**Multimodal Sentiment Analysis Using Images and Audio**

**I. Introduction**

**A. Background**

Sentiment analysis seeks to identify and extract subjective information from various forms of communication. In the realm of **multimodal data**, where visual cues, vocal intonations, and linguistic content interact, deciphering sentiment becomes a complex but critical challenge.

The **CMU-MOSI (Multimodal Opinion Sentiment and Intensity)** dataset provides a rich source of information, containing continuous sentiment scores derived from short monologue videos along with synchronized **acoustic**, **visual**, and **textual** features.

**B. Motivation**

The motivation behind this project is twofold:

1. Understanding human sentiment in multimodal communication is crucial for applications ranging from human–computer interaction to social media analytics.
2. The inherent challenges—such as variable-length feature representations and class imbalances when discretizing continuous sentiment—necessitate advanced techniques for data preprocessing and analysis.

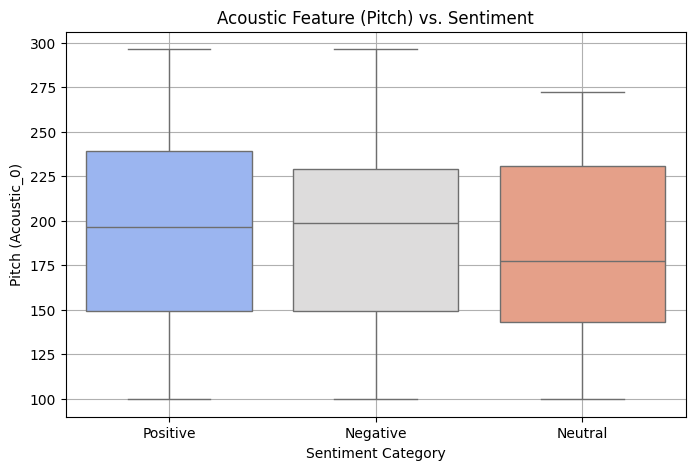
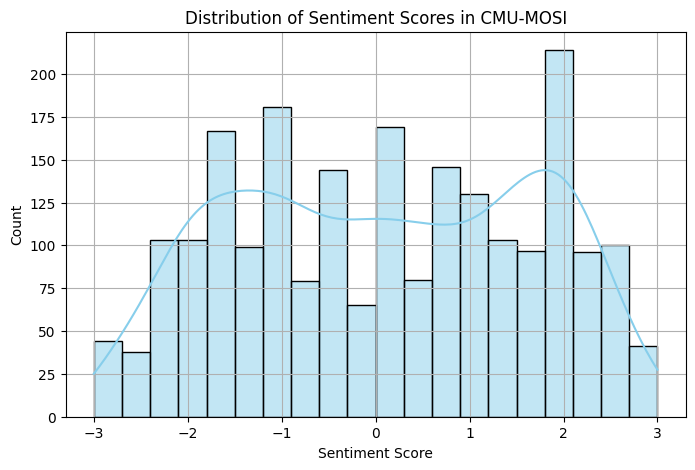
By extracting, aligning, and visualizing these features, we aim to develop robust models that capture the nuances of sentiment across different modalities.

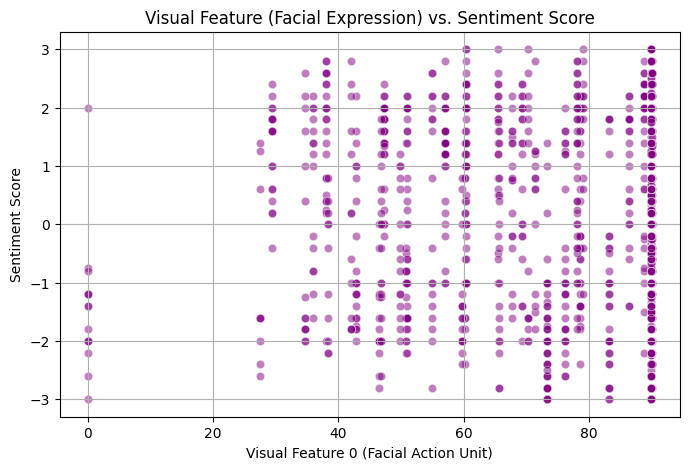
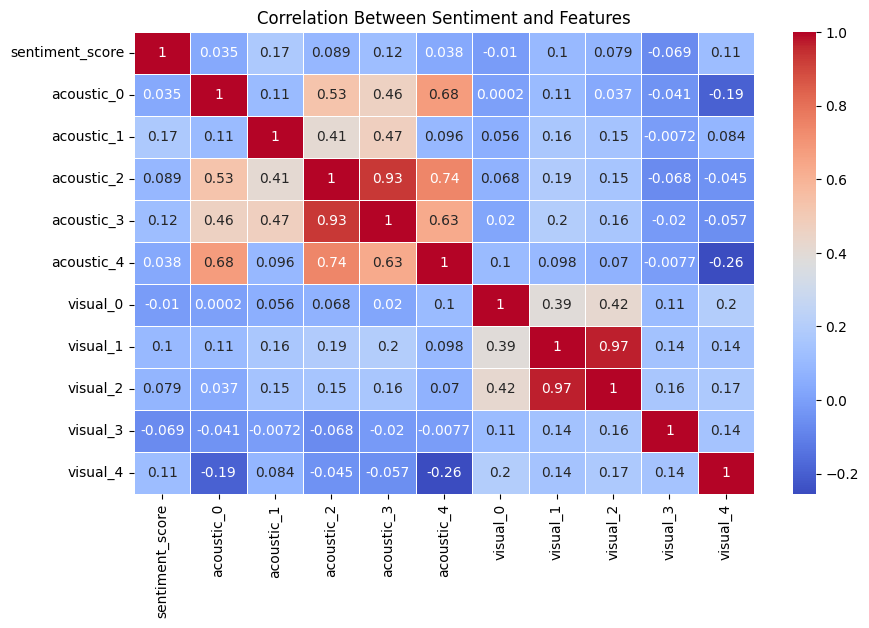
**II. Dataset Description**

The dataset utilized for this project is the **CMU-MOSI dataset**, which contains:

* **Opinion Labels**: Continuous sentiment scores ranging from -3 (very negative) to +3 (very positive).
* **Acoustic Features**: Extracted via **COVAREP**, capturing vocal attributes such as pitch, MFCCs, and energy.
* **Visual Features**: Extracted from facial expressions using **OpenFace**, such as Action Units and gaze.
* **Textual Features**: Timestamped transcripts embedded using pretrained vectors like **GloVe**.

All modalities are aligned segment-wise. Features were extracted from raw videos and compiled into a single CSV format, with each row corresponding to a segment annotated by video\_id, start\_time, end\_time, continuous sentiment\_score, and its modality features.





**III. Problem Statement**

The objective of this project is to **classify sentiment** from multimodal inputs. Key challenges include:

* **Feature Extraction and Alignment**: Aggregating variable-length modality features into fixed-length representations.
* **Score Discretization**: Mapping continuous sentiment scores into discrete sentiment classes (Negative, Neutral, Positive).
* **Data Visualization**: Understanding feature distributions and inter-modality relations.
* **Class Imbalance**: Using oversampling and undersampling techniques like **SMOTE** for balanced classification.

**IV. Understanding and Preprocessing Data**

**A. Text Modality**

* Raw transcripts from video segments
* Preprocessing: Tokenization, lowercasing
* Embedding: GloVe embeddings → 300-dimensional

**B. Audio Modality**

* Speech signals with emotional tone and intensity
* Feature Extraction: MFCCs, pitch, energy via **COVAREP**
* Output: 74-dimensional vector

**C. Visual Modality**

* Facial cues, Action Units, gaze
* Feature Extraction: OpenFace
* Output: 430-dimensional vector

# Final Feature Format

Each row in the dataset corresponds to one utterance (video segment) and includes:

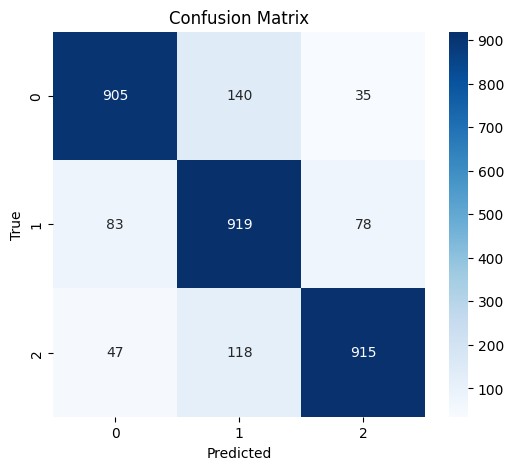
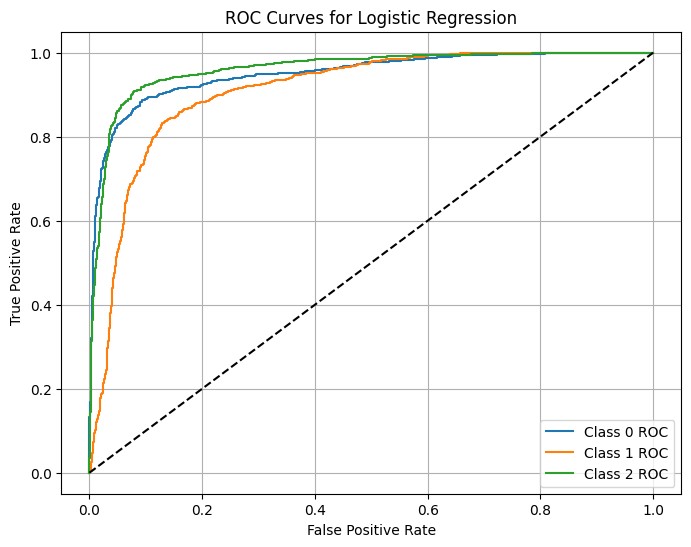
|  |  |  |
| --- | --- | --- |
| Modality | Features | Description |
| **SegmentId** | 3100 | To combine all features |
| **Audio** | 74 | MFCCs, pitch, energy, delta features |
| **Text** | 300 | Averaged embedding vector |
| **Visual** | 430 | Action Units, landmarks, head pose, gaze |
| **Total** | **804** | Concatenated multimodal feature vector |
| **Label** | 1 | Sentiment (positive = 1, negative = -1,neutral=0) |

**V. Baseline Models**

**A. Logistic Regression**

Logistic Regression is extended to a multinomial logistic regression (softmax classifier) for handling multiple classes. It is a linear model that maps the input features to class probabilities using the softmax function.

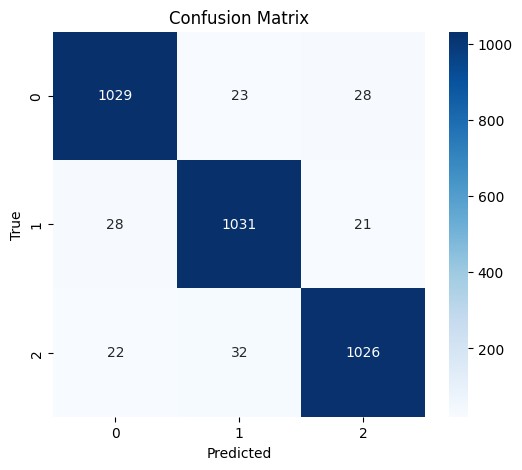
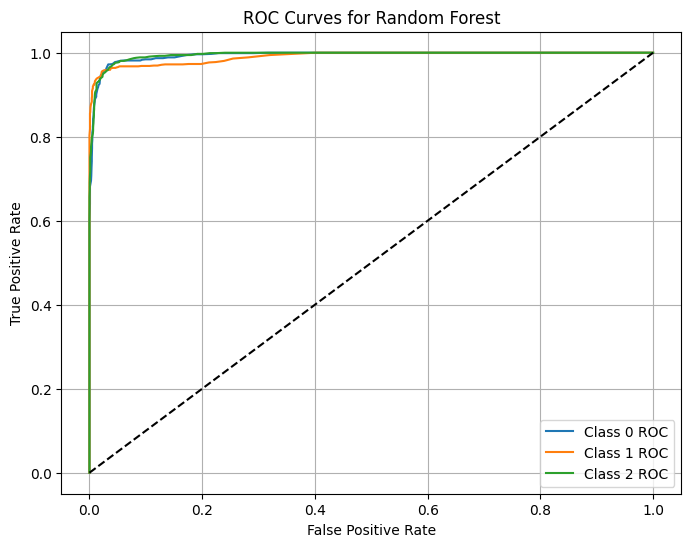
* Input: 804-dimensional feature vector (74 acoustic + 300 text + 430 visual)
* Output: One of three sentiment classes
* Advantage: Fast and interpretable baseline
* Limitation: Cannot capture complex patterns or non-linear interactions



**B. Random Forest**

Random Forest is a tree-based ensemble classifier. It handles non-linear feature interactions and can naturally extend to multiclass classification by voting across decision trees.

* Input: 804-dimensional multimodal feature vector
* Output: Multiclass prediction (positive, neutral, or negative)
* Advantage: Robust to noise and overfitting
* Limitation: Slower training on large datasets; harder to interpret

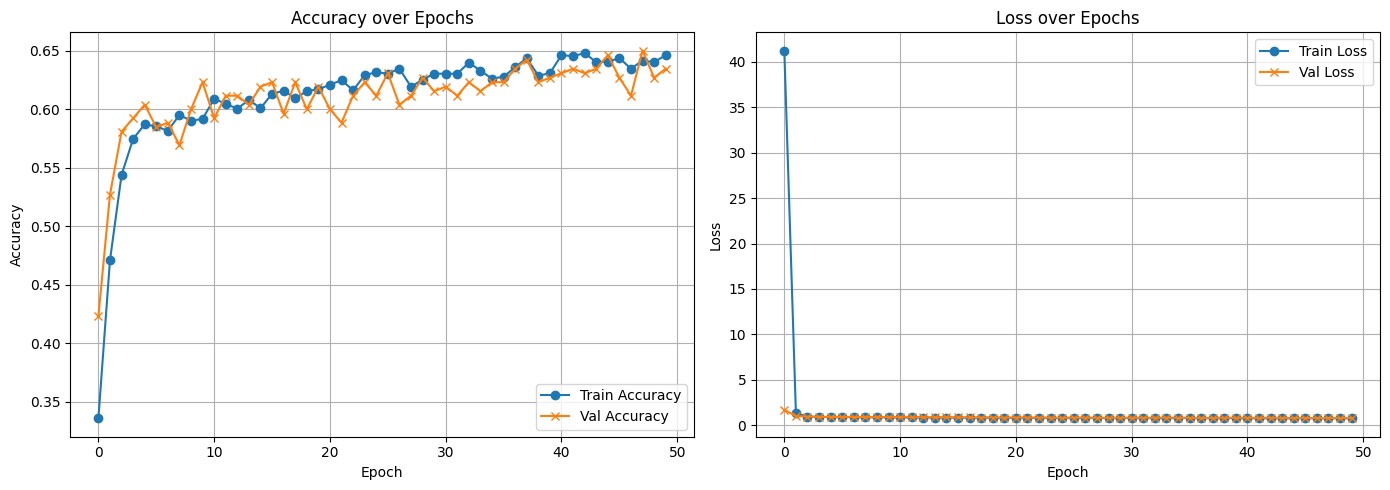


**C. CNN**

A custom 1D CNN architecture is designed to process each modality separately with convolutional layers before merging the features for final classification.

Architecture Summary:

* Separate 1D Conv layers for each modality
* Concatenation of modality outputs
* Fully connected layers with dropout
* Softmax activation for 3-class prediction
* Input: (74-dim acoustic, 300-dim text, 430-dim visual)
* Output: 3-class softmax
* Advantage: Learns cross-modal, non-linear, and temporal patterns
* Limitation: More computationally demanding and data-hungry



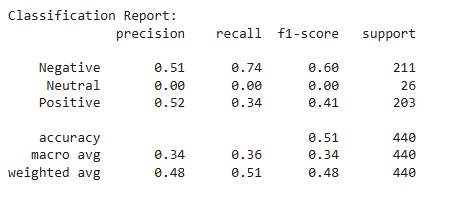
**VI. Advanced Deep Learning Models**

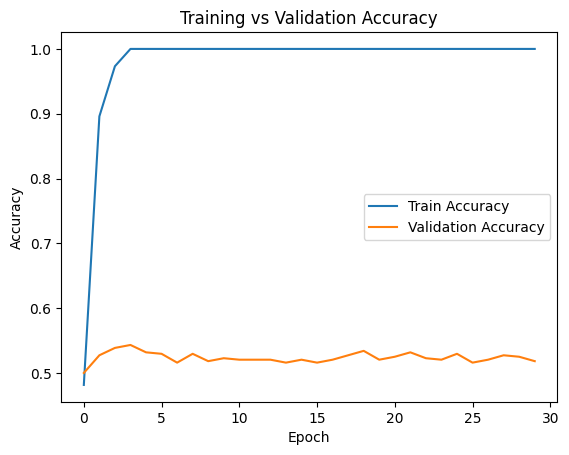
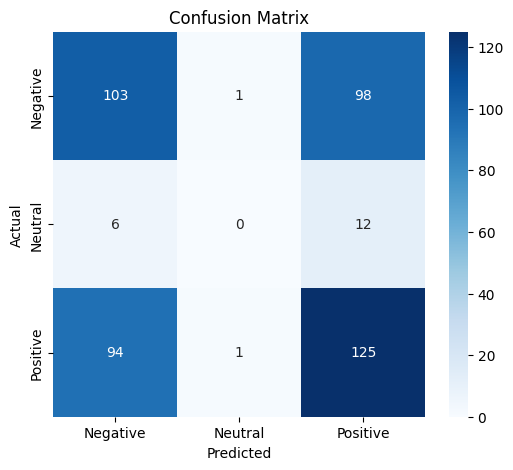
To improve over classical machine learning models, we explored multiple deep learning architectures capable of capturing intra- and inter-modality dependencies. These models are designed to work with high-dimensional inputs and exploit the temporal or hierarchical nature of multimodal features. Below are the key architectures evaluated:

**A. Early Fusion with Multihead Attention**

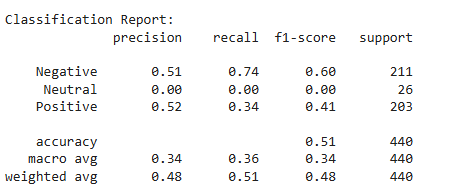
This model projects features from each modality (text, audio, visual) into a common embedding space, applies attention across them, and uses the aggregated representation for classification.

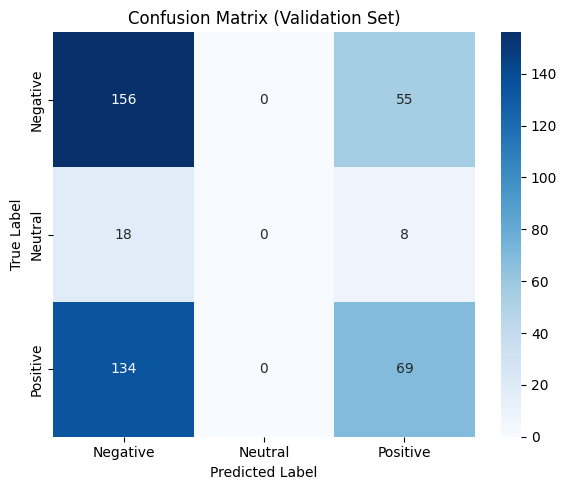
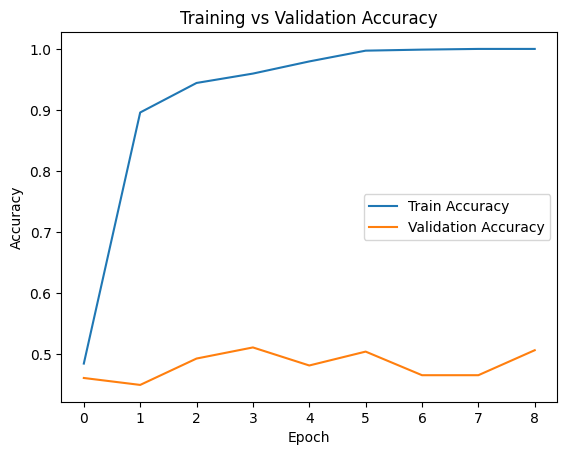
* Modality-specific input branches map features to 64 dimensions.
* The attention mechanism enables the model to focus on the most informative modality for a given input.
* Each modality → FC layers → 64-d vector
* Features fused using **MultiheadAttention**
* Attention output pooled and classified
* Used dropout (0.5), AdamW, scheduler, and early stopping



The model was overfitting as training accuracy was 1 and testing accuracy was up to 0.53. So we made changes such as early stopping, higher dropout, smaller FC layers, lower learning rate and increased weight decay. However, any changes in accuracy were not observed.

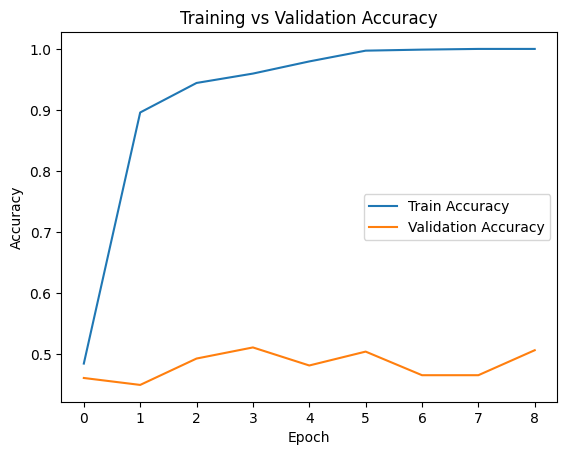
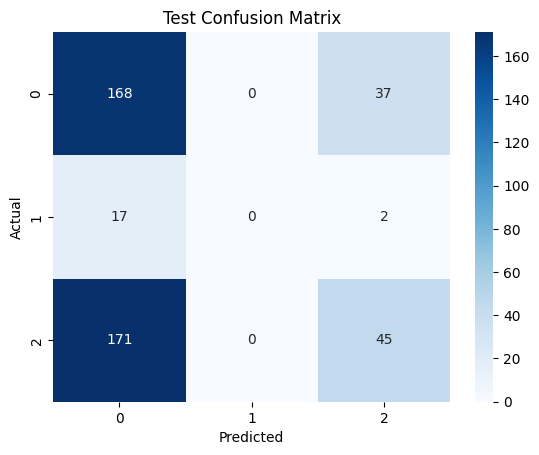




**B. Lightweight Late Fusion (Low Memory)**

This architecture is optimized for low-memory environments and uses simple feed-forward networks.

* Each modality is processed independently through a small FC network.
* Outputs are concatenated and passed to a shallow classifier.
* Avoids use of CNNs or recurrent layers, making it ideal for deployment on resource-constrained devices.
* Despite simplicity, performs competitively due to effective feature engineering.
* Each branch: FC(Input → 128)
* Fusion: Concatenation → FC(384 → 256 → 3)
* Extremely efficient, no CNN/LSTM, suitable for low-resource setups

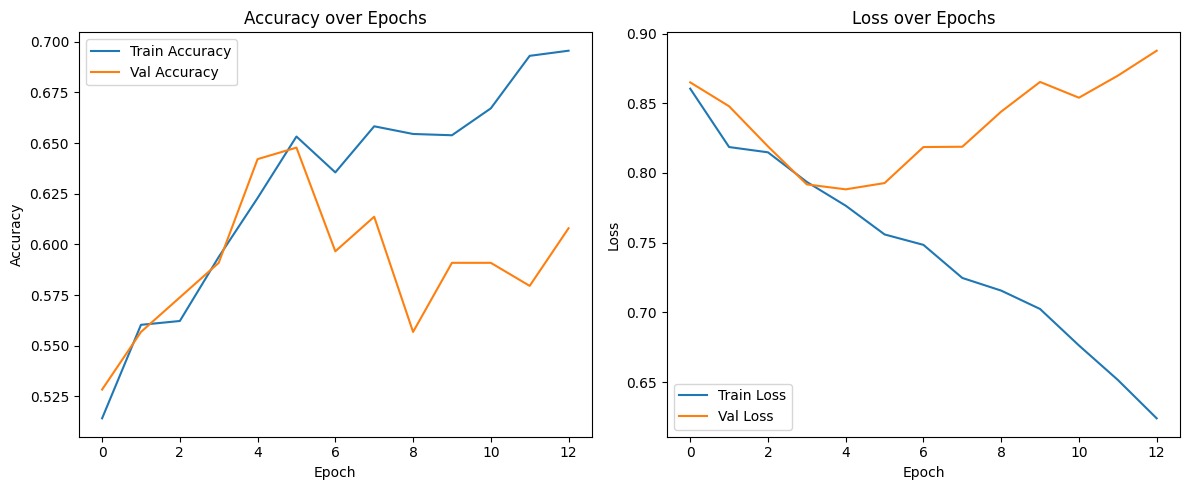
 

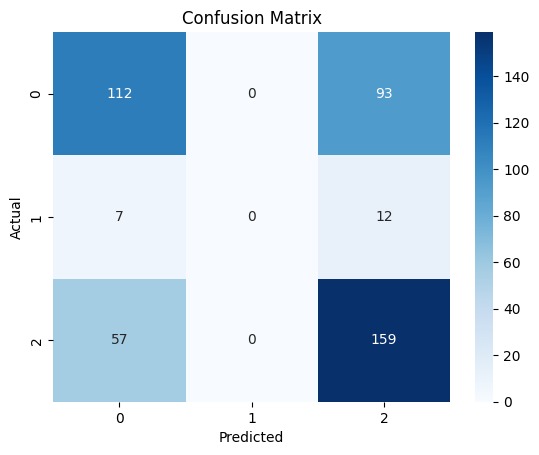
In this case the model was again being overfitted and was giving biased output.

**C. LSTM-Based Late Fusion**

Designed to capture temporal dependencies within the text modality using BiLSTM, while applying standard FC layers to audio and visual inputs.

* BiLSTM captures sequential semantic dependencies in spoken content.
* Outputs from all three branches are fused and passed through dense layers.
* This model balances complexity and performance, offering superior generalization compared to simpler architectures.
* Performs best in terms of both accuracy and F1 score, albeit with slightly higher computational cost.
* **Text**: BiLSTM(300 → 128×2) → FC(256 → 128)
* **Audio & Visual**: FC → 128
* Fusion: Concatenation → FC(384 → 128 → 3)
* Strong generalization but slightly more expensive to train



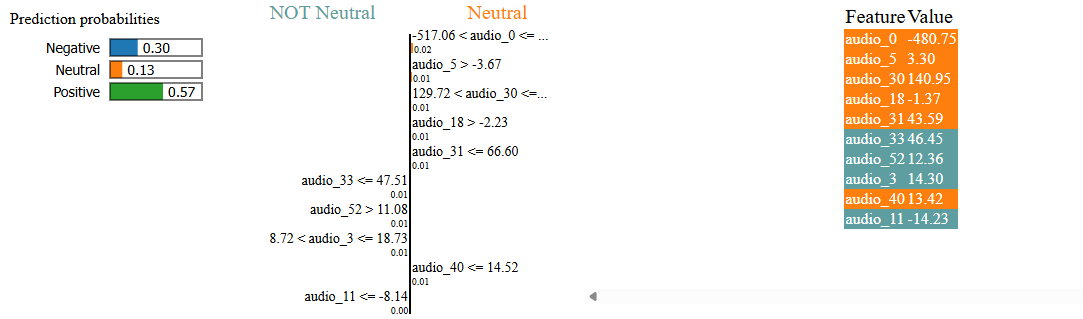


**VII. Evaluation and Testing**

|  |  |  |
| --- | --- | --- |
| **Model** | **Accuracy** | **Notes** |
| Logistic Regression | 0.86 | Fast, interpretable |
| Random Forest | 0.96 | Handles non-linearity well |
| CNN (1D) | 0.65 | Captures sequence patterns |
| Early Fusion | 0.53 | Best among early fusion |
| Lightweight Fusion | 0.48 | Ultra-low resource usage |
| LSTM Fusion | 0.70 | Best overall DL model |

**VIII. Explainability**

Used **LIME** on late fusion model for tabular-like explanation.

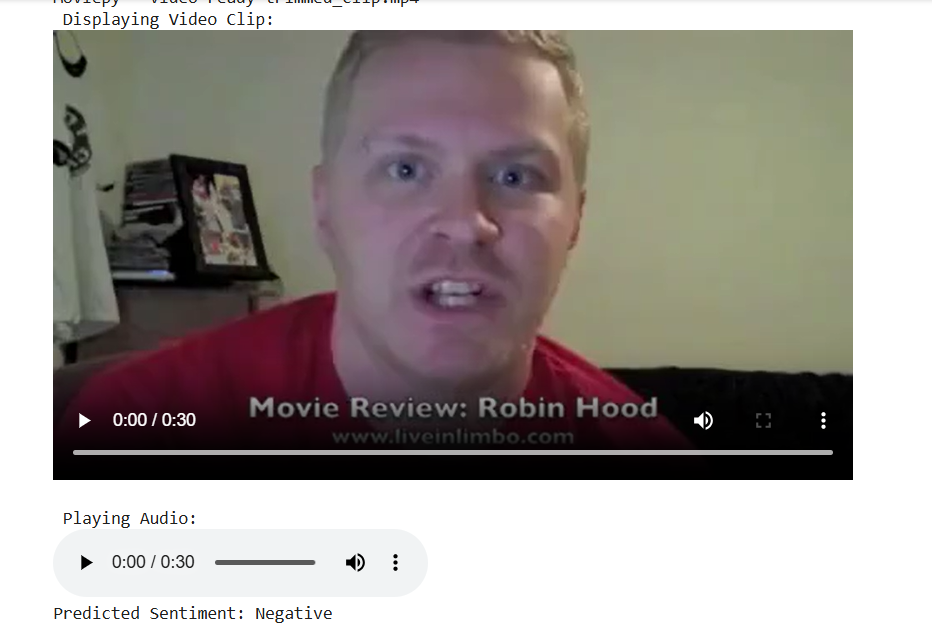


**IX. Realtime processing**

In real-world applications such as emotion-aware chatbots, surveillance analytics, and assistive technologies, **real-time sentiment prediction** is critical. Our models are designed to support real-time inference by using fixed-length feature vectors and avoiding computationally expensive components (e.g., full BERT or video decoding pipelines).

Key strategies for real-time deployment include:

* **Lightweight Model Design**: The late fusion model avoids deep CNNs and uses efficient FC layers.
* **Pre-extracted Features**: Using extracted features (e.g., GloVe for text, OpenFace for visuals) reduces online processing cost.
* **Batching and Quantization**: Mini-batching and model quantization can further accelerate inference on edge devices.
* **TorchScript Export**: All models are compatible with PyTorch's JIT or ONNX export for deployment on mobile or embedded platforms.

****

**X. Conclusion**

This milestone demonstrates how combining **text**, **audio**, and **visual** signals improves sentiment classification. We developed and compared various models to balance performance and computational cost. The **LSTM-based late fusion** model provided the best results, while the **lightweight late fusion** model offered a scalable alternative.