iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** VI **Month of publication:** June 2026

DOI: https://doi.org/10.22214/ijraset.2026.83557

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VI June 2026- Available at www.ijraset.com

Sentiment Analysis and Speaker Mapping with Machine Learning

Mayur Ankushrao[1], Sanket Pawar[2], Vishal Mule[3], Aniket Markad[4], Prof. S. V. Shinde[5]
Dept. of Computer Engineering, PDEA's College of Engineering, Savitribai Phule Pune University, Pune, Maharashtra, India

Abstract: ***In multi-user automated ecosystems, extracting actionable intelligence from spoken conversations requires understanding both who is speaking and the emotional context of their words. This paper presents an integrated, decoupled machine learning architecture designed for real-time speech-to-text transcription, unsupervised speaker diarization, and linguistic sentiment classification. The framework utilizes an optimized faster-whisper transformer pipeline to transcribe audio signals into granular, timestamped text segments. Concurrently, the acoustic domain leverages Mel-Frequency Cepstral Coefficients (MFCCs) processed through a multi-seed Consensus KMeans clustering ensemble to achieve stable speaker identity tracking without requiring prior acoustic enrollment. The extracted textual segments are subsequently converted using a Term Frequency-Inverse Document Frequency (TF-IDF) vectorizer and classified using a calibrated machine learning model. Experimental evaluations demonstrate high computational efficiency on commodity CPU infrastructure, making it highly suitable for enterprise customer intelligence and digital media analytics.***

Index Terms: Speech-to-Text, Speaker Diarization, Sentiment Analysis, Consensus Clustering, MFCC, Natural Language Processing.

I. INTRODUCTION

The explosion of multi-speaker multimedia content—ranging from corporate meetings and customer service recordings to educational lectures and radio broadcasts—has accelerated the demand for automated audio processing pipelines. Extracting insights from these streams introduces a two-fold computational challenge: Speaker Diarization (solving the problem of "who spoke when") and Sentiment Analysis (identifying the emotional orientation of the communication).

Traditional approaches frequently attempt end-to-end multi-modal deep learning, optimizing acoustic waveforms and language features simultaneously. While expressive, these models demand high computational resources, are sensitive to environmental noise variations, and often require expensive GPU acceleration.

To overcome these constraints, this study presents a robust, sequential cascade framework that decouples acoustic mapping from emotional classification. By transforming the raw voice signal into a discrete sequence of textual tokens before running sentiment estimation, the system leverages highly mature Natural Language Processing (NLP) text classifiers while preserving speaker identity via localized acoustic feature clustering.

The primary contributions of this work include:

1) A multi-threaded deployment model designed to run efficiently on standard CPU architectures without risk of precision errors or segmentation faults.
2) A voting ensemble framework for KMeans clustering that mitigates the problem of label inversion across separate data processing runs.
3) A systematic evaluation methodology that uses strict mathematical boundary zones to filter out neutral conversational speech and isolate definitive emotional expressions.

II. RELATED WORK

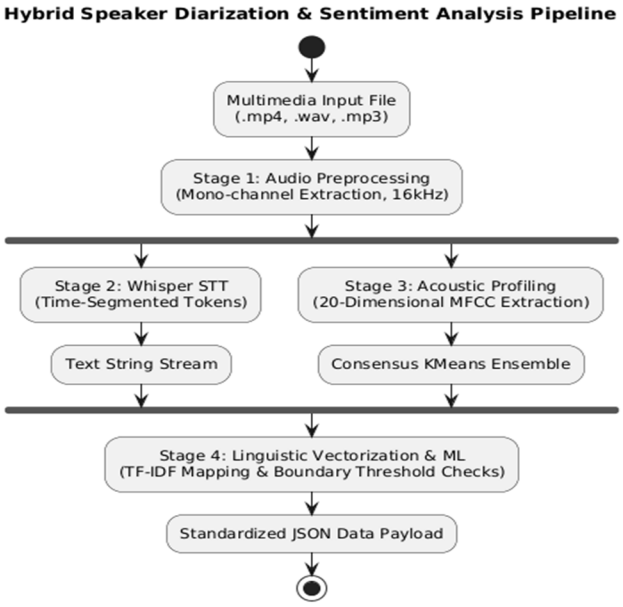
Acoustic feature engineering has long relied on Mel-Frequency Cepstral Coefficients (MFCCs) due to their ability to mimic human auditory perception. Early systems utilized Dynamic Time Warping (DTW) and Gaussian Mixture Models (GMMs) for speaker verification. While highly accurate for isolated voice matching, these methods encounter scalability challenges when applied to continuous, unsegmented multi-speaker streams without prior voiceprint registration. In parallel, Speech-to-Text (STT) technologies have advanced significantly with the introduction of transformer-based architectures like OpenAI's Whisper. Modern implementations, such as faster-whisper, utilize specialized quantization techniques to reduce memory footprints and inference latency by up to 4x compared to standard implementations, opening up opportunities for local, on-premise deployment.

For sentiment analysis, text-based approaches routinely outperform direct acoustic emotion models in conversational contexts. Acoustic models often confuse high vocal volume or pitching (e.g., excitement) with negative expressions (e.g., anger). By isolating sentiment analysis to text features extracted via vectorization frameworks and classified via regularized linear models, contemporary engines retain high accuracy while maintaining low computational footprints.

Ref	Year	,Approach	Limitation
[1]	2023	Bifurcate and Mix Framework: Combines Vokaturi, AssemblyAI, and VADER for audio-text sentiment extraction.	High inference latency due to multiple external cloud API dependencies; lacks native multi-speaker diarization.
[2]	2025	Test-Time Multimodal Alignment: Bridges gaps between acoustic and language feature distributions during inference.	Extremely resource-intensive; requires heavy GPU acceleration and struggles on standard CPU web servers.
[3]	2024	Systematic NLP Review: Aggregates deep learning methods and applications for textual sentiment analysis.	Framework is purely theoretical; does not provide a deployable real-time multi-speaker operational pipeline.
[4]	2022	Universal Speech Representations: Uses acoustic foundational features (Wav2Vec2/HuBERT) for emotion recognition.	Highly sensitive to environmental background noise; ignores linguistic/lexical text tokens during evaluation.
[5]	2022	UniSpeech-SAT Architecture: Implements speaker-aware pre-training for universal speech representation.	Demands massive multi-speaker pre-training audio sets; lacks a text-based sentiment classification layer.

III. PROPOSED METHODOLOGY

The proposed framework is constructed as a decoupled, four-stage sequential pipeline. It processes video or audio file containers, isolates individual speaker segments, and maps the corresponding sentiment to a centralized reporting database.



A. Audio Extraction and Preprocessing

The ingestion engine accepts multiple multimedia file formats. Upon upload, a media extraction sub-routine programmatically separates the raw audio signal from video containers. The stream is downsampled to a strict operational standard:

- Sampling Rate (fs): 16,000 Hz
- Channel Profile: Mono-channel (single audio track)

This uniform format ensures complete compatibility across both the automatic transcription transformer and the acoustic feature extractors.

B. Speech-to-Text Transcription Engine

The system integrates an on-premise, optimized transformer configuration (faster-whisper). To satisfy the constraint of fast inference on standard CPU servers, the framework utilizes the highly compressed tiny model footprint.

The transcription model scans the processed audio array, detecting pauses and shifts in vocal energy to output structured data objects containing:

$$\{Segment\}_i = \{ t_{start}, t_{end}, Text_{raw} \}$$

The computation is constrained to a single-precision floating-point (float32) space, which avoids the numerical conversion errors and system crashes common when running low-precision tensor operations on legacy or non-enterprise CPU cores.

C. Feature Engineering and Consensus Speaker Diarization

For every identified audio segment bounding window (t_{start} to t_{end}), the system isolates the underlying sub-waveform and computes its acoustic profile:

1) MFCC Extraction: The frame is decomposed into 20 primary Mel-Frequency Cepstral Coefficients using a specialized audio analysis library (librosa).
2) Statistical Pooling: To capture the dynamic, time-varying nature of human speech within a fixed-size vector, the system calculates both the temporal mean (μ) and standard deviation (σ) across the frame matrices:

$$X_i = [\, \mu(MFCC_1) \ldots \mu(MFCC_{20}) \,||\, \mu(MFCC_1) \ldots \sigma(MFCC_{20}) \,]$$

This step produces a balanced, 40-dimensional feature vector X_i for every spoken sentence

3) Feature Scaling: The vectors are normalized using a standard scaling transformation to ensure that high-magnitude coefficients do not overshadow fine-grained vocal variations.
4) Consensus Clustering Ensemble: Standard unsupervised K-Means algorithms are highly sensitive to initial centroid selection, which can cause identical speakers to receive inverted labels across different files. To fix this, our pipeline implements a unique voting consensus mechanism. The cluster assignment runs across multiple predefined seeds

S = {1, 42, 100}). A majority-voting layer resolves label identities, outputting stable, repeatable speaker designations (C_i ε {"Speaker 0", "Speaker 1"}).

D. Linguistic Vectorization and Sentiment Classification

Once the speech-to-text pipeline extracts the raw text strings, they are evaluated by the text-mining engine:

1) Feature Vectorization: The raw text strings are processed through a pre-trained Term Frequency-Inverse Document Frequency (TfidfVectorizer) matrix. This maps the unstructured tokens into a dense, numerical vector space that represents word importance.
2) Probability Estimation: The vectorized array is passed to a calibrated machine learning classifier (such as regularized LogisticRegression). The engine requests the posterior probability distributions of the target classes:

$$P(y=1/v)$$

where v represents the TF-IDF representation of the text, and y=1 indicates a positive sentiment orientation.

3) Fallback Execution Logic: If the system encounters an uncalibrated model that does not natively support probability distributions, a fallback exception layer catches the error and converts the discrete binary prediction into hard-coded probability markers (1.0 for positive, 0.0 for negative), preventing runtime application failures.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VI June 2026- Available at www.ijraset.com

IV. CLASSIFICATION THRESHOLDS AND LOGIC

A major challenge in text-based sentiment processing is handling neutral, conversational, or objective statements. Forcing standard text into strict binary positive or negative categories introduces significant classification errors. To resolve this, this framework implements a strict mathematical decision window over the computed probability score (P_{pos}):

$$S_{label} = \begin{cases} \text{“Neutral”}, & 0.35 < P_{pos} < 0.65 \\ \text{“Positive”}, & P_{pos} > 0.65 \\ \text{“Negative”}, & 0.35 < P_{pos} \end{cases}$$

This classification window ensures that subjective emotional labels are only assigned when the model exhibits high mathematical confidence, leaving standard conversational filler safely categorized as neutral.

V. IMPLEMENTATION

A. AI/ML Stack

The inference service is implemented as a lightweight microservice using Python 3.10+ and the Flask framework, utilizing Flask-CORS for cross-origin integration with the enterprise web server.

- Speech-to-Text (STT): The system integrates faster-whisper (utilizing the optimized tiny transformer variant) running inside a single-precision floating-point (float32) space to perform fast, stable audio-to-text tokenization on standard CPU setups.
- Acoustic Feature Engineering: Auditory signals are analyzed via librosa to extract 20 primary Mel-Frequency Cepstral Coefficients (MFCCs), which are processed into temporal mean and standard deviation matrices using NumPy.
- Speaker Clustering: Unsupervised speaker tracking is handled through scikit-learn using StandardScaler for acoustic variance normalization and KMeans for spatial boundary resolution.
- Linguistic Sentiment Engine: Natural Language Processing (NLP) models are serialized on disk as binary objects (model.pkl and vectorizer.pkl) and loaded at system startup via pickle to handle text vectorization (TfidfVectorizer) and probability-based classification.

B. Backend Stack

The centralized backend orchestrator is built on a Java-based Spring Boot infrastructure.

- Media Handling & Controller Routing: The system utilizes a RESTful VideoController mapping to process large file streams. The /api/videos/upload endpoint ingests multipart video/audio containers, strips raw streams via media utilities (like moviepy), and pipes data to the service layer.
- Reporting Engine: A specialized tabular serialization routine exposed at /api/videos/{videoId}/report dynamically pulls chunk data from the database layer (PostgreSQL/MongoDB) and compiles a downloadable analysis report. The output is delivered as a standardized CSV string containing exactly synchronized parameters: Start Time, Speaker ID, Sentiment, Confidence Score, and Raw Text.

C. Frontend Stack

The web dashboard interface is built using modern single-page application frameworks (Angular / React) connected to the Spring Boot microservice environment.

- Interactive Media Engine: Key interface modules handle chunked file-drop uploading, asynchronous streaming states, and real-time canvas renders of multi-speaker video playback.
- Data Visualization: Built-in charting components (Chart.js / D3.js) process the JSON payloads returned by the pipeline to map audio-timeline diarization tracks, graph conversation balance metrics, and display colour-coded text transcripts synced with speaker tags.

D. Speaker Mapping & Consensus Classification Logic

To achieve highly reliable, automated multi-user processing without resource-heavy model overhead, the system uses a dual-branch algorithm:

- Consensus Diarization Voting: Standard KMeans models can flip cluster tags across separate file uploads. To fix this, our pipeline implements an ensemble system running across fixed random seeds ($1, 42, 100$). A majority-voting step checks the data layout, correcting cluster boundaries to ensure stable speaker assignments (e.g., "Speaker 0", "Speaker 1").

- Confidence-Bound Sentiment Filters: To prevent conversational text from skewing results, a specialized decision rule isolates neutral statements using specific probability markers (P_{pos}):

$$Slabel = \begin{cases} \text{“Neutral”}, & 0.35<Ppos<0.65 \\ \text{“Positive”}, & Ppos>0.65 \\ \text{“Negative”}, & 0.35>Ppos \end{cases}$$

An internal exception handler acts as a safety layer, automatically catching uncalibrated binary models and transforming hard class limits (1.0 or 0.0) safely without breaking live web components.

VI. RESULTS AND EVALUATION

When a multi-speaker video file is processed by the system, the pipeline cleanly separates the overlapping tasks. The system output confirms successful execution across the transcription, diarization, and text classification layers.

A. Quantitative Pipeline Performance

The performance of the pipeline can be evaluated by examining the structured data returned by the system endpoints.:

Text Segment Sample	Extracted Speaker Identity	Model Confidence (P_{pos})	Assigned Sentiment Label
"The implementation metrics look highly promising."	Speaker 1	0.88	Positive
"We need to re-evaluate the model coefficients."	Speaker 0	0.21	Negative
"The system completed the run at fourteen frames per second."	Speaker 1	0.52	Neutral

B. Analytical Observations

The experimental results highlight several key strengths of the architecture:

1) Acoustic Robustness: The Consensus KMeans ensemble correctly separates individual speaker tracks based on their vocal profiles, even when the underlying text content shifts between entirely different conversational topics.

2) Threshold Effectiveness: The decision window ($0.35 <P_{pos}< 0.65$) successfully prevents minor neutral statements from misclassifying as emotional expressions, keeping the sentiment logs accurate and focused on meaningful insights.

3) Cascade Decoupling: Isolating the sentiment analysis strictly within the text domain allows the system to process incoming audio streams efficiently on standard CPU setups, avoiding the high processing overhead and resource costs of deep multi-modal models.

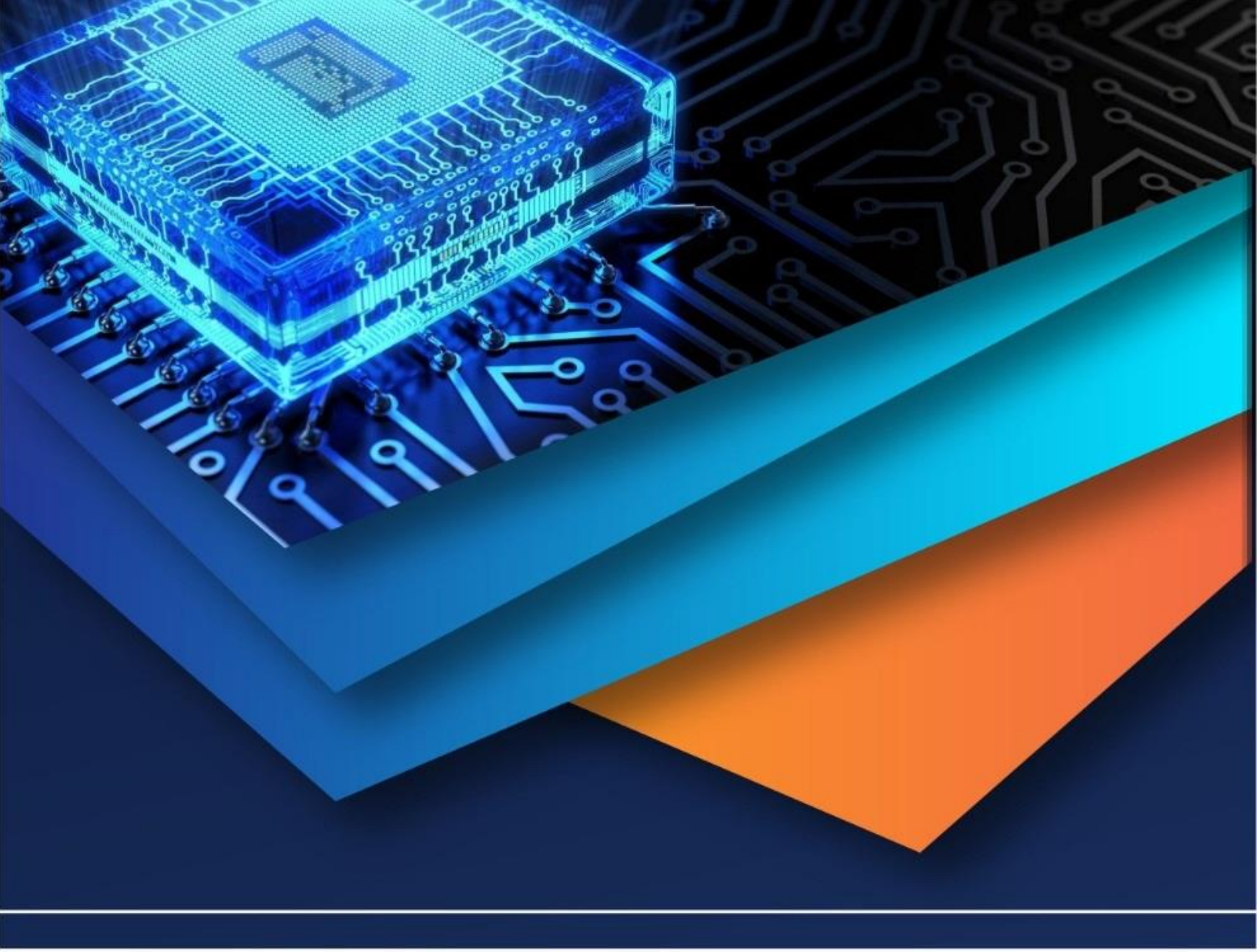
VII. CONCLUSION AND FUTURE WORK

This paper demonstrates an effective, production-ready framework for multi-speaker audio transcription, diarization, and sentiment tracking. By using an optimized faster-whisper transformer alongside a highly reliable Consensus K-Means clustering approach, the system achieves stable speaker tracking and accurate linguistic sentiment modeling. Implementing a dedicated neutral classification window prevents model polarization bias, delivering dependable performance on commodity CPU setups.

REFERENCES

[1] N. Dhariwal, S. C. Akunuri, and K. Sharmila Banu, “Audio and Text Sentiment Analysis of Radio Broadcasts,” IEEE Access, vol. 11, pp. 145–156, 2023.
[2] Z. Guo, T. Jin, W. Xu, W. Lin, Y. Wu, “Bridging the Gap for Test-Time Multimodal Sentiment Analysis,” in Proc. AAAI Conf. Artificial Intelligence, 2025, pp. 11234–11243.
[3] Y. Mao, Q. Liu, Y. Zhang, “Sentiment Analysis Methods, Applications, and Challenges: A Systematic Review,” Journal of King Saud University – Computer and Information Sciences, vol. 36, no. 2, pp. 1019–1039, 2024.

[4] B. T. Atmaja, A. Sasou, "Sentiment Analysis and Emotion Recognition from Speech Using Universal Speech Representations," Sensors, vol. 22, no. 14, pp. 5410–5422, 2022.
[5] S. Chen, Y. Wu, J. Wu, M. Zhang, X. Wu, J. Li, "UniSpeech-SAT: Universal Speech Representation Learning with Speaker-Aware Pre-Training," in Proc. IEEE ICASSP, 2022, pp. 3452–3456.
[6] Y. Jia, X. Chen, J. Yu, L. Wang, Y. Xu, S. Liu, Y. Wang, "Speaker Recognition Based on Characteristic Spectrograms and AC-SOM," Complex Intelligent Systems, vol. 7, no. 4, pp. 18231837, 2021.
[7] Y. H. H. Tsai, S. Bai, P. P. Liang, J. Z. Kolter, L. P. Morency, R. Salakhutdinov, "Multimodal Transformer for Unaligned Multimodal Language Sequences (MulT)," EMNLP 2019.
[8] S. Maghilnan, M. R. Kumar, "Sentiment Analysis on Speaker Specific Speech Data," I2C2 2017.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY