how reliably the classifier avoids false positives. Recall measures the ratio of true positives to all actual positive instances, capturing the model's ability to detect all relevant cases. The F1-score is the harmonic mean of precision and recall, providing a single balanced metric particularly useful when the two are in tension. We hope to improve our model optimization by properly utilizing the ideal use of each metric [4].

Experiments:

(i) Hypothesis:

This work hypothesizes that enhancing the preprocessing pipeline will yield improved classification performance. Specifically, segmenting audio clips into a greater number of training samples is expected to expose the model to finer-grained speech patterns, resulting in more accurate classifications. To evaluate this, the original and modified pipelines are compared using the evaluation metrics described above. The hypothesis is supported if the modified pipeline improves classifier performance while maintaining or reducing computational cost, reflected by gains in accuracy and F1-score.

(ii) Figures/tables:

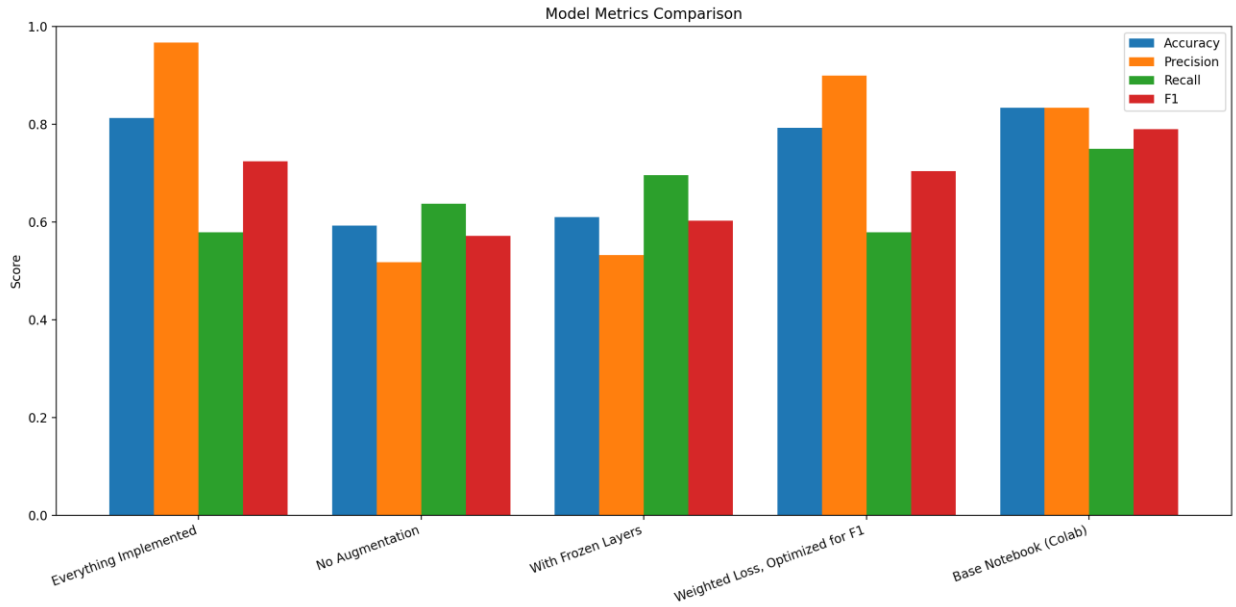

Fig 1. Visual comparison of model performance across accuracy, precision, recall, and F1-score for all executable configurations.

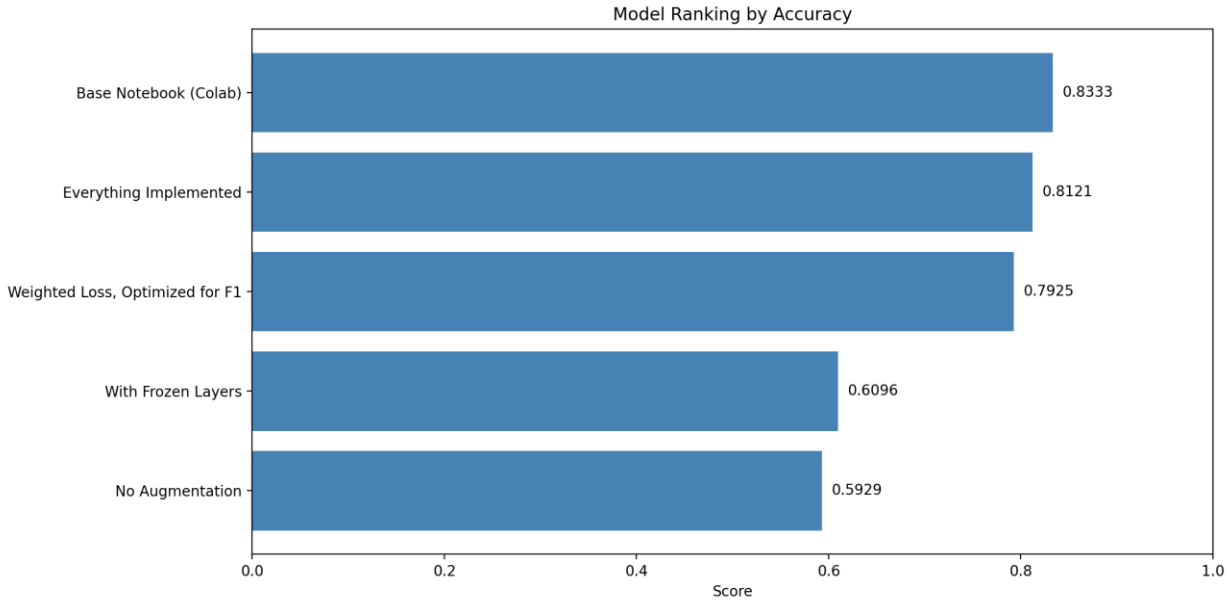

Fig 2. Model ranking by accuracy (overall classification correctness). The Base Notebook (Colab) configuration achieves the highest accuracy, followed by Everything Implemented and Weighted Loss (F1-optimized), while No Augmentation and Frozen Layers underperform.

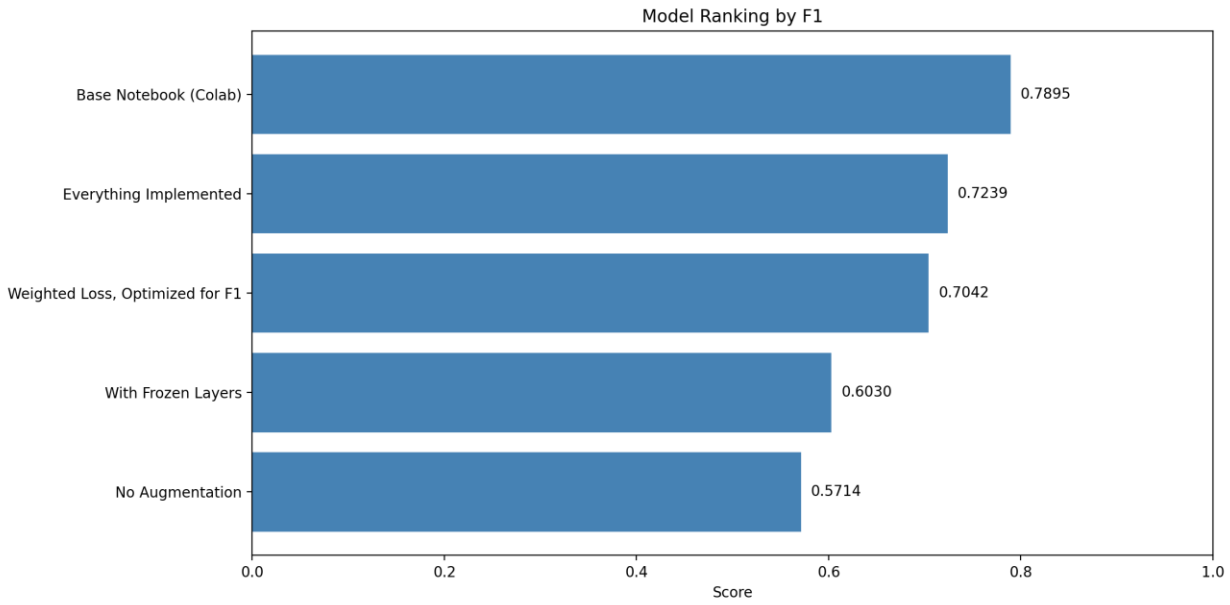

Fig 3. Model ranking by F1-score (balance between precision and recall). Base Notebook (Colab) achieves the best overall balance, with Everything Implemented second and Weighted Loss close behind, indicating competitive but less balanced trade-offs.

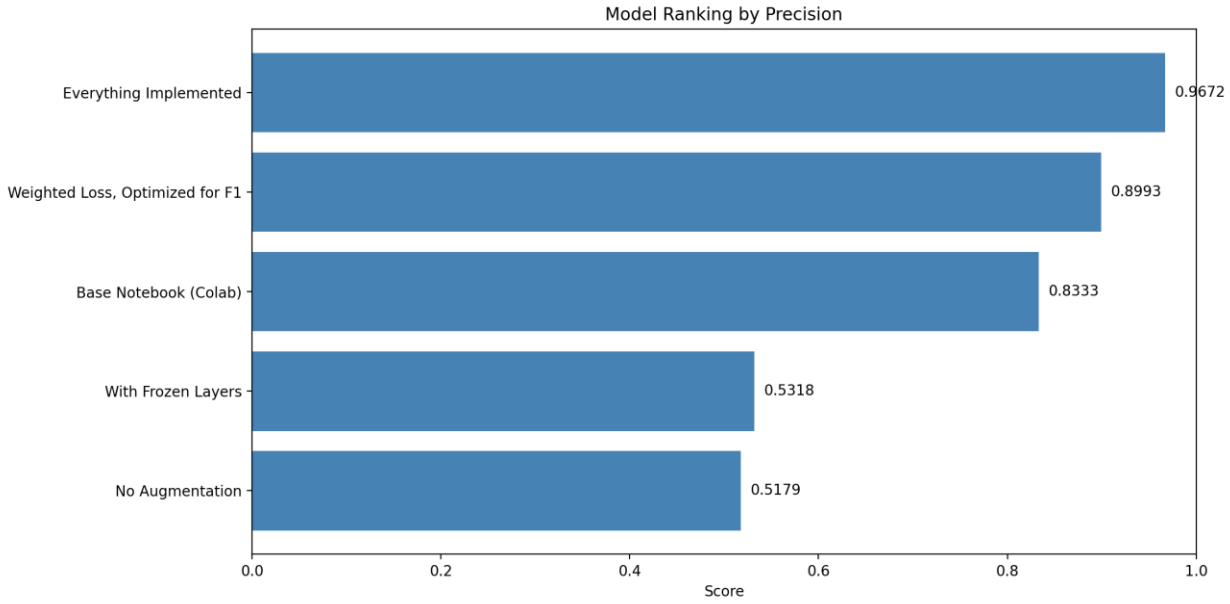

Fig 4. Model ranking by precision (reliability of positive dementia predictions). Everything Implemented leads with the highest precision, indicating the strongest control of false positives, with Weighted Loss second and Base Notebook (Colab) third.

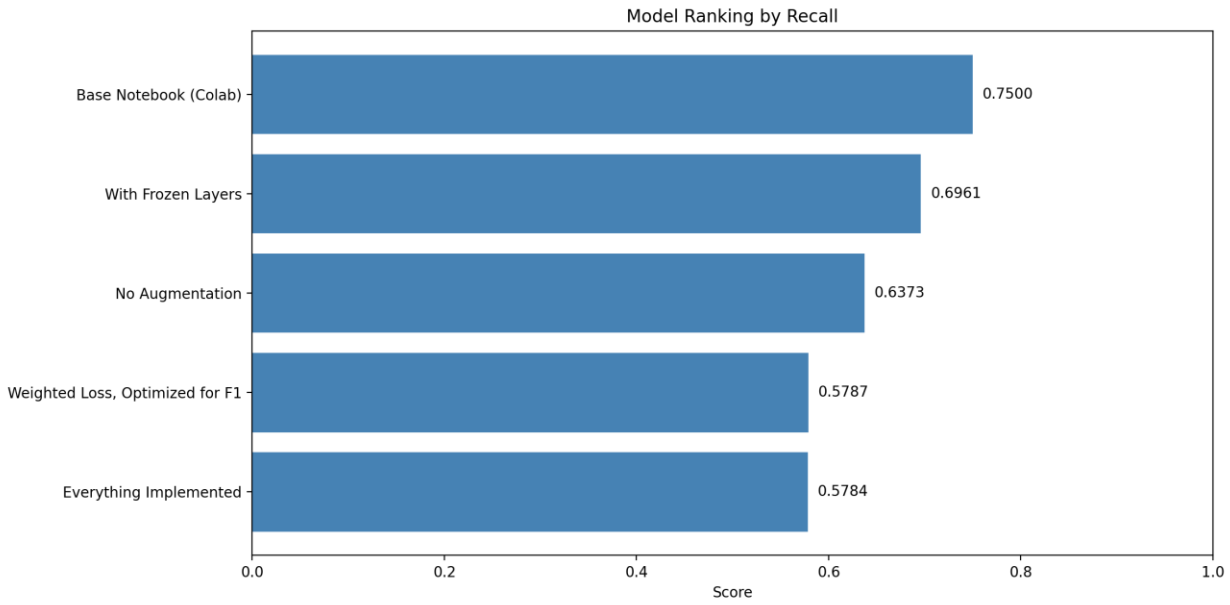

Fig 5. Model ranking by recall (ability to detect dementia cases). Base Notebook (Colab) and Frozen Layers show the strongest recall, whereas Everything Implemented and Weighted Loss trade recall for higher precision.

Model	**Accuracy**	**Precision**	**Recall**	**F1**	**Avg**	**Notes**
Everything Implemented	0.8121	0.9672	0.5784	0.7239	0.7704	
No Augmentation	0.5929	0.5179	0.6373	0.5714	0.5799	
No Sliding Window	-	-	-	-	-	Failed (excess resources)
With Frozen Layers	0.6096	0.5318	0.6961	0.6030	0.6101	
Weighted Loss, Optimized for F1	0.7925	0.8993	0.5787	0.7042	0.7437	
Base Notebook (Colab)	0.8333	0.8333	0.7500	0.7895	0.8015	

Fig 6. Numeric comparison of model performance across accuracy, precision, recall, and F1-score for all executable configurations.

(iii) Analysis:

In this stage of our project, we compare multiple pipeline and training variants to evaluate whether our modifications improve classification quality while remaining computationally feasible. Our results show that the modified configurations produce meaningful shifts in model behavior across accuracy, precision, recall, and F1-score, though improvements are not uniform across all metrics.

The configuration with everything implemented produces the strongest precision, indicating more reliable positive dementia predictions and fewer false positives. However, this gain is accompanied by lower recall relative to other variants, showing a clear precision-recall trade-off. The With Frozen Layers configuration exhibits the opposite tendency: recall improves, but precision and overall balanced performance decline. The Weighted Loss, Optimized for F1 setup improves precision compared to weaker baselines and remains competitive in F1-score, but it still does not fully close the gap in balanced performance against the strongest baseline result.

The Base Notebook (Colab) remains the top-performing executable setup in overall balance, with the highest accuracy and F1-score among the reported runs. At the same time, this configuration requires higher resource allocation than our constrained local environment. In contrast, No Sliding Window is not included in the ranked chart interpretation because it fails to execute under limited hardware resources (8GB memory/VRAM constraint), preventing completion of a comparable evaluation run.

From our perspective, these outcomes support the core motivation of the project: preprocessing and training modifications can materially influence performance in low-resource dementia detection, but gains must be interpreted jointly with compute feasibility. Our current findings suggest that no single change is sufficient on its own; instead, the best path forward is targeted refinement of augmentation and optimization choices to preserve precision gains while recovering recall, all under realistic resource limits.

Discussion:

(i) Scope of this study:

The scope of this study was to see if making changes to the given code and data will help improve the evaluation metrics. Which includes accuracy, precision, recall, and f1-score. We tried several different methods to improve these metrics, such as data augmentation, which was our main idea going into this. As well as segmentation, which is going off our hypothesis and objective.

(ii) Implications of the results; trade offs:

The results show that with our changes, precision increased. This means that there are some tradeoffs. For example, if we wanted to use layer freezing which really helped with efficiency, but it had the tradeoff of not being as accurate as we wanted it to be. Meaning that layer freezing will not have the overall better results if you are looking to make everything as accurate as can be. Which is usually the case. The results we got are helpful in some ways in terms of evaluation metrics, but they do have some issues that come with it, and then we have to decide if it is worth the tradeoffs.

(iii) Limitations:

Some limitations that we had while doing this whole process was time. It took a long time every time we wanted to run the full code so that means we could not try as many things as we wanted to. Our data was also pretty limited, with more data this process would have definitely been better. As well as the clips that we used were not perfect as well and that might have lead to some skewed data. These are some things that could have affected how our results turned out.

(iv) Future directions:

In the future, we believe that technology is going to get better, faster, and stronger. Therefore, we will be able to go even further with this research and process. One thing specifically that we should be able to do better is cut the amount of time it takes to run the code, therefore we can test more theories. Another direction that we can go in the future is being more specific as to where exactly in the audio there are issues. This will help us see exactly what we need to look at in order to get better. This technology is changing so quickly, therefore I am positive we will be able to do so much more with this research as the improvements occur.

Summary of contributions:

(i) New problem:

This work addressed the automated dementia detection from speech under an extreme data scarcity constraint, with only 227 labeled recordings. Unlike general speech recognition tasks where data collection is relatively straightforward, clinical dementia detection requires medical oversight and structured interview protocols that make dataset expansion difficult. Despite these constraints, our results show that reliable classification is achievable without collecting additional data, primarily through smarter use of what already exists. This demonstrates that low resource clinical audio classification is a distinct and solvable problem, and one that deserves more attention in the research community.

(ii) New application, algorithmic design, engineering/theoretical framework:

We modified the original wav2vev2 speech recognition model to the clinical task of dementia recognition and classification following the DementiaNet architecture. However, the original architecture was extended in three ways. First, by applying selective freezing of the bottom transformer encoder layers to reduce computational cost while preserving high-level representational capacity. Next latent dropout was applied between the CNN encoder output and the transformer input as an additional regularization point that was not present in the original design Lastly we extended pooling framework supporting mean, mean+std, and learnable attention pooling to replace the fixed average pooling that dilutes clinically significant speech moments. A weighted loss variant was also implemented to directly optimize F1-score rather than accuracy, acknowledging the asymmetric cost structure of clinical prediction tasks.

(iii) New methodologies (e.g., data, experimental settings, analytical tools):

First, random 32 second subsampling was replaced with a structured sliding window segmentation scheme using 10-second windows and 5-second hop strides with 50% overlap, combined with RMS energy thresholding to discard near-silent segments. This expanded a typical 2 minute recording into approximately 21 usable training segments and increased the effective dataset size by an order of magnitude. Second, a stochastic multi-augmentation pipeline was introduced combining time masking, frequency masking via spectrogram roundtrip, Gaussian noise injection, and speed perturbation, applied independently at training time only with tuned per technique probabilities. Third, evaluation was measured around the clinically meaningful positive class using sklearn's precision, recall, and F1-

score with explicit positive label assignment, replacing the hand computed metrics in the original codebase that contained implementation errors and lacked zero-division protection.

(iv) New findings leading to new research topics, new principles, new paradigm:

Our key finding is that augmentation and regularization reliably improved precision but hurt recall, and weighted loss optimization did not fix it. This tells us the imbalance is structural; it comes from how the model learns from the segmented data, not something easily corrected after the fact. Our best model catches false positives well at 96.7% precision, but still misses 4 in 10 actual dementia cases, which remains the most important problem to solve going forward.

Future work should focus on loss functions that explicitly penalize missed detections, or augmentation strategies designed specifically to improve recall rather than general robustness. More broadly, this project showed that in extremely small datasets, how you slice and augment your data matters just as much as the model itself, and that lesson applies to any medical speech task where collecting more labeled data is difficult.

References

[1] D. S. Park et al., “SpecAugment: A simple data augmentation method for automatic speech recognition,” arXiv.org, https://arxiv.org/abs/1904.08779

[2] D. Oneata and H. Cucu, “Improving Multimodal Speech Recognition by Data Augmentation and Speech Representations,” *arXiv.org*, 2022. https://arxiv.org/abs/2204.13206

[3] M. Bartelds, N. San, B. McDonnell, D. Jurafsky, and M. Wieling, “Making More of Little Data: Improving Low-Resource Automatic Speech Recognition Using Data Augmentation,” *arXiv.org*, 2023. https://arxiv.org/abs/2305.10951

[4] Z. C. Lipton, C. Elkan, and B. Narayanaswamy, “Thresholding Classifiers to Maximize F1 Score,” *arXiv.org*, 2014. https://arxiv.org/abs/1402.1892