Audio classification for feature-based majority voting optimization and hyperparametric tuning

Lorena Telembici
Signal Processing Group, Basis of Electronics Dept.
Technical University of Cluj-Napoca, Romania
https://orcid.org/0000-0002-7206-0198

Corneliu Rusu
Signal Processing Group, Basis of Electronics Dept.
Technical University of Cluj-Napoca, Romania
https://orcid.org/0000-0001-5209-2734

***Abstract*—This paper presents an optimized audio recognition system that integrates feature-based and deep learning approaches, fine-tuned for high-accuracy classification. The study builds upon previous research, where all possible feature-classifier combinations were analyzed to determine the most effective configurations. Based on these findings, we focus on MFCC-34 and MFCC-38 for SVM and kNN, as well as spectrograms and Mel-spectrograms for CNN, forming six possible model combinations. A majority voting mechanism is implemented to enhance classification robustness. While grid search was previously applied to SVM and kNN, this work further refines the system by performing hyperparameter tuning for CNN, optimizing Conv2D filters, layer units, dense layer size, learning rate, dropout, and optimizer type. Additionally, the number of epochs is systematically tested from 10 to 30 in steps of 5 to determine the optimal training duration. The final implementation follows a structured pipeline, including data preparation, feature extraction, model training, evaluation and deployment preparation. The system is validated using multiple performance metrics, tuning process visualizations, and a 5-fold cross-validation repeated 20 times. Results demonstrate the effectiveness of our approach, achieving 97.65% accuracy for CNN, 97.42% for SVM, and 95.53% for kNN, confirming the reliability of the proposed majority voting-based classification system.**

***Index Terms*—Audio recognition, MFCC, spectrograms, Mel-spectrograms, CNN, SVM, kNN, Hyperparameter tuning, Majority voting, Classification Performance, Box and whisker plot, audio-based assistive systems**

I. Introduction

Audio recognition has become an essential component of modern artificial intelligence applications, enabling various real-world use cases such as speech recognition, environmental sound classification [1], speaker identification, and assistive technologies. Among these, audio-based assistive systems for elderly individuals [2] play a crucial role in improving accessibility, safety, and daily communication. Recognizing and interpreting sounds accurately allows intelligent systems to respond effectively to user needs, making speech-based interfaces, emergency detection, and smart home automation more efficiently. Although the method presented in this paper can be implemented in other scenarios, here it is evaluated on an audio dataset intended for Romanian-speaking patients (Section II-A).

Traditional audio classification approaches rely on hand-crafted features such as Mel-Frequency Cepstral Coefficients (MFCC). However, the rise of deep learning-based models, particularly Convolutional Neural Networks (CNNs), has significantly improved classification performance by learning hierarchical representations from spectrograms. Existing audio classification systems often face several challenges, including the selection of the most effective feature representation [3], optimization of classifier parameters [4], computational efficiency for real-time applications, and the need for robust decision-making strategies. Selecting the right features, such as MFCCs or spectrograms, greatly influences the performance of different classifiers, and tuning hyperparameters is essential for maximizing model accuracy. Finally, relying on a single model for classification may lead to misclassifications in certain cases, making ensemble techniques such as majority voting a promising solution. This study aims to address these challenges by designing an optimized audio recognition framework that integrates traditional machine learning and deep learning models while leveraging hyperparameter tuning and majority voting for classification.

Previous studies have conducted extensive evaluations of various feature-classifier combinations, testing multiple feature types, including MFCC, LPC, LPCC, MPEG-7, PLP, and RASTA-PLP, with dimensions ranging from 10 to 38 in steps of 2, as well as 64 features. These features were tested with eight traditional classifiers, as well as three deep learning models, namely ANN, CNN, and LSTM-RNN. The findings demonstrated that MFCC-34 and MFCC-38 were the most effective feature representations for SVM and kNN, while spectrograms and Mel-spectrograms yielded the highest performance for CNN. Grid search was applied to optimize SVM and kNN parameters, leading to significant improvements in classification accuracy. These findings serve as the foundation for this study, which further refines the system by performing hyperparameter tuning for CNN and evaluating the best combination of classifiers through majority voting (see [5] and references therein).

Building on prior research, this paper proposes an optimized audio recognition system that integrates MFCC-34 and MFCC-38 for SVM and kNN, along with spectrograms and Mel-spectrograms for CNN, to ensure optimal feature selection. This study focuses on hyperparameter tuning for CNN and optimizing key parameters. Furthermore, the impact of the number of training epochs is systematically analyzed.

A majority voting-based decision mechanism is implemented across six model configurations to enhance classification robustness and ensure greater reliability in predictions. The final implementation follows a structured pipeline consisting of data preparation, feature extraction, model training, evaluation, and deployment preparation. Results confirm the effectiveness of this approach, with the optimized model achieving a classification accuracy of 97.65% for CNN, 97.42% SVM and 95.53% for kNN, demonstrating the reliability of the proposed majority voting-based classification system.

The remainder of this paper is structured as follows. The next section presents an analysis of the dataset, the feature extraction process, classifier selection, and hyperparameter tuning. The following sections describe the experimental setup, including the implementation pipeline and performance metrics (Section III), as well as a discussion of the results (Section IV). Finally, the paper concludes with a summary of findings and potential future research directions (Section V).

II. AUDIO ANALYSIS, MODEL DEVELOPMENT AND OPTIMIZATION

A. Dataset

The dataset used in this study consists of 26,640 audio signals, meaning 148 classes, recorded by both male and female voices in Romanian. The primary focus is on recognizing the names of medicines, as the final optimized audio recognition system is intended for an assistive robot. The dataset is structured into five different scenarios, each representing a distinct category of sound events. The detailed breakdown is as follows:

- Kitchen Scenario – Consists of 9 classes, with each class containing 30 original sound events, which were expanded using 5 data augmentation methods, resulting in a total of 1,620 audio signals (9 × 30 × 6 sets).
- Room Scenario – Includes 11 classes, each with 30 original sound events, augmented into 1,980 audio signals (11 × 30 × 6 sets).
- Appliances Scenario – Covers 5 classes, with augmentation resulting in 900 audio signals (5 × 30 × 6 sets).
- Voice Scenario – Represents 117 classes, primarily focusing on spoken words, particularly medicine names. With augmentation, this category contributes the largest portion of the dataset, totaling 21,060 audio signals (117 × 30 × 6 sets).
- Non-Verbal Scenario – Comprises 6 classes, including non-verbal sounds such as breathing, coughing, and other relevant noises. This section contains 1,080 audio signals (6 × 30 × 6 sets).

Data augmentation methods include pitch shifting, speed variation, noise injection, time stretching, and time-frequency masking, ensuring that the system performs robustly under diverse real-world conditions. Note the dataset's composition and the dominance of the voice scenario, which is critical for the final application of medicine name recognition.

B. Feature Extraction

Feature selection plays a crucial role in optimizing the performance of the audio recognition system. Based on previous research findings and experimental results, this study focuses on two primary feature extraction techniques: Mel-Frequency Cepstral Coefficients (MFCCs) and Spectrogram-based representations.

MFCCs have been widely used in speech and audio processing due to their ability to capture essential frequency characteristics relevant for classification tasks [6]. After extensive evaluation of different feature subsets, two optimal configurations were selected: MFCC-34 and MFCC-38. MFCC-34 includes 34 selected coefficients extracted from each audio signal. The heatmap in Fig. 1 visualizes the distribution of these coefficients across multiple samples. The feature heatmap in Fig. 2 illustrates the variation of MFCC-38 values. These selected MFCC representations serve as inputs for traditional machine learning classifiers, specifically SVM and kNN, which demonstrated superior performance in prior studies.

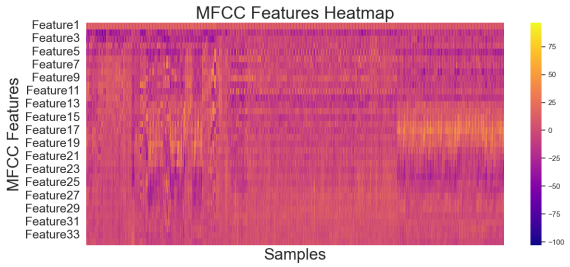

Fig. 1. MFCC-34 Feature Representation; Heatmap of Extracted Coefficients.

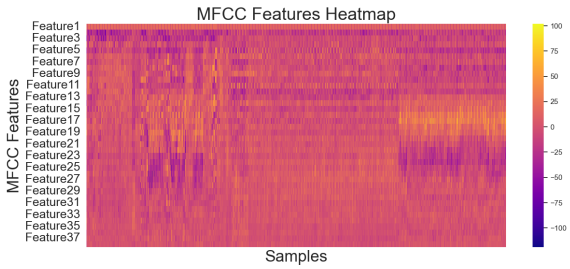

Fig. 2. MFCC-38 Feature Representation; Heatmap of Extracted Coefficients.

Fig. 3 and 4 illustrate the comparison between amplitude-based and dB-based spectrograms. The amplitude-based spectrogram represents raw energy distribution over time and frequency, while the dB-based spectrogram provides a logarithmic scaling that enhances the visibility of lower-energy frequency components. This transformation is essential for audio recognition as it aligns more closely with human auditory perception, making features more distinguishable and robust against variations in signal amplitude. Consequently, dB-scaled spectrograms are utilized in our implementation to improve classification accuracy. For further analysis, Fig. 4

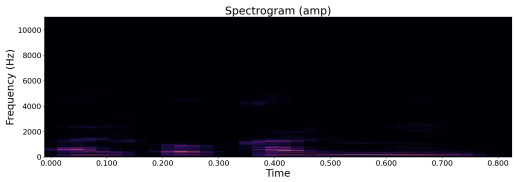

Fig. 3. Spectrogram Representation – Signal Amplitude over Time and Frequency for *algocalmin* recorded by man voice.

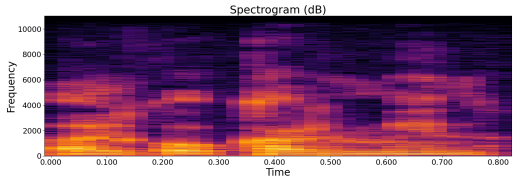

Fig. 4. Spectrogram Representation – Log-Scale dB Visualization for *algocalmin* recorded by man voice.

and Fig. 5 illustrate an example of the word "Algocalmin", recorded by a male speaker, displayed in both Spectrogram and Mel-Spectrogram formats. These visualizations highlight the distinctive patterns used by CNN models for classification. By leveraging these carefully selected feature sets, the proposed system ensures high classification accuracy while maintaining a computationally efficient framework for real-time assistive applications.

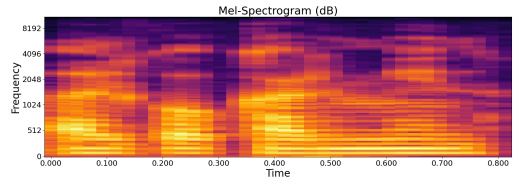

Fig. 5. Mel-Spectrogram Representation – Log-Scale dB with 64 mels for *algocalmin* recorded by man voice.

C. Classifier Selection

We selected SVM, kNN, and CNN based on their effectiveness in prior studies [5]. SVM and kNN excel in structured feature-based classification, making them suitable for MFCC features, while CNN leverages spatial patterns in spectrograms and Mel spectrograms. Grid search and hyperparameter tuning further optimized performance.

Additionally, the choice of classifiers was influenced by their ability to generalize across diverse acoustic conditions. SVM and kNN, known for their strong performance with smaller feature sets, effectively handle MFCC-based representations [7]. CNN, on the other hand, excels in learning hierarchical patterns from spectrograms, making it ideal for capturing time-frequency dependencies. The combination of these approaches ensures a balance between interpretability, computational efficiency, and high classification accuracy, ultimately enhancing the robustness of the recognition system.

D. Hyperparameter Tuning

Hyperparameter tuning played a crucial role in optimizing classifier performance. For SVM and kNN, a grid search was conducted to determine the best kernel functions, distance metrics, and regularization parameters [5], leading to significant accuracy improvements. CNN tuning focused on optimizing the number of convolutional filters, dense layer units, dropout rates, learning rate and optimizer type. A systematic evaluation of epoch values (ranging from 10 to 30 in steps of 5) ensured optimal convergence while preventing overfitting. This fine-tuning process significantly enhanced model performance, making the final system both accurate and efficient for real-time deployment.

III. EXPERIMENTAL SETUP AND EVALUATION

A. Implementation Pipeline

The implementation pipeline follows a structured sequence of steps to ensure optimal model training, evaluation, and deployment. Fig. 6 outlines the key stages in the pipeline, from data preparation to deployment. Each stage is designed to handle specific tasks that contribute to the robustness of the final model. The process begins with the dataset, followed by

Data Preparation	Create or reset training and testing directories
	Organize data into training and testing sets
Feature Extraction and Image Generation	Extract MFCC features and save arrays
	Generate and save Spectrograms
	Generate and save Mel Spectrograms
Train and Save Models	Train and save *k*NN models
	Train and save SVM models
	Train and save CNN models
Model Evaluation	Generate classification reports
	Calculate CM, ROC curves, AUC scores and metrics
	Visualize and save evaluation results
Deployment Preparation	Organize trained model files
	Copy essential scripts for deployment
	Prepare deployment directories

Fig. 6. End-to-End Pipeline for Feature Extraction, Training, and Evaluation.

feature extraction, where MFCC-34, MFCC-38, Spectrograms, and Mel-Spectrograms are extracted. These extracted features are then processed by their respective classifiers—SVM and kNN for MFCCs and CNN for Spectrogram-based features. The models undergo hyperparameter tuning, followed by a comprehensive performance evaluation. A key aspect of this pipeline is the majority voting mechanism, which aggregates the best-performing models to form the final optimized system. Additionally, a validation process is performed using box-and-whisker plots with 5-fold cross-validation repeated 20 times to ensure model consistency and robustness. The final optimal model selection is based on the highest-performing configurations (Table I).

For testing, 20% of the dataset is used, corresponding to 36 audio signals. These were not randomly chosen but carefully

TABLE I
CLASSIFICATION ACCURACY [%] FOR 26,640 AUDIO DATASET

Classifier	Number of features	Metric	Before Tuning	After Tuning
SVM	MFCC-38	Accuracy	96.4	97.42
	MFCC-34	Accuracy	96.36	97.1
kNN	MFCC-38	Accuracy	93.38	95.53
	MFCC-34	Accuracy	93.58	95.11
CNN	Spectrogram	Accuracy	96.46	97.01
	Mel-Spectrogram	Accuracy	95.83	97.65

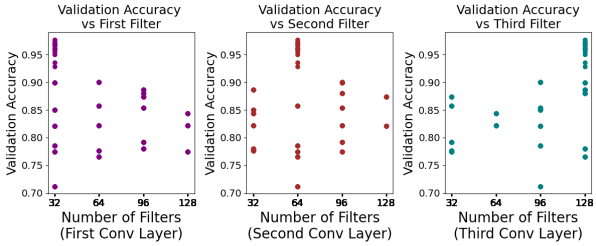

Fig. 7. Validation Accuracy vs. Number of Filters/Conv 2D Layer.

selected to ensure generalization. Each class includes sounds from all augmentation sets (original, noise injection, speed variation, loudness variation, pitch shifting, time/frequency masking), ensuring balanced representation. 6 audio signals from each augmentation set were equally distributed, considering different distances from the microphone during recording. This ensures the model generalizes well to real-world conditions, making it suitable for deployment in assistive robotic systems.

B. Performance Metrics

The performance evaluation of the optimized audio recognition system was conducted using multiple classification metrics to ensure a comprehensive assessment. The evaluation included standard metrics such as accuracy, balanced accuracy, classification report, confusion matrix, Matthews correlation coefficient (MCC), Cohen's kappa, and log loss. For each model, predictions were compared against ground truth labels, and confusion matrices were generated to visualize misclassifications. Additionally, ROC curves and AUC scores were computed to analyze the models' ability to distinguish between classes effectively.

IV. RESULTS AND DISCUSSION

Fig. 7 presents the impact of varying the number of filters in the three convolutional layers on validation accuracy. Each subplot corresponds to one of the convolutional layers, showing the number of filters (X-axis) and the resulting validation accuracy (Y-axis).

- The first subplot (left) shows how the accuracy changes as the number of filters in the first convolutional layer varies.
- The second subplot (middle) focuses on the second convolutional layer.
- The third subplot (right) represents the third convolutional layer.

After analyzing the trends, the optimal filter sizes were determined as 32 for the first layer, 64 for the second layer, and 128 for the third layer. These values provided the highest validation accuracy before proceeding with further hyperparameter tuning.

Fig. 8 presents an overview of multiple experimental trials conducted during hyperparameter tuning, allowing for a detailed comparison of model performance across different settings. Each point in the plot corresponds to a specific trial where a unique combination of parameters—such as the number of convolutional filters, dropout rates, learning rates, and optimizer types—was evaluated.

Based on the insights gained from this analysis, the final CNN model configuration for further evaluation will use: **Number of filters**: 32 on 1st layer, 64 on 2nd layer, 128 on 3rd layer; **Dense layer**: 256; **Dropout rate**: 0.5; **Optimizer**: Root Mean Square Propagation; **Learning rate**: 0.001; **Epochs**: 15 With these optimized hyperparameters, the final model was implemented following the structured pipeline outlined earlier. The dataset was split into 80% for training and 20% for testing, ensuring a robust evaluation of the model's generalization capabilities.

The ROC curve plots from Fig. 9 the True Positive Rate (TPR) against the False Positive Rate (FPR), illustrating the classifier's performance at different thresholds for kNN with MFCC-38. The AUC score of 0.9931 indicates an excellent classification performance, as an AUC close to 1 suggests the model is highly capable of distinguishing between positive and negative instances of the "algocalmin" class with minimal errors. kNN with MFCC-34 achieved an accuracy of 94.76% and a balanced accuracy of 94.43%, while kNN with MFCC-38 slightly improved to 94.91% accuracy and 94.53% balanced accuracy. Additionally, MCC and Cohen's Kappa values were consistently high for both kNN models, with MFCC-38 providing marginally better stability and lower log loss, indicating improved probability estimates. For SVM models, SVM with MFCC-34 achieved 94.33% accuracy and 94.14% balanced accuracy, whereas SVM with MFCC-38 further improved performance to 95.46% accuracy and 95.23% balanced accuracy. MCC and Cohen's Kappa values confirm strong consistency in classification, with SVM outperforming kNN, particularly in handling complex feature variations. The variability in accuracy scores across 20 repeated 5-fold cross-validations, showing the consistency and performance of the SVM classifier when using 38 MFCC features are in Fig. 10. The green triangles represent the mean accuracy for each repeat, with whiskers indicating the range of scores.

V. CONCLUSION

This study successfully developed an optimized audio recognition system for assisting elderly individuals in identifying spoken medicine names in Romanian. By evaluating MFCC-38, MFCC-34, and Spectrogram-based features, along with SVM, kNN and CNN, we identified the most

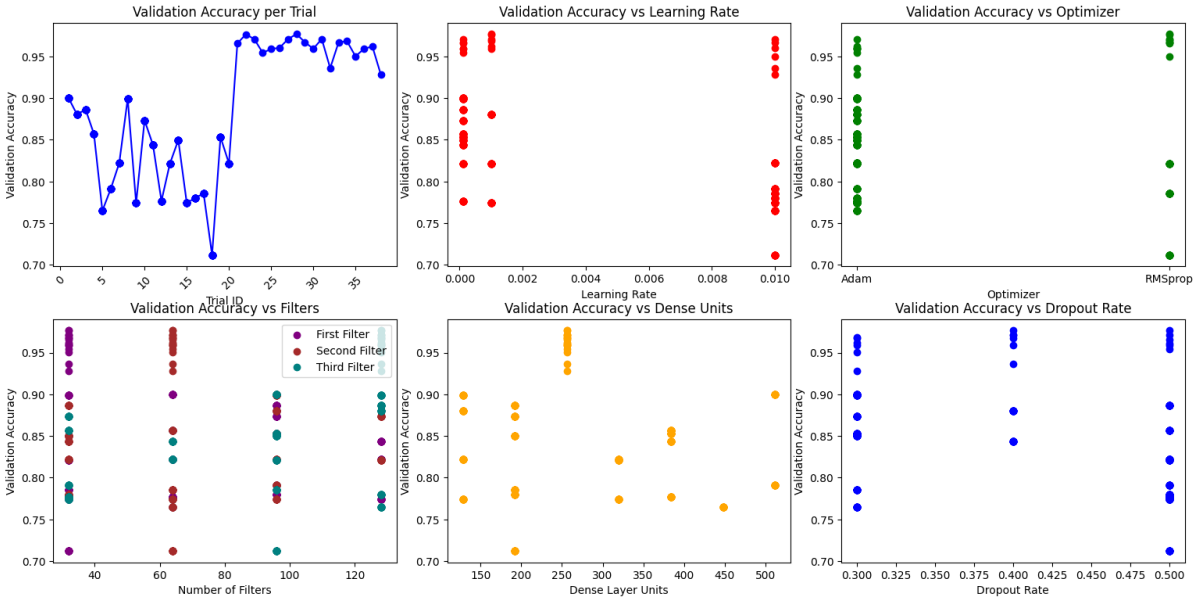

Fig. 8. All Trials Performance Comparison.

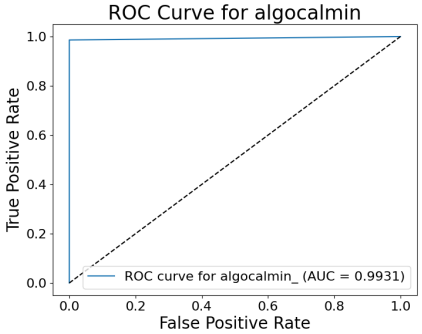

Fig. 9. ROC Curve for *Algocalmin Class* – Model Performance Evaluation.

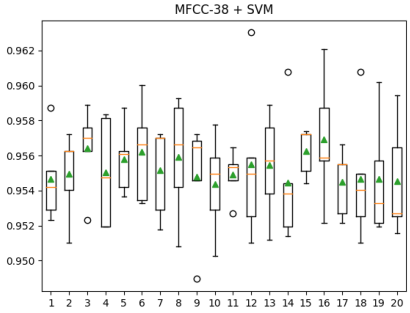

Fig. 10. MFCC-38 with SVM: Results 5-Fold Cross-Validation 20 Repeats.

effective configurations for high-accuracy speech recognition. After hyperparameter tuning, the best results were 97.65% for CNN (Mel-Spectrogram), 97.42% for SVM (MFCC-38), and 95.53% for kNN (MFCC-38). The final implementation of the optimal models where we take care what samples to be tested to 98.02% for CNN, 97.12% for SVM, and 96.2% for kNN, demonstrating the effectiveness of the selected features and tuning process. The final models were assessed using multiple performance metrics: balanced accuracy, classification reports, MCC, Cohen's kappa, log loss, and 5-fold cross-validation (repeated 20 times). The box-and-whisker plot analysis for SVM confirmed the model's stability across multiple runs.

Future enhancements could include expanding the dataset with more diverse speaker variations to improve generalization and deploying the system in real-world environments to evaluate robustness in practical scenarios.

References

[1] L. S. Puspha Annabel, S. P. G, and T. V, "Environmental sound classification using 1-D and 2-D convolutional neural networks," in *2023 7th Int. Conf. on Electronics, Communication and Aerospace Technology (ICECA)*, 2023, pp. 1242–1247.

[2] M. Geng, X. Xie, Z. Ye, T. Wang, G. Li, S. Hu, X. Liu, and H. Meng, "Speaker adaptation using spectro-temporal deep features for dysarthric and elderly speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 2597–2611, 2022.

[3] H. Redelinghuys and Z. Wang, "Evaluating audio features for speech/non-speech discrimination," in *2022 First Int. Conf. on Artificial Intelligence Trends and Pattern Recognition (ICAITPR)*, 2022, pp. 1–6.

[4] S. Padman and D. Magare, "Performance evaluation of various deep learning techniques with learners memorization optimization algorithm for multi-modal speech emotion detection," in *2024 Second Int. Conf. on Intelligent Cyber Physical Systems and Internet of Things (ICoICI)*, 2024, pp. 1489–1494.

[5] L. Muscar, T. Telembici, and C. Rusu, "Deep learning-based sound classification algorithms for enhanced service robots audio capabilities," in *2024 15th International Conference on Communications (COMM)*, 2024, pp. 1–6.

[6] G. Jo, H. Lim, and S. Yoon, "Combining wavelet and MFCC features for emotion recognition in conversation," in *2023 IEEE International Conference on Big Data (BigData)*, 2023, pp. 6178–6179.

[7] B. S. N. Reddy, B. S. Venkat, I. Manohar, S. Abhishek, and A. T, "Aural signatures: Audio fingerprinting techniques for real-time audio recognition," in *2024 3rd Int. Conf. on Sentiment Analysis and Deep Learning (ICSADL)*, 2024, pp. 22–30.