

Speech-to-Text and Tonal Analysis for Voice-based Sentiment Analysis using Small Language Modeling

Submitted in the Fulfillment of the Requirements in
CPE313 Data Science Track Elective 3:
Advanced Machine Learning and Deep Learning

Submitted by
Dejoras, Dylan James N.
Villamor, Kurt Russel A.

May 23, 2025

Speech-to-Text and Tonal Analysis for Voice-based Sentiment Analysis using Small Language Modeling

Dejoras, Dylan James N.
Computer Engineering
Technological Institute of the Philippines
Quezon City
qdjndejoras@tip.edu.ph

Villamor, Kurt Russel A.
Computer Engineering
Technological Institute of the Philippines
Quezon City
qkravillamor@tip.edu.ph

ABSTRACT

Natural language processing (NLP) and speech-to-text (STT) systems often face challenges in accurately detecting sentiment due to biases, noise, and limitations in handling spontaneous speech and low-resource languages. To address these issues, we compare the effectiveness of sentiment classification using transcribed text and raw audio. For text, we leverage pre-trained models such as MobileBERT, TinyBERT, and ELECTRA-Small, balancing performance with computational efficiency. On the audio side, we evaluate Wav2Vec 2.0, DaBloatCNN, and ResNet50. Results show that Wav2Vec 2.0 achieved the highest accuracy and F1 score (97.8%), outperforming all other models, while ELECTRA-Small led among text-based approaches. These findings highlight the strength of raw audio models in capturing emotional tone and suggest that combining both modalities could improve the robustness and fairness of sentiment analysis systems.

Keywords: *Sentiment Analysis, Wav2Vec 2.0, MobileBERT, TinyBERT, ELECTRA, speech emotion recognition, audio classification, natural language processing, multimodal learning, deep learning.*

I. INTRODUCTION

Sentiment analysis is widely used across various industries to understand customer satisfaction levels and feedback. It can utilize both text and audio data, enabling systems to make data-driven, responsive decisions that enhance user experience and operational efficiency. This process is powered by Natural Language Processing (NLP), a field in which computers are programmed to comprehend and replicate human language. NLP underpins a range of applications, including chatbots, speech recognition systems, and search engines.

Text-based sentiment analysis has become popular in many applications like analyzing social media posts and customer reviews. However, relying only on text does not always give accurate results, especially in real-life situations where emotions are also expressed through voice. For example, a sentence can sound happy or angry depending on how it is spoken. This type of emotional information cannot be captured through text alone. Because of this, voice-based sentiment analysis has become an important area of research. It uses features like tone, pitch, and speed of speech to better understand how someone feels [16]. With recent progress in deep learning, especially with models that can learn directly from raw audio, speech emotion recognition systems are becoming more accurate and reliable [17], [18]. As a result, using voice is now seen as a strong alternative to text in understanding human emotions in many systems, such as virtual assistants or call centers [19].

In this work, the researchers aim to compare the effectiveness of sentiment classification using text alone versus audio alone. The researchers want to better understand which type of input carries more emotional information and which one performs better in practical classification tasks. In order to achieve it, the researchers explore different deep learning architectures and training or fine-tuning methods for both text-based and voice-based sentiment analysis. By comparing the results of these audio-based sentiment models with text-based sentiment models, we aim to highlight the strengths and limitations of each approach.

II. REVIEW OF RELATED LITERATURE

Transformer-based models have significantly advanced the field of Natural Language Processing (NLP), with BERT and ELECTRA being among the most influential architectures. BERT (Bidirectional Encoder Representations from Transformers) is a multi-layer transformer pre-trained using masked language modeling and next sentence prediction tasks [2]. It can be fine-tuned for downstream tasks or used as a feature extractor, allowing for flexible adaptation to applications like sentiment analysis [1].

ELECTRA presents a more computationally efficient alternative by replacing tokens and training a discriminator to predict whether each token is original or replaced [4], [7]. This results in faster training and improved sample efficiency. Both models demonstrate strong performance on various classification tasks, including hate speech detection [3] and rumor identification [11].

To address the computational constraints of deploying such models, researchers have developed lightweight alternatives. MobileBERT has been proven useful for NLP tasks such as text classification and sentiment analysis [7]. It has proven effective in tasks like fake news detection and online review analysis [8], [9]. TinyBERT further compresses BERT using knowledge distillation, operating with reduced layers and hidden sizes while maintaining competitive performance [6]. It has been applied in mobile environments for smishing detection and efficient rumor detection [10], [11]. ELECTRA-Small also scales down the original model, enabling rapid training on lower-powered devices, and has been used in mental health text classification and sentiment analysis [12].

Despite the benefits of these lightweight models, integrating large language models into embedded systems and mobile platforms remains challenging. Their large parameter counts demand high memory bandwidth and powerful hardware accelerators, limiting real-time application [13], [14]. These limitations underscore the importance of balancing model performance and computational efficiency in real-world scenarios [15].

Parallel to advancements in text-based models, deep learning has made substantial progress in speech emotion recognition (SER). Traditional approaches convert audio signals into spectrograms and apply Convolutional Neural Networks (CNNs) to detect emotional cues [17], [18]. For instance, ResNet-50—a deep residual network originally used for image classification—has been successfully adapted for SER by leveraging its ability to capture fine-grained spectral features [20], [24].

Self-supervised models such as Wav2Vec 2.0 represent a newer class of SER models that learn directly from raw audio, bypassing the need for handcrafted features [21]. Fine-tuning Wav2Vec 2.0 on emotion-labeled datasets yields strong performance, especially in low-resource environments. Furthermore, multimodal models that integrate acoustic embeddings from Wav2Vec 2.0 with text embeddings from BERT have been shown to outperform unimodal counterparts, offering robust emotion classification across diverse inputs [22].

Emerging approaches also incorporate multi-feature fusion and attention mechanisms to enhance model accuracy and noise robustness [23], [24]. These techniques allow models to focus on emotionally salient regions of input data, providing improved performance without incurring the computational cost of larger networks.

III. METHODOLOGY

A. Data Preprocessing

The text sentiment analysis dataset used in this study comprises 118,334 labeled text samples, partitioned into 95,039 training instances and 23,295 test instances. These samples were gathered from four different sentiment analysis datasets, including Twitter, IMDB, and HuggingFace. Each sample is annotated with one of three sentiment labels: negative, neutral, or positive. Such datasets are concatenated and only the text and sentiment column were extracted. Furthermore, sentiment values were numerically encoded, with 'negative' represented as 0, 'neutral' as 1, and 'positive' as 2.

The Voice-based sentiment analysis dataset is the combination of four datasets: TESS, SAVEE, RAVDESS, and CREMA-D. To enable a direct comparison between voice-based and text-based sentiment analysis models, we

grouped these emotions into three broader sentiment categories. Negative sentiment includes audio samples labeled as sad, angry, fearful, and disgusted. Positive sentiment comprises happy and surprised emotions, while neutral remains as its own category.

Each raw audio file undergoes a series of preprocessing steps before being fed into the model. Initially, the raw audio is converted into a Mel spectrogram. This spectrogram is then transformed using a logarithmic scale to produce a log-Mel spectrogram. To enhance model robustness and generalization, data augmentation techniques are applied exclusively to the training set. These include pitch shifting by 70%, time stretching by 80%, adding random noise, and further spectrogram stretching.

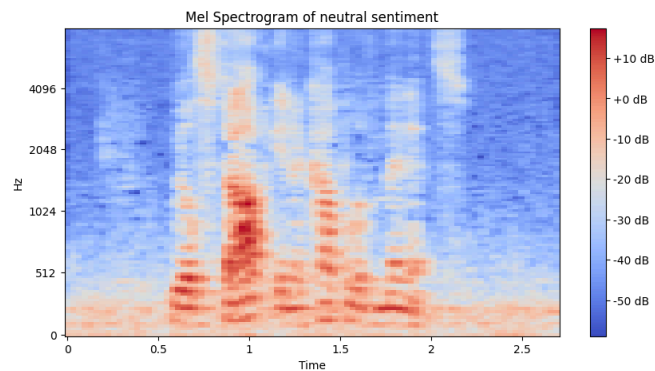


Fig 3.a.1 Mel Spectrogram of an audio file with Neutral sentiment

In the second stage of preprocessing, each log-Mel spectrogram is converted into a 224×224 RGB image to make it compatible with vision-based neural network architectures. To further enhance the variability and robustness of the training data, a series of image augmentation techniques are applied. These include Time Masking and Frequency Masking to simulate missing information along the temporal and spectral dimensions, respectively. Additionally, visual augmentations such as color jittering, random cropping, and horizontal flipping are employed to introduce subtle distortions that prevent overfitting and improve the model's generalization on unseen data.

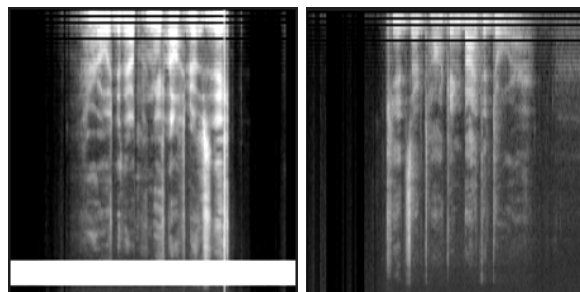


Fig 3.a.2 224x224 images of a Negative sentiment audio (train, test)

The final preprocessing method involves directly extracting the raw waveform array from each audio file, sampled at a consistent rate of 16,000 Hz.

B. Model Architectures

b.1 BERT

BERT is a multi-layer transformer-based language model that is pre-trained on large-scale datasets using tasks such as masked word prediction and next sentence prediction [2]. It is known for its strong empirical performance and straightforward architecture. The BERT framework involves two main stages: pre-training and fine-tuning. During pre-training, the model learns general language representations from unlabeled data through various tasks. Once pre-training is complete, BERT can be adapted to specific downstream tasks in one of two ways: the feature-based approach or full fine-tuning. In the feature-based method, fixed representations are extracted from BERT and used as input features for task-specific models, while in the fine-tuning approach, a task-specific layer is added on top of BERT, and all model parameters are jointly trained using labeled data from the target task [3]. This allows BERT to effectively transfer its learned knowledge to a wide range of natural language processing applications. For this Study, the researchers used variants of BERT namely; MobileBERT and TinyBERT

MobileBERT shares a similar architectural foundation to BERT [5], but it is specifically designed as a compact and efficient version optimized for mobile and resource-constrained environments. It incorporates structural modifications such as layer decomposition and bottleneck layers, which significantly reduce computational cost while maintaining competitive model performance [7]. MobileBERT utilizes a standard WordPiece tokenizer with a vocabulary size of 30,000 tokens, making it more lightweight compared to large language models that employ larger vocabularies or byte-level tokenization schemes.

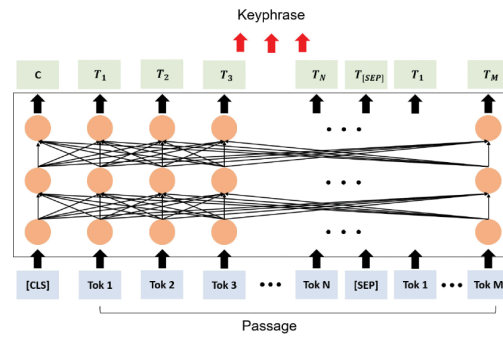


Fig 3.b.1. Transformer Architectures [15]

b.2 ELECTRA

ELECTRA is a pre-training method designed to identify altered tokens within a sentence by training a discriminator model to distinguish between original and replaced tokens [4]. Unlike traditional masked language modeling approaches, ELECTRA employs a generator network to produce high-quality negative samples, which are then used to train the main encoder to effectively detect token substitutions. This approach enables more computationally efficient pre-training while achieving superior performance with greater parameter efficiency compared to other language modeling techniques [4]. Electra primarily utilizes two neural networks: the Generator and the Discriminator.

The compact ELECTRA-Small model follows a downscaled version of the BERT-Base architecture, featuring reduced sequence length (from 512 to 128), smaller batch size (from 256 to 128), and decreased hidden dimension size (from 768 to 256), along with smaller token embeddings (from 768 to 128) [7]. These modifications allow ELECTRA-Small to achieve strong performance on NLP tasks while significantly lowering computational requirements, making it highly suitable for fast training on limited hardware.

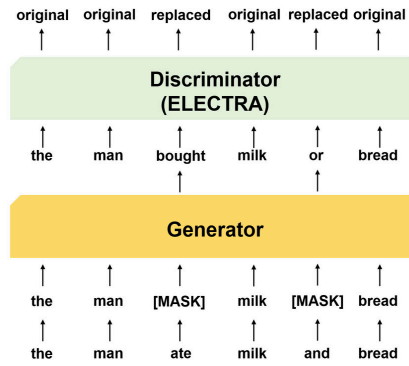


Fig 3.b.2. ELECTRA Architecture [15]

b.3 DaBloat-CNN

A custom convolutional neural network(CNN) was developed using PyTorch and made Hugging Face-compatible by subclassing the PreTrainedModel and PretrainedConfig classes. This allowed the model to be trained and evaluated using Hugging Face's Trainer API, aligning it with modern NLP and speech models. It is CNN designed to classify speech sentiment using raw audio array inputs. The architecture features five convolutional blocks, each followed by batch normalization, max pooling, and dropout layers where applicable. These layers progressively extract increasingly abstract features from the 1D input. After convolutional processing, the output is flattened and passed through two fully connected layers to predict one of the three sentiment labels: positive, negative, or neutral.

The model dynamically computes the flattened size of the convolutional output to accommodate different input lengths. This allows flexibility for varying audio lengths while maintaining architectural integrity.

Tab 3.b.1 DaBloat-CNN Structure using Pytorch and Hugging Face

Layer Block	Description	Output Shape (Example)
Conv1 + BN + Pool (x2)	Conv1d layers with BatchNorm and MaxPool reduce time dimension by 4x	[B, 512, T/4]
Dropout	Regularization (p=0.2)	[B, 512, T/4]
Conv1 + BN + Pool (x3)	Conv1d layers with BatchNorm and MaxPool further reduce time dimension by 8x	[B, 128, T/32]
Dropout	Regularization (p=0.2)	[B, 128, T/32]
Flatten + FC + BN	Flatten features and fully connected layer to 512 units	[B, 512]
FC (Output)	Final fully connected layer to 3 output logits	[B, 3]

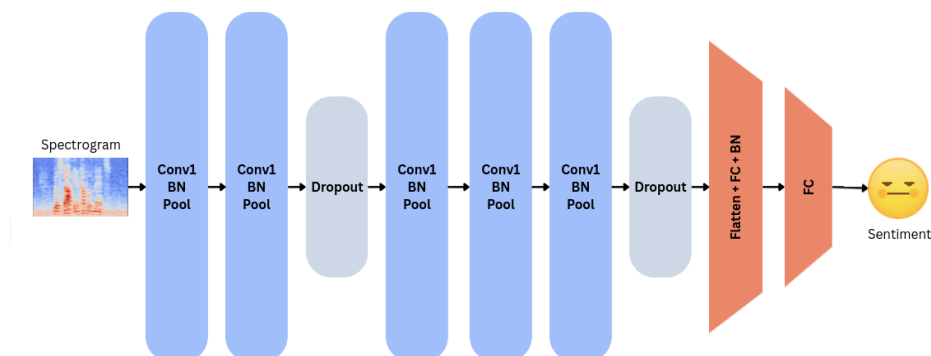


Fig 3.b.3 DaBloat-CNN Structure

b.4 ResNet50

ResNet50 is a popular deep learning model designed for image recognition. It uses a special technique called “residual learning” to help train very deep networks by allowing layers to skip connections, which prevents the problem of vanishing gradients. In this project, ResNet50 is used as a feature extractor to analyze spectrogram images of speech, helping the model learn important audio patterns for classification.

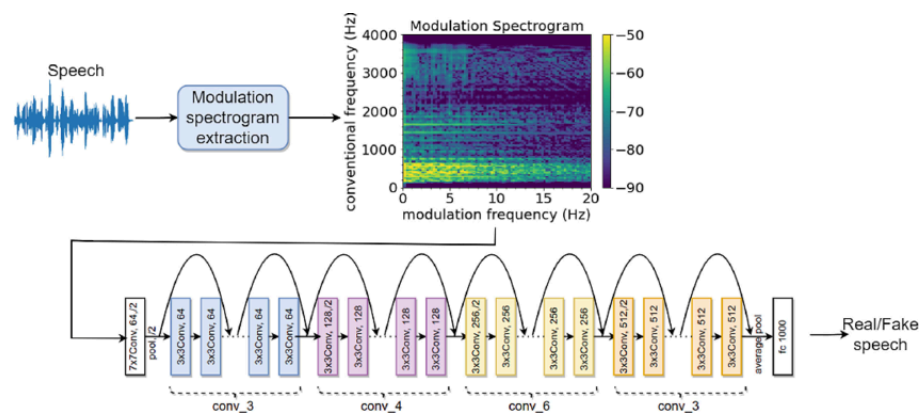


Fig 3.b.4 Sample Spectrogram in Fine Tuning ResNet50

b.5 Wav2Vec 2.0

Wav2Vec 2.0 is a state-of-the-art self-supervised model designed to process raw speech audio without relying on handcrafted features. It learns directly from the raw waveform by first pre-training on large amounts of unlabeled audio data. During this pre-training phase, the model masks parts of the audio signal and learns to predict these masked sections using the surrounding context. This helps the model develop rich and general speech representations without any labeled data.

After pre-training, Wav2Vec 2.0 is fine-tuned on smaller, labeled datasets for specific tasks such as speech recognition or emotion classification. The model architecture combines a convolutional feature encoder, which extracts local acoustic features, with a Transformer network that captures long-range dependencies in the speech signal. This design allows Wav2Vec 2.0 to effectively understand complex speech patterns, making it highly accurate for various speech-related applications.

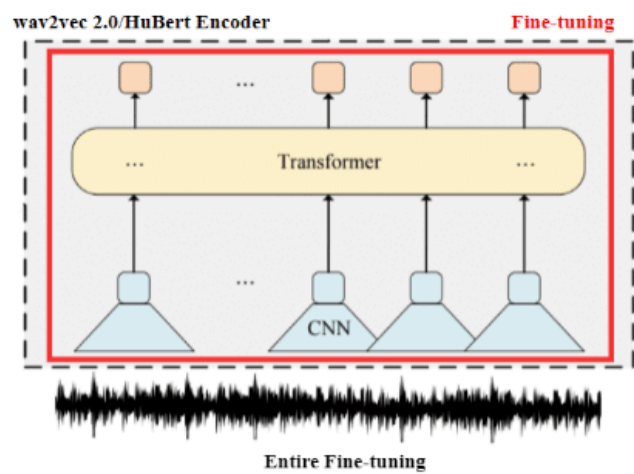


Fig 3.b.5 Wav2Vec 2.0 Fine Tuning

C. Training and Fine-Tuning Strategy

The text-based models are trained using PyTorch, with data loading handled through a custom `SentimentDataset` class that enables efficient batching and GPU acceleration. A batch size of 16 was used, along with the AdamW optimizer at a learning rate of 2×10^{-5} . A linear learning rate scheduler was applied over 12 training epochs to improve convergence.

The training phase involved fine-tuning three different models namely; CNN, ResNet50, and Wav2Vec 2.0—each tailored to a specific type of input: spectrograms (1D and 2D) and raw audio. All models were trained using Hugging Face's Trainer API, which provided a consistent interface for managing training, evaluation, and checkpointing.

CNN was trained for 15 epochs with a batch size of 64 and a learning rate of $5e-4$. Mixed-precision training (fp16) was enabled to accelerate performance and reduce memory usage. Evaluation, saving, and logging were all performed per epoch, and the best model was selected based on validation accuracy. The training process used dropout, batch normalization, and cross-entropy loss to improve generalization and stability.

The ResNet50 model, adapted for 2D spectrogram image classification, was trained over 15 epochs with a batch size of 32 and a learning rate of $1e-5$. Evaluation and saving were done at the end of each epoch, and the model checkpoint with the best F1-score was retained. A custom metric function computed accuracy and weighted F1, and a processor was used to ensure input consistency.

Wav2Vec 2.0, which directly processes raw waveforms, training was performed over 15 epochs with a small batch size of 4 and the same learning rate of $1e-5$. Regularization techniques such as weight decay and a warm up schedule were applied to stabilize the training process. Evaluation and checkpointing occurred every 500 steps, with logging every 100 steps. Mixed-precision (fp16) was used for efficiency, and the best model was chosen based on the highest F1-score. The built-in feature extractor handled raw waveform preprocessing, enabling end-to-end training without handcrafted features.

D. Evaluation Metrics and Experimental Setup

The primary evaluation metrics included accuracy, loss, and weighted F1-score, providing a balanced view of overall prediction accuracy and performance across imbalanced emotion classes. A detailed classification report was generated for each model, presenting precision, recall, and F1-score per class. Confusion matrices were used to visualize misclassifications, helping to identify systematic errors or frequently confused emotions. To measure inter-rater reliability beyond chance, Cohen's kappa coefficient was computed.

Furthermore, training time was included as a critical performance indicator to assess the computational efficiency of each model. This metric is particularly relevant when considering the practical deployment of emotion recognition systems, where faster retraining cycles and resource constraints play a significant role. By incorporating training duration into the evaluation, the study ensures a more holistic comparison—balancing predictive performance with scalability and computational cost.

E. Deployment.

Each of the models developed during experimentation across both text-based and audio-based categories were saved and versioned using appropriate model serialization formats. These trained models were then pushed to the researchers’ respective GitHub repositories for documentation, reproducibility, and public access. After evaluation, the researchers selected the best-performing model from each category based on a combination of results of the evaluation metrics earlier

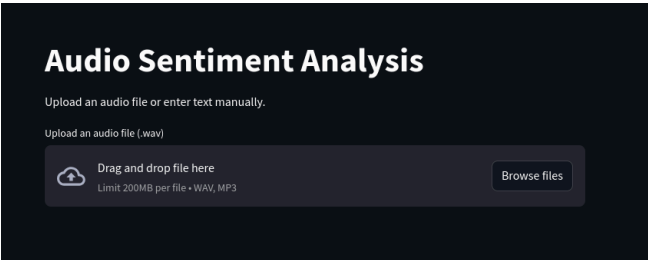


Fig 3.b.6 Website Application Interface

For deployment, the selected models were integrated into a cloud-based interactive interface using Streamlit, a lightweight Python framework for building data science web applications. This enabled real-time emotion or sentiment inference through audio upload. The deployment not only provides an accessible demonstration of the model's capabilities but also serves as a proof-of-concept for practical, real-world integration of effective NLP and audio-based systems.

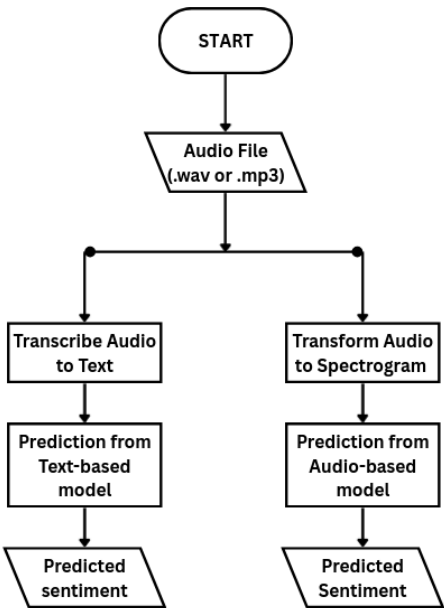


Fig 3.b.7 Website Application Flowchart

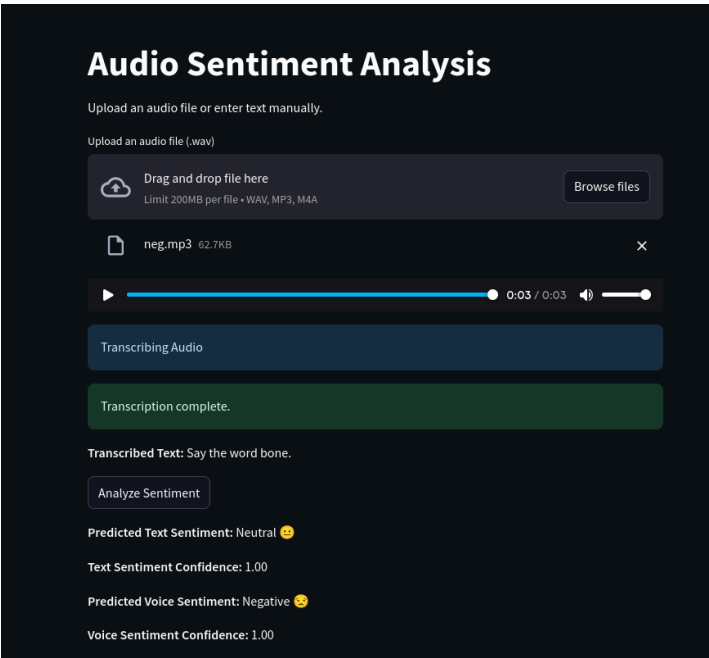


Fig 3.b.8 Website Application testing

The program begins by accepting an audio file input in either .wav or .mp3 format, which is then processed through two parallel pipelines to predict sentiment. In the first pipeline, the audio is transcribed into text using the ‘whisper’ library, and the resulting text is passed to the best-performing text-based sentiment model to determine the sentiment. In the second pipeline, the same audio is transformed into a log-Mel spectrogram and fed into the best-performing audio-based model to predict sentiment based on vocal features. This dual-path setup enables sentiment prediction using both linguistic content and acoustic cues, enhancing the model’s overall interpretability and accuracy.

IV. RESULTS AND DISCUSSION

A. Text-Based Models for Sentiment Analysis

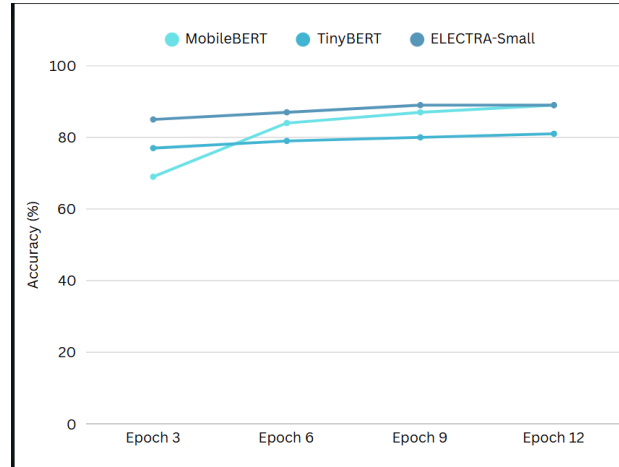


Fig 4.a Validation Accuracy

Tab 4.b Overall Evaluation Metrics

Metric	MobileBERT	TinyBERT	ELECTRA-Small
Accuracy	89%	81%	89%
Macro F1	0.8814	0.7979	0.8884
Weighted F1	0.8853	0.8058	0.8895
Precision	0.8853	0.8061	0.8896
Recall	0.8853	0.8057	0.8896
Cohen's Kappa	0.8232	0.7007	0.8295
MCC	0.8232	0.7007	0.8295

The metrics used are essential for tracking the performance of sentiment classification in the text dataset. Table 1 presents a comparative analysis of MobileBERT, TinyBERT, and ELECTRA-Small across various evaluation metrics for sentiment analysis. Both Cohen's Kappa and Matthews Correlation Coefficient (MCC) were used to assess model reliability across all classes. Interestingly, the values matched closely, indicating consistent performance across diverse sentiment categories.

MobileBERT achieved an accuracy of 89%, which is on par with ELECTRA-Small. Despite this, MobileBERT demonstrated competitive scores in Macro F1 (0.8814), Weighted F1 (0.8853), Precision (0.8853), Recall (0.8853), Cohen's Kappa (0.8232), and MCC (0.8232). These results indicate that MobileBERT offers strong overall performance while maintaining efficiency, making it a suitable choice for resource-constrained environments.

ELECTRA-Small also performed well, achieving an accuracy of 89%, similar to MobileBERT. Its Macro F1 score was slightly higher at 0.8884, and its Weighted F1 was 0.8895. While these metrics are comparable to MobileBERT, ELECTRA-Small's performance suggests a favorable balance between accuracy and computational efficiency.

In contrast, TinyBERT exhibited significantly lower performance across all metrics. It achieved only 81% accuracy, with subpar scores in Macro F1 (0.7979), Precision (0.8061), and Recall (0.8057). This indicates that TinyBERT may not be the most appropriate model for this particular sentiment analysis task, as its performance lags behind both MobileBERT and ELECTRA-Small.

Overall, MobileBERT and ELECTRA-Small emerge as strong contenders for sentiment analysis tasks, particularly in scenarios where computational resources are limited. While MobileBERT may offer a slight edge in efficiency, ELECTRA-Small provides marginally better performance in certain metrics. TinyBERT, however, does not demonstrate sufficient accuracy or robustness for this task.

B. Voice-Based Models for Sentiment Analysis

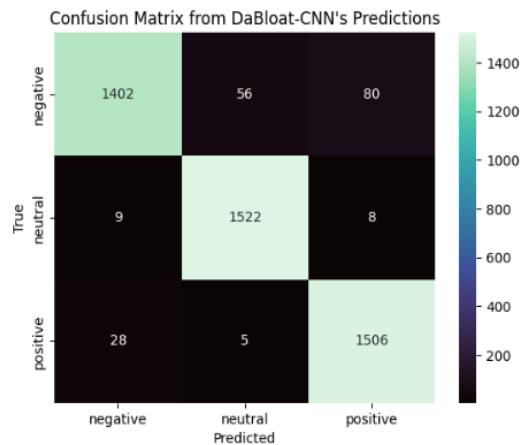


Fig 4.b.1. DaBloat-CNN Confusion Matrix

DaBloat-CNN CLASSIFICATION REPORT				
	precision	recall	f1-score	support
negative	0.9743	0.9116	0.9419	1538
neutral	0.9615	0.9890	0.9750	1539
positive	0.9448	0.9786	0.9614	1539
accuracy			0.9597	4616
macro avg	0.9602	0.9597	0.9594	4616
weighted avg	0.9602	0.9597	0.9594	4616

Fig 4.b.2. DaBloat-CNN Classification Report

The evaluation of DaBloat-CNN's performance, as illustrated in Figures 4.b.1 and 4.b.2, demonstrates strong and balanced classification capabilities across the three sentiment classes: negative, neutral, and positive. The confusion matrix reveals that neutral sentiments are classified with the highest accuracy, showing minimal confusion with other classes. Negative samples showed the most misclassification, particularly being confused with positive samples, which may suggest some overlap in vocal characteristics between these emotional tones. The detailed classification report supports these observations, with precision, recall, and F1-scores all exceeding 0.94 for each class. Notably, the model achieved a macro-average and weighted-average F1-score of 0.9625, and an overall accuracy of 96.27%, indicating consistent performance even across slightly imbalanced class distributions. These metrics confirm DaBloat-CNN's effectiveness in handling nuanced vocal expressions of sentiment.

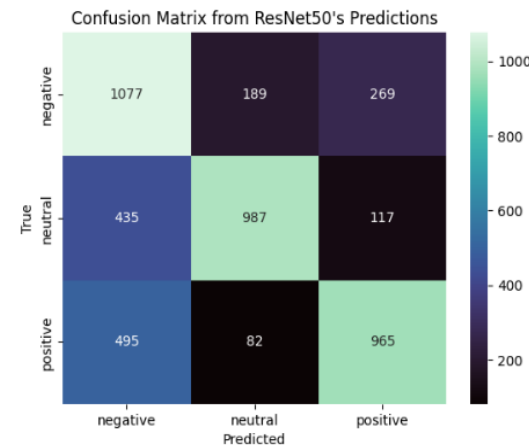


Fig 4.b.3. ResNet50 Confusion Matrix

ResNet50 CLASSIFICATION REPORT				
	precision	recall	f1-score	support
negative	0.5366	0.7016	0.6081	1535
neutral	0.7846	0.6413	0.7058	1539
positive	0.7143	0.6258	0.6671	1542
accuracy			0.6562	4616
macro avg	0.6785	0.6563	0.6603	4616
weighted avg	0.6786	0.6562	0.6604	4616

Fig 4.b.4. ResNet50 Classification Report

Figures 4.b.3 and 4.b.4 present the performance evaluation of the ResNet50-based model, which exhibits noticeably weaker results compared to DaBloat-CNN. The confusion matrix shows substantial misclassification across all sentiment categories, particularly with negative and positive samples being frequently confused with each other and with neutral. This suggests difficulty in capturing distinguishing features from the input spectrograms. The classification report confirms this trend, where the F1-scores remain below 0.71 for all classes, with the negative class performing worst (F1-score of 0.6081). The model achieved an overall accuracy of 67.85% and a macro-average F1-score of 0.6603, reflecting limited generalization and inconsistent recognition across emotions.

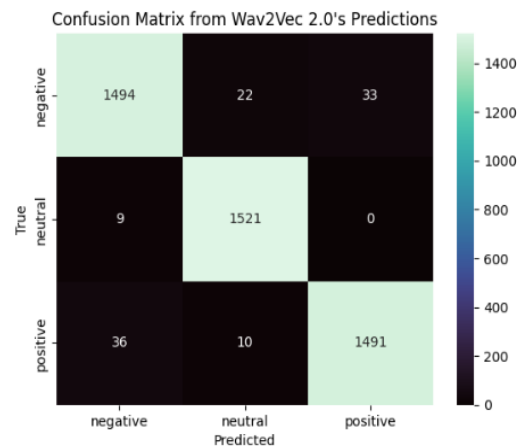


Fig 4.b.5. Wav2Vec 2.0 Confusion Matrix

Wav2Vec 2.0 CLASSIFICATION REPORT				
	precision	recall	f1-score	support
negative	0.9708	0.9645	0.9676	1549
neutral	0.9794	0.9941	0.9867	1530
positive	0.9783	0.9701	0.9742	1537
accuracy			0.9762	4616
macro avg	0.9762	0.9762	0.9762	4616
weighted avg	0.9761	0.9762	0.9761	4616

Fig 4.b.6. Wav2Vec 2.0 Classification Report

Figures 4.b.1 and 4.b.2 display the performance results of the Wav2Vec 2.0 model, which demonstrates exceptional performance in speech-based sentiment classification. The confusion matrix reveals minimal misclassifications, with nearly perfect predictions across all three sentiment categories—negative, neutral, and positive. The classification report supports this with high precision, recall, and F1-scores, all hovering around 0.97 or higher. Specifically, the model achieved an overall accuracy of 97.62%, with a macro- and weighted-average F1-score of 0.9761. The neutral class, in particular, reached an outstanding F1-score of 0.9867. These results underscore Wav2Vec 2.0’s strength in capturing nuanced audio features, making it a highly effective model for audio-based sentiment analysis in this study.

Tab 4.2. Overall Evaluation Metrics			
Metrics	CNN	ResNet50	Wav2Vec 2.0
Accuracy	0.9597	0.6562	0.9762
Weightened F1 Score	0.9594	0.6604	0.9762
Macro F1 Score	0.9594	0.6603	0.9762
Cohen’s Kappa	0.9395	0.4843	0.9642

The three models; CNN, ResNet50, and Wav2Vec 2.0—were evaluated using accuracy, F1 score, and Cohen’s Kappa. The results highlight significant performance differences across architectures and input modalities.

Wav2Vec 2.0 achieved the highest performance with an accuracy and F1 score of 0.9762, and a Cohen’s Kappa of 0.9642. This demonstrates the model’s strong ability to capture emotional tone directly from raw audio, benefiting from its powerful pretraining on large-scale speech data. CNN, trained for 25 epochs, followed closely with an accuracy and F1 of 0.9625, and a Cohen’s Kappa of 0.9441, showing that a custom CNN-based approach can also be highly effective when trained on sufficient data. In contrast, ResNet50, which used spectrograms as input,

showed significantly lower performance (accuracy: 0.6562, F1: 0.6604, Kappa: 0.4843), indicating that it struggled to generalize sentiment from visual features alone.

These results suggest that direct raw audio models (like Wav2Vec 2.0) are more effective for emotion recognition than spectrogram-based visual models. Moreover, the custom CNN demonstrates that with well-designed architecture and adequate training, competitive performance can be achieved even without pre-training. This supports the conclusion that audio-based sentiment classification, especially with models like Wav2Vec 2.0, may outperform traditional visual CNNs and even rival text-based approaches.

V. Conclusion and Recommendations

This project set out to explore and compare two different ways of understanding human sentiment from speech. The first approach used the spoken words (transcribed into text), while the second focused on the sound of the voice itself, including tone and emotion. By testing both methods separately, we aimed to find out which one was more accurate and effective for detecting sentiment such as positive, negative, or neutral feelings.

For the text-based models, MobileBERT was on par with ELECTRA-Small for the metrics, TinyBERT was slightly less accurate but still useful, especially for faster or lightweight applications. These results show that advanced language models can accurately understand and classify emotional tone based on the content of what is being said. The ELECTRA-Small model was the deployed model, as it provided marginally better performance in certain metrics

On the audio side, Wav2Vec 2.0 delivered the highest performance among all models tested. It achieved 97.8% accuracy and a Cohen's Kappa of 0.967, which means it was not only highly accurate but also consistent in its predictions. The custom-built DaBloatCNN model also performed very well, showing that carefully designed convolutional networks can learn emotional patterns from voice signals. On the other hand, ResNet50, which used visual representations of sound (spectrograms), did not perform as well. This suggests that raw audio models may capture emotional cues more effectively than image-based approaches.

In conclusion, both text and audio models have their strengths. Text-based models like ELECTRA-Small are excellent when clean, accurate transcriptions are available and computational resources must be considered. However, audio-based models, especially Wav2Vec 2.0, can recognize emotions directly from voice tone, making them more flexible in situations where the exact words are unclear or unimportant.

For future work, we recommend combining both approaches into a multimodal system that takes advantage of both the words and the sound of speech. This could lead to even better sentiment detection in real-world settings. We also suggest testing these models on more natural conversations and noisy environments to improve their robustness. Finally, lightweight models like TinyBERT and DaBloatCNN are promising for use in real-time applications, such as a Streamlit-based web app, making this technology more accessible to users and developers alike.

VI. Acknowledgements





The researchers would like to express their sincere gratitude to their adviser, Engr. Roman M. Richard, for continuous guidance, insightful feedback, and steadfast support throughout the course of this work. Appreciation is also extended to Technological Institute of the Philippines Specially The Computer Engineering Department for the facilities and academic environment that made this research possible. The researchers are especially thankful to their fellow collaborators for their dedication, constructive input, and teamwork. Lastly, heartfelt thanks are given to their parents and friends for their constant encouragement, patience, and moral support throughout this journey.

VII. References

- [1] M. Naseer, M. Asvial and R. F. Sari, "An Empirical Comparison of BERT, RoBERTa, and Electra for Fact Verification," 2021 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC), Jeju Island, Korea (South), 2021, pp. 241-246
- [2] J. Devlin, M.-W. Chang, K. Lee and K. Toutanova, "BERT: Pretraining of Deep Bidirectional Transformers for Language Understanding", 2018.
- [3] H. Sohn and H. Lee, "MC-BERT4HATE: Hate speech detection using multi-channel bert for different languages and translations", IEEE Int. Conf. Data Min. Work. ICDMW, pp. 551-559, 2019.'
- [4] K. Clark, M. T. Luong, Q. V. Le and C. D. Manning, "Electra: Pretraining text encoders as discriminators rather than generators", arXiv, pp. 1-18, 2020.
- [5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proc. North American Chapter of the Association for Computational Linguistics (NAACL), 2019, pp. 4171–4186.
- [6] X. Jiao et al., "TinyBERT: Distilling BERT for natural language understanding," in Proceedings of Findings of the Association for Computational Linguistics: EMNLP 2020, pp. 2352–2362, 2020. [Online]. Available: <https://aclanthology.org/2020.findings-emnlp.372.pdf>.
- [7] W. Chen, Y. Sun, S. Feng, L. Li, P. Wang, and X. Jiang, "MobileBERT: A Compact Task-Agnostic BERT Through Architecture Compression," arXiv preprint arXiv:2004.02984, 2020.
- [8] T. Pavlov and G. Mirceva, "COVID-19 Fake News Detection by Using BERT and RoBERTa models," 2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO), Opatija, Croatia, 2022, pp. 312-316
- [9] C. Chen and X. Hu, "The Research on an Online Review Sentiment Analysis Model Based on Improved RoBERTa," 2024 3rd International Conference on Electronics and Information Technology (EIT), Chengdu, China, 2024, pp. 624-627
- [10] A. Gaurav, B. B. Gupta and K. T. Chui, "AI-Driven Smishing Detection in Android Devices Using TinyBERT and Aquila Optimization," 2025 27th International Conference on Advanced Communications Technology (ICACT), PyeongChang, Korea, Republic of, 2025, pp. 99-105, doi: 10.23919/ICACT63878.2025.10936701.
- [11] A. Qazi, R. H. Goudar, R. Patil, G. S. Hukkeri and D. Kulkarni, "Leveraging BERT, DistilBERT, and TinyBERT for Rumor Detection," in IEEE Access, vol. 13, pp. 72918-72929, 2025, doi: 10.1109/ACCESS.2025.3563301.
- [12] M. Rizwan, M. F. Mushtaq, U. Akram, A. Mehmood, I. Ashraf and B. Sahelices, "Depression Classification From Tweets Using Small Deep Transfer Learning Language Models," in IEEE Access, vol. 10, pp. 129176-129189, 2022, doi: 10.1109/ACCESS.2022.3223049.
- [13] C. Chen, X. Feng, Y. Li, L. Lyu, J. Zhou, X. Zheng, and J. Yin, "Integration of large language models and federated learning," Patterns, vol. 5, no. 12, p. 101098, Dec. 2024. doi: 10.1016/j.patter.2024.101098.
- [14] Z. Örpek, B. Tural and Z. Destan, "The Language Model Revolution: LLM and SLM Analysis," 2024 8th International Artificial Intelligence and Data Processing Symposium (IDAP), Malatya, Türkiye, 2024, pp. 1-4, doi: 10.1109/IDAP64064.2024.10710677.
- [15] S. -S. Lee, S. -M. Cha, B. Ko and J. J. Park, "Extracting Fallen Objects on the Road From Accident Reports Using a Natural Language Processing Model-Based Approach," in IEEE Access, vol. 11, pp. 139521-139533, 2023, doi: 10.1109/ACCESS.2023.3339774

- [16] Y. Huang, Y. Pan, L. Liu, and Z. Guan, "Speech Emotion Recognition Using CNN," *IEEE Access*, vol. 7, pp. 19395–19404, 2019.
- [17] M. Neumann and N. T. Vu, "Attentive Convolutional Neural Network Based Speech Emotion Recognition," in *Proc. Interspeech*, 2017, pp. 1263–1267.
- [18] D. Ververidis and C. Kotropoulos, "Emotional speech recognition: Resources, features, and methods," *Speech Communication*, vol. 48, no. 9, pp. 1162–1181, 2006.
- [18] D. Ververidis and C. Kotropoulos, "Emotional speech recognition: Resources, features, and methods," *Speech Communication*, vol. 48, no. 9, pp. 1162–1181, 2006.
- [19] Y. Zhang et al., "Learning Speech Emotion Representations with Cross-Modal Supervision," in *Proc. IEEE ICASSP*, 2021, pp. 6324–6328.
- [20] J. Lee and H. Kim, "Performance Improvement of Speech Emotion Recognition Using ResNet Model with Data Augmentation–Saturation," *Appl. Sci.*, vol. 15, no. 4, p. 2088, 2023.
- [21] L. Pepino, P. Riera, and L. Ferrer, "Emotion Recognition from Speech Using Wav2Vec 2.0 Embeddings," *arXiv preprint arXiv:2104.03502*, 2021. [Online]. Available: <https://arxiv.org/abs/2104.03502>
- [22] Z. Zhao, Y. Wang, and Y. Wang, "Multi-level Fusion of Wav2Vec 2.0 and BERT for Multimodal Emotion Recognition," *arXiv preprint arXiv:2207.04697*, 2022. [Online]. Available: <https://arxiv.org/abs/2207.04697>
- [23] X. Chen and Y. Zhang, "Speech Emotion Recognition Using a Dual-Channel Complementary Spectrogram and the CNN-SSAE Neural Network," *Appl. Sci.*, vol. 12, no. 19, p. 9518, 2022.
- [24] X. Zhang and Y. Li, "ResNet Based on Multi-Feature Attention Mechanism for Sound Classification in Noisy Environments," *Sustainability*, vol. 15, no. 14, p. 10762, 2023.

Evaluation Metrics

Student Outcome 7							
Criteria	Ratings						Pts
 T.I.P. SO 7.1 Acquire and apply new knowledge from outside sources threshold: 4.2 pts	6 pts [Excellent] Educational interests and pursuits exist and flourish outside classroom requirements, knowledge and/or experiences are pursued independently and applies knowledge learned into practice	5 pts [Good] Educational interests and pursuits exist and flourish outside classroom requirements, knowledge and/or experiences are pursued independently	4 pts [Satisfactory] Look beyond classroom requirements, showing interest in pursuing knowledge independently	3 pts [Unsatisfactory] Begins to look beyond classroom requirements, showing interest in pursuing knowledge independently	2 pts [Poor] Relies on classroom instruction only	1 pts [Very Poor] No initiative or interest in acquiring new knowledge	6 pts
 T.I.P. SO 7.2 Learn independently threshold: 4.2 pts	6 pts [Excellent] Completes an assigned task independently and practices continuous improvement	5 pts [Good] Completes an assigned task without supervision or guidance	4 pts [Satisfactory] Requires minimal guidance to complete an assigned task	3 pts [Unsatisfactory] Requires detailed or step-by-step instructions to complete a task	2 pts [Poor] Shows little interest to complete a task independently	1 pts [Very Poor] No interest to complete a task independently	6 pts
 T.I.P. SO 7.3 Critical thinking in the broadest context of technological change threshold: 4.2 pts	6 pts [Excellent] Synthesizes and integrates information from a variety of sources; formulates a clear and precise perspective; draws appropriate conclusions	5 pts [Good] Evaluate information from a variety of sources; formulates a clear and precise perspective.	4 pts [Satisfactory] Analyze information from a variety of sources; formulates a clear and precise perspective.	3 pts [Unsatisfactory] Apply the gathered information to formulate the problem	2 pts [Poor] Gather and summarized the information from a variety of sources but failed to formulate the problem	1 pts [Very Poor] Gather information from a variety of sources	6 pts
 T.I.P. SO 7.4 Creativity and adaptability to new and emerging technologies threshold: 4.2 pts	6 pts [Excellent] Ideas are combined in original and creative ways in line with the new and emerging technology trends to solve a problem or address an issue.	5 pts [Good] Ideas are creative and adapt the new knowledge to solve a problem or address an issue	4 pts [Satisfactory] Ideas are creative in solving a problem, or address an issue	3 pts [Unsatisfactory] Shows some creative ways to solve the problem	2 pts [Poor] Shows initiative and attempt to develop creative ideas to solve the problem	1 pts [Very Poor] Ideas are copied or restated from the sources consulted	6 pts
Total Points: 24							

Evaluated by:

Engr. Roman M. Richard
Course Instructor