PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH
SCIENTIFIC RESEARCH

MOHAMED BOUDIAF UNIVERSITY - M'SILA

FACULTY :TECHNOLOGY
DOMAIN: COLLEGE OF TECHNOLOGY
DEPARTMENT: ELECTRONICS
FIELD OF STUDY: AUTOMATIC

OPTION: AUTOMATIC AND SYSTEMES

THESIS SUBMITTED FOR THE ATTAINMENT
OF THE ACADEMIC MASTER'S DEGREE

Presented by:

HAROUN KHEIR
YASSINE ZELLAGUI

TITLE

AUTOMATIC KEYWORD DETECTION AND CRIME PREDICTION FROM PHONE CALL ANALYSIS (SEERIUM)

Defended before the jury composed of:

Dr. SAHED Mohamed	**University of M'sila**	**President**
Dr. HERIZI Abdelghafour	**University of M'sila**	**Supervisor**
Dr. ABED Ahcene	**University of M'sila**	**Co-Supervisor**
Dr. LAIB Abderrzak	**University of M'sila**	**Examiner**

Academic year: 2024 / 2025

Acknowledgments

We would like to express our sincere gratitude to all those who supported and guided us throughout this project.

Special thanks go to our supervisors, Mr. Abed Ahcene and Mr. Abdelghafour Herizi, for their valuable guidance, encouragement, and continuous support from the beginning to the end of this work.

We are also thankful to Ms. Ben Toumi Sarah, Manager of the Business Incubator of M'sila, for her trust and generous support, as well as to Mr. Chouder Aïssa, our dear teacher, whose presence and advice always motivated us.

We warmly thank Yassine Khier for his assistance in information-related aspects, and all the individuals who provided their voice samples—your contributions were essential to building our database.

Finally, we extend our gratitude to all members of the Business Incubator of M'sila for an inspiring and enriching experience filled with learning, challenges, and teamwork. Together, we didn't just build a project—we built a family.

Dedications

To those who instilled in us values and principles,

To the light of our lives—our parents,

I dedicate this humble work to my beloved father, whose strength and support have always been my foundation, and to the eternal memory of my dear mother—may Allah have mercy on her soul—whose love, prayers, and sacrifices continue to guide me even in her absence.

To my cherished aunties, Aïcha and Safia, who stood by me like second mothers, offering unwavering encouragement and boundless affection—thank you for believing in me when I doubted myself.

To my brothers and sisters,

To my professors abdelghafour herizi and ahcene abed, whose knowledge has shaped my journey,

And to every soul who contributed, in words, actions, or silent prayers—this achievement is yours as much as it is mine.

This work is the fruit of years of learning, perseverance, and faith. As I stand at this milestone, I carry the name of my family, the values of my upbringing, and the hope that this effort will contribute, even modestly, to a better future.

— Haroun Kheir

Dedications

To my family and friends,

Your unwavering presence, constant encouragement, and profound kindness were indispensable pillars throughout the writing of this thesis. Thank you for believing in me every step of the way.

— Yassine zellagui

Table of Contents

List of Abbreviations

Bi-GRU	:	Bidirectional Gated Recurrent Units
CNN	:	Convolutional Neural Networks
dB	:	décibel
DCT	:	Discrete Cosine Transform
DNN	:	Deep Neural Network
F_0	:	Fundamental Frequency
FFT	:	Fast Fourier Transform
GMM	:	Gaussian Mixture Model
HMM	:	Hidden Markov Model
LSTM	:	Linear Short Time Memory
MFCC	:	Mel Frequency Cepstral Coefficients
MSE	:	Mean Squared Error
ReLU	:	Rectified Linear Unit
RNN	:	Recurrent Neural Netw
STFT	:	Short-Time Fourier Transform

List of Figures

List of Tables

1 General Introduction

Voice recognition technology has become a core component in modern systems, powering applications from virtual assistants and biometric authentication to security surveillance. Conventional systems primarily depend on the physical attributes of speech production—such as vocal tract shape, articulation, and pronunciation style. While effective under ideal conditions, these methods face significant limitations in real-world scenarios. Background noise, speaker health or age-related vocal changes, and the rise of spoofing attacks using synthetic or recorded voices all undermine the reliability of traditional approaches. These challenges have prompted the search for more robust, secure, and adaptable voice recognition techniques.

Voice recognition is an interdisciplinary field that integrates digital signal processing, artificial intelligence, and biometric technologies. It serves a critical function across numerous sectors, including telecommunications, security, healthcare, and financial services. The advent of deep learning models—such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs)—has substantially enhanced speech recognition accuracy, even in adverse acoustic environments, thereby increasing its reliability for real-world applications.

Voice recognition can be broadly categorized into two primary branches:

Speaker Identification, which involves recognizing individuals based on unique vocal characteristics, and Speech Content Analysis, which focuses on extracting and interpreting spoken words.

In recent years, research efforts have increasingly targeted security-related applications. Keyword spotting within speech serves as a proactive mechanism for detecting fraudulent activity, cybercrime, and other illicit behaviors. This has driven the development of sophisticated keyword detection and speaker identification systems that contribute to crime prediction and forensic investigations.

Building upon these advancements, this thesis focuses on leveraging frequency-based voice features in conjunction with deep learning techniques to enhance voice recognition accuracy, particularly in noisy and real-world scenarios.

1.1 Keyword Detection

Keyword detection plays a vital role in voice recognition systems, particularly for applications in security and crime prevention where identifying specific trigger words in speech can provide early warnings.

In 2019, Arik et al. introduced Deep Voice 3, a deep learning-based speech recognition model that significantly reduced false positives in noisy environments. Their approach utilized Convolutional Neural Networks (CNNs) to efficiently extract robust speech features, enhancing recognition accuracy under challenging acoustic conditions.

In 2021, Kim and Park proposed a hybrid model combining CNN and Recurrent Neural Networks (RNNs) to enable real-time keyword spotting. Their system demonstrated the capability to detect crime-related keywords in telephone conversations with an accuracy exceeding 92%, showcasing its effectiveness for practical security applications.

1.2 Speaker Identification

Speaker identification is a critical component of voice recognition systems, enabling the recognition of individuals based on their unique vocal characteristics. This capability is especially valuable in law enforcement and forensic applications, where precise speaker attribution can significantly aid investigations.

In 2020, Snyder et al. proposed the x-vector framework, now widely regarded as a standard approach in speaker recognition. Their method combined Mel-Frequency Cepstral Coefficients (MFCCs) with Deep Neural Networks (DNNs), effectively classifying speakers and demonstrating robustness across diverse acoustic conditions.

More recently, in 2022, Zhang et al. introduced a transformer-based speaker recognition model that substantially improved performance on low-quality recordings. This advancement enhanced the practical utility of speaker identification in challenging scenarios such as crime investigations involving degraded audio data.

1.3 Crime Prediction Using Voice Recognition

Recent advancements in deep learning have significantly expanded the capabilities of voice analysis for crime prediction and security monitoring. In 2023, Li et al. developed an LSTM-based speech emotion recognition system capable of detecting aggression and threats in emergency calls, providing early alerts for potentially violent situations. In the

same year, Garcia et al. investigated keyword spotting for fraud detection using Bidirectional Gated Recurrent Units (Bi-GRU) to analyze real-time conversations for suspicious activities. These studies highlight the growing efficacy of deep learning techniques in enhancing the accuracy of voice based crime prediction and speaker verification systems.

1.4 Our Contribution

Building upon these advancements, this project implements a CNN model for keyword detection alongside an MFCC-based speaker identification approach. Notably, the system is specifically trained on the Algerian dialect to address local linguistic variations, thereby improving detection accuracy in crime-related conversations. This dialect-focused adaptation also enhances robustness to dialect-specific speech patterns, making the system more effective in real-world applications.

2 Problem Statement

Despite significant advances, existing voice recognition systems continue to face critical challenges. Conventional approaches heavily rely on individual vocal tract characteristics, rendering them vulnerable to variations caused by environmental noise, emotional states, or health conditions. Additionally, traditional methods often lack the ability to generalize effectively across diverse speakers and languages, limiting their scalability and applicability. Moreover, most current systems primarily focus on speaker identification or verification, with limited integration of real-time keyword detection crucial for security applications.

Beyond identifying speakers, voice analysis can be extended to speech content recognition, particularly for detecting suspicious or harmful keywords. This capability is essential for security domains such as surveillance, fraud detection, and law enforcement, where early identification of specific keywords can aid in preventing criminal activities. However, current voice recognition technologies predominantly emphasize speaker verification, often overlooking the potential of real-time keyword detection for proactive security monitoring.

For example, in counter-terrorism operations, detecting particular words within intercepted communications may provide early warnings of potential threats, enabling timely preventive measures. Similarly, in corporate settings, recognizing fraudulent

discussions related to financial transactions can strengthen anti-fraud strategies and enhance overall security protocols. The integration of keyword detection with speaker identification remains an underexplored area, and this research aims to address this critical gap.

3 Objectives

This research aims to evaluate the feasibility of frequency-based voice recognition for both speaker identification and speech content analysis. By leveraging spectral analysis techniques such as Fourier Transform and Mel-Frequency Cepstral Coefficients (MFCCs), combined with deep learning classifiers, we seek to develop a system that enhances recognition accuracy and resilience. Additionally, this study will explore the implementation of a real-time keyword detection module to identify suspicious words in spoken language, offering potential applications in security-sensitive environments.

The primary objectives of this study include:

1. Investigating the effectiveness of frequency-based features for voice recognition.
2. Comparing the performance of spectral analysis and deep learning techniques against traditional speaker recognition methods.
3. Developing a classification model using spectral features and deep learning to improve speaker identification accuracy.
4. Implementing a keyword detection system capable of identifying suspicious words in real-time speech analysis.
5. Evaluating the system's robustness against environmental noise, voice modifications, and spoofing attempts.

By addressing both speaker identification and speech content analysis, this research contributes to advancing voice recognition technology for applications in law enforcement, cybersecurity, and automated monitoring systems. Future developments will focus on improving detection accuracy, optimizing computational efficiency, and expanding the dataset to enhance real-world applicability. Additionally, ethical considerations must be prioritized to ensure responsible use of voice recognition systems in sensitive domains.

The findings of this study will pave the way for more advanced voice recognition models that integrate real-time keyword detection, helping to bridge the gap between biometric security and speech content analysis. With continued innovation and responsible

implementation, frequency-based voice recognition could revolutionize the way voice authentication and security monitoring systems operate, making them more secure, efficient, and adaptable to a wide range of environments.

4 Automatic keyword detection and crime prediction

This section outlines the methods and techniques employed to develop a system for automatic keyword detection and crime prediction based on phone call audio analysis. The methodology consists of several stages: data acquisition, preprocessing, feature extraction, keyword classification, and crime prediction using machine learning algorithms.

4.1 System Overview

The proposed system processes audio calls to detect specific keywords indicative of criminal activity and predicts the potential crime category. The architecture comprises five main modules: audio input, preprocessing, feature extraction, keyword detection, and crime classification.

4.2 Data Collection

Audio data were sourced from publicly available speech datasets, recorded phone call samples, containing criminal vocabulary. The dataset was manually annotated to label keywords associated with various crime types, such as robbery, bribe , and drugs .

4.3 Preprocessing

Preprocessing involves cleaning and preparing audio signals for analysis by applying noise reduction filters, removing silence segments, segmenting the audio into frames, and normalizing amplitude to ensure consistent input to the models.

4.4 Feature Extraction

Acoustic features were extracted using digital signal processing techniques, focusing on parameters that capture the distinctive properties of speech. The primary features include Mel-Frequency Cepstral Coefficients (MFCCs), spectral centroid, zero-crossing rate, pitch, and energy. These features enable effective discrimination between spoken keywords.

4.5 Acoustic Signal Parameterization

Speech signals exhibit complex, time-varying acoustic structures that carry vital information for keyword detection. Key parameters analyzed include:

- Fundamental Frequency (F0): Reflects vocal fold vibration rate and varies by speaker group (e.g., 80–200 Hz for adult males).
- Spectral Content (Timbre): Frequency distribution crucial for identifying vocal qualities and distinguishing keywords.
- Energy: Indicates speech intensity, with voiced segments typically exhibiting higher energy than background noise.
- Spectrogram Analysis: Visualizes frequency content over time, facilitating pattern recognition for specific keywords.
- Amplitude Variations: Changes in loudness can signal emotional emphasis or suspicious speech cues.

Incorporating these parameters enriches feature extraction, enhancing keyword detection accuracy and crime classification.

4.6 Keyword Detection Using convolutional Neural Networks

A convolutional neural network (CNN) was designed to classify audio segments containing suspicious keywords. The network architecture consists of multiple convolutional layers followed by fully connected layers, with the final output layer corresponding to the number of keyword classes.

The CNN input is a time-frequency representation of the audio frames, and the network includes:

- Three convolutional layers with 42 and 60 filters respectively, combined with batch normalization, leaky ReLU activations, and max-pooling layers for effective feature extraction.
- Three fully connected layers, each with 256 neurons, to learn complex representations before the final classification layer.
- A softmax output layer and classification loss function for multi-class classification.

The dataset was split into training and testing sets with an 80/20 ratio, where the test data was also used for validation during training. The network was trained for 5 epochs

using the Adam optimizer with a mini-batch size of 128. Training progress was monitored through validation accuracy and loss.

This CNN model achieved rapid convergence, effectively learning to detect keywords associated with crime-related speech patterns, thereby forming a crucial part of the overall crime prediction system.

4.7 Crime Prediction Model

Recognized keywords were mapped to potential crime categories using either a rule-based system or a supervised machine learning classifier. Each keyword was associated with probability scores for different crime types, and the final classification decision considered the presence, frequency, and contextual relationships of multiple keywords within a call.

4.8 Evaluation Metrics

System performance was assessed using multiple metrics: Mean Squared Error (MSE), confusion matrices, recognition accuracy, and precision, recall, and F1-score for crime classification.

5 Methods and materials

5.1 Tools and Environment

- MATLAB: Used for neural network training and audio processing.
- Praat: Utilized for manual audio inspection and segmentation.

5.2 Data Collection

To build an efficient crime prediction and voice identification system, we selected specific words related to crimes such as theft, bribery, and drug-related offenses. Each chosen word was incorporated into three different sentences: one where the word appears at the beginning, one where it appears in the middle, and one where it appears at the end. Additionally, we included non-crime-related sentences to train the system with neutral speech data, ensuring that it does not generate false alarms. The sentences were all in Algerian slang, as the system is designed to be effective in our linguistic and cultural context.

The following table summarizes the crime types, keywords, and examples collected for training and evaluation."

Table 1: Keywords and examples for Drugs crime (مخدرات)

Keyword	Example Sentences	Note
الحلوة	- الحلوة الي قتلك عليها حطهالي في القهوة كيما موالفين . - شكون لي يبيع الحلوة المخيرة في الشارع هذا - اسمع راه جابلي الدراهم كي يجيك غدوة أعطيلو الحلوة	Code word for drugs (sweet)
الحشيش	- الحشيش لي دخل لبارح للكارتي مراحش تلقاه في بلاصة خلاف - ضركا هو كي دخل كونتيتي نتاع حشيش علابالك بلي رايج يطيح علينا السوق - اليوم نشوفلك في الليل جاري مزال عندو شويا حشيش	Direct mention of "Hashish"
الكاشي	-الكاشي نتاع لبارح ناقصة شوية كشما كاين الجديد اليوم -راها دخلت وحد الكاشي متتعاودش وماراحش تلقاها بهذي سومة - اي ماشي عقلية نتا تحوس يفيقو بينا لي يجي ليك تبيعلو الكاشي	Hidden term for drugs
الشيكولة	- الشيكولة لي راه ي ثور هنا غالي وناقصة تسمى كون ندخلو سلعتنا نطغاو في السوق - عندي 50الف ملخر اذا تريقلي نديها عليك شيكولة - واش موح جبتلي الشيكولة لي وصيتك عليها	Slang term for drugs

Table 2: Keywords and examples for Theft crime (السرقة)

Keyword	Example Sentences	Note
كاس	- كاسيتلو الخزنة ولا مزال - البونكة لي زعما راح نكاسيوها ركبولها كاميرات تسمى كيفاش ضركا نشوفو حاجة خلاف - هيا نتا كلانا الفقر شوفلنا كاش دار ولا حانوت نكاسيوه	Refers to robbing (stealing)
السرقة	- نسرقولو الحانوت في اليل كي يخرج من دار صباح كي تجي الدولة ما يلقاو والو - شوف نخلوه حتى يخرج من الدار ندخلو نسرقولو غير بالحاجة باش ما يفيقش - ماتقلقش حنا نلعبوها نعاونوه ندخلو معاه القش ونحفظو الدخلات والخرجات باش من بعد تجي ساهلة السرقة	Clear plan to commit theft

Table 3: Keywords and examples for Bribery crime (الرشوة)

Keyword	Example Sentences	Note
قهوة	- قهوة وحنا فاريينها معاه السيد هذاك طماع - قالولي اعطيه قهوة برك و يحكم عليك الدوصي - شوف لعزيز الشغل لي حكينا فيه راه مفري بصح السيد طلب فيا قهوة	"Coffee" as bribe
مصروف	- مصروف هو الحل باش ياش نفريلك كواغطك بلخف - مانعيي روحي ما نستنا لاشان هذا كامل نديلو شويا مصروف ويفريهالي ليه ليه - باش تدي البوست هذاك لازم عليك تعطيلهم مصروف	Bribery via "expenses"

Slang for paying bribes	- تشيبا ويخليك تفوت بلا ما يفتحلك الكابة - الناس كامل علابالهم بلي التشيبا هي الحل باش تفريها معاه - اذا تحوس تنجح في الكونكور بلا تكسار راس غير اعطيلهم تشيبا	تشيبا
Bribery hidden as "gift for kids"	- حاجة للذراري برك مانكثرش عليك - اذا ما قبلش اديلو حاجة للذراري برك يحشم منك ويفريلك - الحل الاخير نروحلو للدار وناديلو معايا حاجة للذراري	حاجة للذراري

Table 4: Examples for Neutral Sentences (جمل عادية)

Example Sentences	Note
- ما نقدرش نفوت نهاري بلا ما نشرب كأس تاي مع العشية. - البارح كان راسي يوجع فيا ، ما قدرتش نرقد مليح. - نهار الجمعة نحب نقعد في الدار ونقرا شوية قرآن. - البارح خرجت نجري شوية في الغابة باش نحافظ على صحتي. - حبيت نسافر هذا الصيف نشوف بلاد جديدة ونتعلم لغة جديدة. - نهار الأحد عندي اجتماع في الخدمة، لازم نوض صباح بكري.	No crime — training data

To gather voice data, we asked individuals from different age groups and both genders to record these sentences in WAV format. The participants were categorized into three age groups:

- Youth (A)
- Middle-aged (B)
- Older than 40 (C)

This diversity ensures the system can adapt to different voices, accents, and pronunciation variations in real-world applications.

5.2.1 Data Processing and Organization

Following the collection of voice recordings, the data was systematically organized by categorizing samples according to their associated crime-related keywords. For each keyword, three distinct sentences were recorded and stored within a structured dataset.

A consistent naming convention was applied to all recordings to facilitate easy identification and management:

- The **first letter** indicates the speaker's gender (M for male, F for female).
- The **second letter** corresponds to the age group (A, B, or C).
- The **next two letters** represent the speaker's initials (first and last name).
- A **numerical code** designates the specific keyword and the sentence position within that keyword group.

This structured labeling approach ensured orderly dataset maintenance and streamlined subsequent processing and analysis.

The preprocessing stage was critical for preparing raw audio signals for reliable feature extraction and classification. It comprised the following steps:

- **Noise Reduction:** A bandpass filter with a frequency range of 300 Hz to 3400 Hz was applied to suppress background noise while preserving speech frequencies common in telephone audio.
- **Silence Removal:** Non-informative silent segments were eliminated to focus the analysis on active speech content.
- **Segmentation:** Audio signals were segmented into short frames optimized for feature extraction.
- **Normalization:** Signal amplitudes were normalized to a uniform range to minimize the impact of volume variations across recordings.
- **Sampling Rate Adjustment:** All audio samples were resampled to 16 kHz, a standard rate balancing speech quality and computational efficiency.

These preprocessing steps ensured that the input audio signals were clean, consistent, and well-prepared for robust acoustic feature extraction and subsequent model training.

5.2.2 Acoustic Feature Extraction

In this stage, the preprocessed audio signals are transformed into representative features that effectively capture both the spectral and temporal properties of speech. These features serve as the foundation for accurate keyword detection and classification.

5.2.2.1 Speech signals preprocessing

For precise analysis, voice signals were segmented using *Praat*, a widely used speech analysis tool. This segmentation involved carefully identifying the exact start and end times of the target keywords within each recording. Accurate segmentation is critical to ensure that the features extracted correspond strictly to the intended keywords, thereby enhancing the reliability and accuracy of the keyword detection model.

Following segmentation, Mel-Frequency Cepstral Coefficients (MFCCs) were extracted from each segment. MFCCs effectively capture the spectral envelope of speech in a way

that aligns with human auditory perception, making them a powerful feature for speech and speaker recognition tasks.

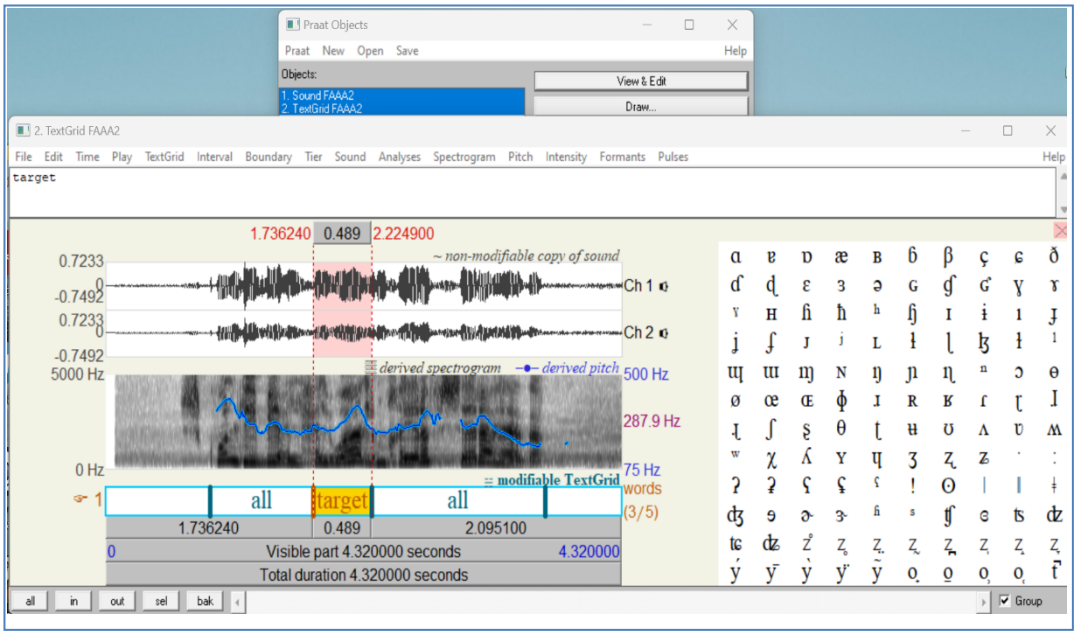


Figure 1. Segmentation of speech signals using Praat to detect keyword positions.

Following segmentation, we extracted key frequency-based features from the speech signals using Fourier Transform and Mel-Frequency Cepstral Coefficients (MFCCs). These features serve as input for the machine learning model, allowing it to differentiate between speakers and detect suspicious words effectively.

MFCCs are widely used to represent the human auditory system's perception of sound. The extraction process involves several steps:

- **Framing and Windowing**: Speech signals were divided into frames of 25 milliseconds with 50% overlap, and a Hanning window was applied to each frame to minimize edge effects.
- **Short-Time Fourier Transform (STFT)**: The Fourier Transform was applied to each windowed frame to obtain its frequency spectrum, providing a time-frequency representation (spectrogram) of the signal.
- **Mel-Scale Transformation**: The frequency axis was warped using a bank of triangular Mel filter banks. The Mel scale reflects the human ear's nonlinear sensitivity to frequencies, calculated by the formula:

$$M(f) = 2595\, log_{10}\left(1 + \frac{f}{700}\right)$$

- **Logarithmic Compression and Discrete Cosine Transform (DCT)**: A logarithm was applied to the filterbank energies, and then the DCT was used to decorrelate the coefficients, resulting in a compact feature set.

From each frame, 13 MFCC coefficients plus 1 log-energy coefficient were extracted. Additionally, first-order delta (Δ) and second-order delta-delta (ΔΔ) coefficients were computed to capture the dynamic changes over time. Thus, the final feature vector per frame consisted of 42 dimensions (MFCC + Δ + ΔΔ).

These features capture both the static spectral characteristics and the temporal dynamics of speech, making them ideal for tasks such as keyword detection and speaker identification [1].

5.2.2.2 Extracting MFCC parameters

To effectively utilize Convolutional Neural Networks (CNNs) for audio classification, it is essential to transform the one-dimensional waveform into a structured representation that retains both spectral and temporal characteristics. Raw audio signals are not naturally suited for CNNs, as they lack the spatial features typically found in images. Therefore, feature extraction techniques are employed to represent the signal in a format compatible with deep learning models.

Among these techniques, the Short-Time Fourier Transform (STFT**)** provides a time-frequency representation of the signal, while Mel-Frequency Cepstral Coefficients (MFCCs**)** offer a compact and perceptually relevant feature set for speech and audio processing. This section provides an overview of these methods and their mathematical formulations.

The extraction of parameters from a speech signal is done by transforming the latter into a sequence of acoustic vectors. This form is much more suitable for statistical and vector modeling. Speaker recognition is applied to speech signals represented in spectral form. For this reason, the pre-accented speech signal can be considered as a quasi-stationary signal after analyzing it by a window such as the Hamming window with a short duration of the order of 25ms:

$$h(n) = \begin{cases} 0.54 - 0.46\cos\dfrac{2\pi n}{N-1} & 0 < n < N-1 \\ 0 & elsewhere \end{cases} \qquad (01)$$

To make the spectral representation of the speech signal, we apply the Fourier transform on each frame of the signal obtained after windowing, in general we use algorithms such as FFT (Fast Fourier Transform) [2].

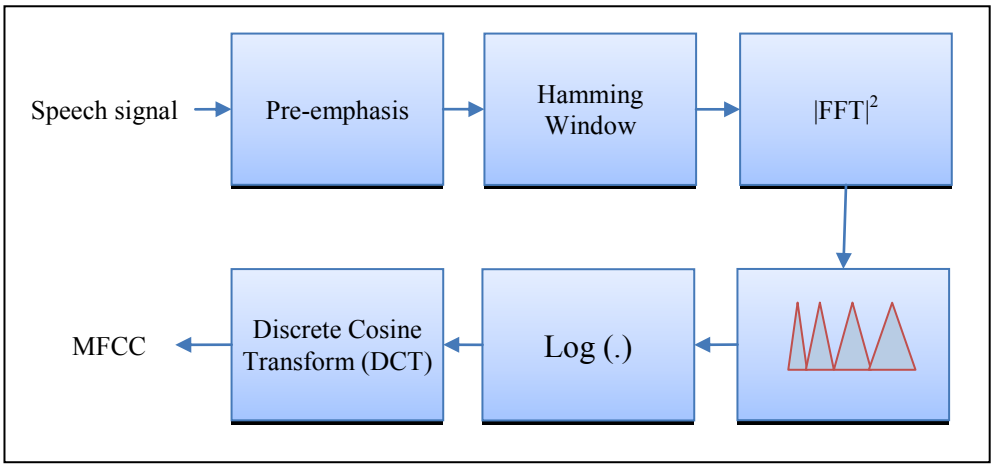


Figure 2. Complete process of extracting Mel-Frequency Cepstral Coefficients from raw speech.

The spectrum obtained contains several fluctuations, but we are only interested in the envelope of the spectrum. Also, to reduce the size of the spectral vectors, the spectrum of the signal must be smoothed; to eliminate the fluctuations and make it smooth, it must be multiplied by a filter bank (a series of filters with equidistant bandwidth in the Mel scale). This filter bank is defined according to the shape of each filter that composes it, and the location of its frequencies (center, right or left), these filters are often triangular in shape. The location of the central frequencies of the filters is given by:

$$f_{mel} = 2595\ log_{10}\left(1 + \frac{f}{700}\right) \quad (02)$$

With: f the frequency in Hz

Finally, the spectral envelope in dB is calculated from the logarithm of this envelope, then a discrete cosine transform is applied to obtain the cepstral coefficients according to logarithms of the energies from the filter bank. The mathematical calculation of the coefficients is defined by the following expression:

$$C_n = \sum_{j-1}^{K} S_j cos\left[(j - 0.5)\frac{n.\pi}{K}\right] \quad (03)$$

With:

- $i = 1,2, \ldots . L$;
- K: number of spectral coefficients calculated previously $(k = 23)$;
- S_j : spectral coefficients;
- L: number of cepstral coefficients that we want to calculate $(L = 12)$.

The MFCCs computed so far represent static spectral features and do not capture temporal dynamics. To incorporate the time evolution of speech, cepstral derivatives (delta and delta-delta coefficients) are calculated, which provide dynamic information useful for distinguishing speakers and speech patterns [3][4].

The first derivatives are the Δ coefficients, which represent the speed of change of the vectors over time. The second derivatives are the ΔΔ coefficients, which represent the acceleration of speech.

These coefficients are expressed by [5]:

$$\Delta C_m = \frac{\sum_{p=-1}^{L} p^2 C_{m+p}}{\sum_{p=-1}^{L} |p|} \tag{04}$$

$$\Delta\Delta C_m = \frac{\sum_{p=-1}^{L} p^2 C_{m+p}}{\sum_{p=-1}^{L} |p|^2} \tag{05}$$

Generally, the number of coefficients is taken as 13, and sometimes reduced to 12, considering two essential points:

1. The first coefficient represents the energy of the frame and cannot really contribute to recognition.
2. The 12 coefficients represent the more or less smoothed cepstral envelope, with suppression of high frequency variations.

5.2.2.3 Spectrograms

In addition to Mel-Frequency Cepstral Coefficients (MFCCs), spectrograms were also extracted to provide a complementary time-frequency representation of the speech signals. These spectrograms were generated using the Short-Time Fourier Transform (STFT), which converts the one-dimensional time-domain signal into a two-dimensional representation where the horizontal axis denotes time, the vertical axis denotes frequency,

and the intensity of each point represents the magnitude of the frequency component at a given time.

To compute the spectrograms, the audio signals were first segmented into overlapping frames of 25 milliseconds with a 50% overlap between consecutive frames. A Hanning window was applied to each frame to reduce spectral leakage. The STFT was then applied to obtain the magnitude spectrum of each frame, and the resulting spectrograms were converted to decibel (dB) scale for improved visibility and interpretability.

These spectrograms preserve both the temporal and spectral dynamics of speech, making them particularly suitable for Convolutional Neural Network (CNN) models. Unlike MFCCs, which summarize the spectral content using filter banks and DCT, spectrograms maintain a more detailed, fine-grained view of the signal, allowing the CNN to learn subtle frequency patterns that may correspond to specific keywords or speaker characteristics.

The spectrograms, along with MFCCs and their delta coefficients, formed the set of features used as input for the neural network models developed for keyword detection and speaker identification tasks.

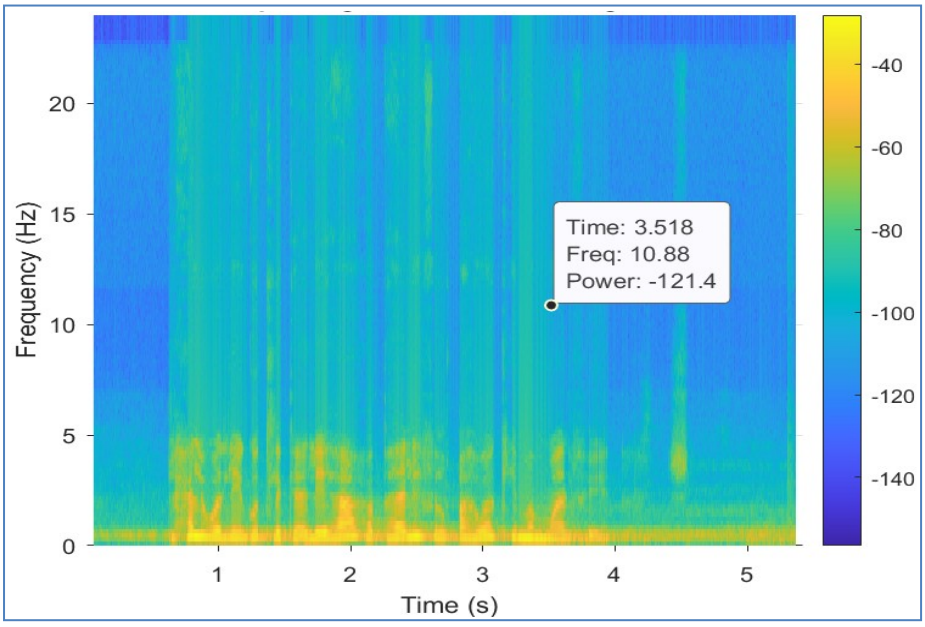


Figure 3. The Spectrogram of an Audio signal for CNN input.

5.2.2.4 Short-Time Fourier Transform (STFT)

The classical Fourier Transform (FT) provides a global frequency representation of a signal but lacks the ability to track how frequency content evolves over time. This limitation makes it unsuitable for analyzing non-stationary signals such as speech, where frequency components change rapidly.

To overcome this, the Short-Time Fourier Transform (STFT) is employed. STFT allows time-localized frequency analysis by segmenting the signal into short overlapping frames and then applying the Fourier Transform to each frame independently. This process results in a two-dimensional representation of the signal—known as a spectrogram—which displays how the frequency spectrum of the signal varies with time.

1. **Framing:** The signal is divided into short time frames, typically 20–40 milliseconds long. In our case, a 25 ms frame with a 50% overlap was used.
2. **Windowing:** A window function (e.g., Hanning or Hamming window) is applied to each frame to minimize edge discontinuities and reduce spectral leakage.
3. **FFT Application:** The Fast Fourier Transform (FFT) is then applied to each windowed frame to obtain the frequency content.
4. **Spectrogram Construction:** The magnitude (and optionally, phase) of the FFT outputs are plotted over time to form the spectrogram.

The STFT of a discrete-time signal $x(n)$ is defined as:

$$X(m,k) = \sum_{n=-\infty}^{\infty} x(n)\, w(n - mR) e^{-j 2\pi k n / N} \tag{06}$$

where:

- $X(m,k)$: represents the time-frequency representation of the signal,
- $w(n)$: is a window function, commonly chosen as a **Hanning window** to reduce spectral leakage,
- R: is the hop size (the step between successive frames),
- N: is the number of frequency bins, and
- k: denotes the frequency index.

Applying the STFT results in a **spectrogram**, a two-dimensional representation where the x-axis represents time, the y-axis represents frequency, and the intensity represents

signal amplitude. However, the spectrogram does not align well with human auditory perception, leading to the use of the **Mel scale** for improved feature representation [6].

5.3 Keyword Detection Techniques

Keyword spotting was tackled through two main approaches: a classical garbage model method and a modern deep learning-based technique.

5.3.1 Classical Garbage Model Approach

The classical garbage model approach is a traditional keyword spotting technique that distinguishes relevant keywords from irrelevant or random speech inputs (referred to as *garbage words*). The goal is to minimize false detections by training the system to recognize and discard non-target words.

5.3.1.1 Garbage Model implementation

The implementation of a Garbage Model is maked from 4 steps:

- **Step 1: Collect Speech Data**
    - Target Words → Words related to crime (e.g., theft, bribery, drugs).
    - Garbage Words → Random, everyday words (e.g., "table", "good morning", "hello", "football").
- **Step 2: Label and Categorize Data**

Each recorded sentence is labeled as:

- Keyword Class → If it contains a crime-related word.
- Garbage Class → If it contains irrelevant words.

- **Step 3: Feature Extraction**

After collecting and labeling the speech data, we extract features that represent the unique characteristics of each word. This is typically done using:

- Mel-Frequency Cepstral Coefficients (MFCCs)
- Linear Predictive Cepstral Coefficients (LPCCs)
- Spectrogram Analysis

These features allow the model to differentiate between meaningful keywords and irrelevant words based on their frequency and spectral properties.

- **Step 4: Train a Classifier**

We train a machine learning model (such as Hidden Markov Models (HMMs) or Neural Networks) to classify words into two categories:

  - Target Words (Keywords) – The system should recognize these.
  - Garbage Words – The system should ignore these to reduce false detections.

The Garbage Model is trained to predict when a spoken word does not belong to the target keyword list. If the model detects a word and is uncertain whether it is a keyword, it compares it with the garbage model. If the probability of it being garbage is higher, it is ignored.

5.3.1.2 Limitations of the Garbage Method

- Requires a large dataset of garbage words to be effective.
- Struggles with unseen words, leading to misclassifications.
- Higher false positives, as some garbage words may resemble target words.
- Less adaptable compared to modern deep learning techniques.

5.4 CNN-Based Deep Learning Approach for Keyword Detection

To overcome the limitations of classical methods such as Hidden Markov Models (HMMs) and the Garbage Model approach, a deep learning-based solution was implemented using Convolutional Neural Networks (CNNs).

CNNs are highly effective in keyword detection tasks due to their ability to automatically learn hierarchical representations from speech data. Unlike traditional methods that rely on handcrafted features, CNNs process spectrogram-based representations such as Mel-Frequency Cepstral Coefficients (MFCCs) to extract both spatial and temporal features critical for recognizing keywords.

5.4.1 CNN-based keyword detection implementation

Figure 4 illustrates a general block diagram of the convolutional neural network used for keyword detection.The CNN-based keyword detection system follows a structured pipeline consisting of three main stages:

5.4.1.1 Preprocessing and Feature Preparation

Speech recordings containing both target crime-related keywords and neutral non-keywords were collected. Each sample was processed to extract MFCC features, creating a two-dimensional time-frequency representation suitable as input to the CNN. This representation is similar to images in computer vision tasks, enabling effective pattern recognition.

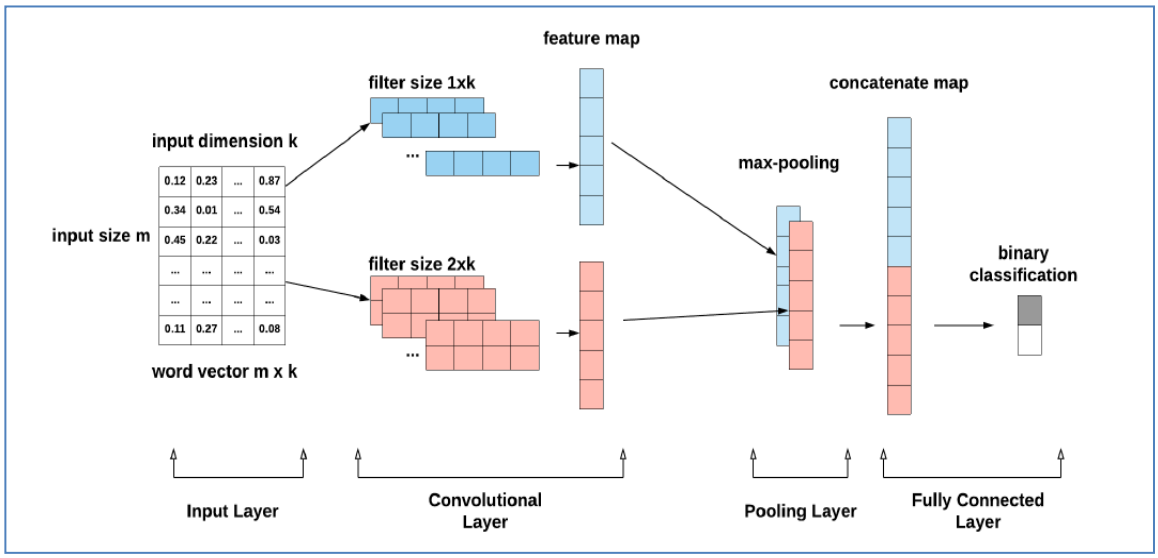


Figure 4. General block diagram of the convolutional neural network used for keyword classification.

5.4.1.2 Feature Extraction Using CNN Layers

Once MFCC features were generated, they were processed through a CNN architecture composed of:

- **Convolutional Layers**: Learn local time-frequency patterns from speech data, detecting important phonetic and spectral structures.
- **Pooling Layers**: Downsample feature maps to reduce dimensionality and increase model robustness to variations such as speaker differences and background noise.
- **Activation Functions**: ReLU (Rectified Linear Unit) activations were used to introduce non-linearity and allow the model to learn complex feature interactions.

5.4.1.3 Classification Stage

After feature extraction, the CNN model classified input samples using:

- **Fully Connected Layers**: These layers combined the extracted features into a global understanding of the input.
- **Softmax Output Layer**: Provided probability scores for each class, predicting whether the input contained a suspicious keyword or not.

5.4.2 Advantages of the CNN Approach

The CNN-based method demonstrated several advantages over classical keyword detection approaches:

- **Superior Noise Robustness**: CNNs maintained high detection accuracy even in noisy environments.
- **Higher Accuracy**: Improved recognition rates compared to traditional methods.
- **Generalization**: The model was able to generalize to unseen speakers and new speech samples effectively.

Thus, the CNN-based deep learning approach provided significant improvements in detection accuracy and real-world applicability for crime-related keyword spotting.

5.4.3 Comparative study of clssical and CNN-Based Keyword Detection Techniques

CNNs offer several advantages over traditional HMM-GMM approaches in keyword spotting tasks:

Table 5: Comparative Analysis of Traditional and CNN-Based Keyword Detection Techniques

Feature	Classical Methods (HMM-GMM)	CNN-Based Approach
Feature Extraction	Manual (handcrafted)	Automated (learned)
Noise Robustness	Low	High
Real-Time Processing	Moderate	Fast
Accuracy	Moderate	High

CNNs significantly improve accuracy and robustness, especially in noisy environments, making them ideal for crime-related keyword spotting applications. [Tang & Lin, 2018]

5.5 Feature Representation for CNN Processing

To leverage the capabilities of Convolutional Neural Networks (CNNs) for audio analysis, it is essential to convert one-dimensional raw waveforms into structured two-

dimensional representations suitable for feature extraction. Unlike images, raw audio signals lack a natural spatial structure; therefore, a time-frequency transformation is necessary to capture both spectral and temporal characteristics of the signal.

Among various feature extraction techniques, Mel-Frequency Cepstral Coefficients **(**MFCCs**)** were selected due to their strong correlation with human auditory perception. MFCCs provide a compact and informative representation of the signal by emphasizing perceptually relevant frequency components. To capture dynamic variations, delta (first-order derivative) and delta-delta (second-order derivative) coefficients were also incorporated, enhancing the robustness of the extracted features.

5.5.1 Feature Extraction Process

To effectively analyze and classify audio signals using a Convolutional Neural Network (CNN), meaningful features that capture both spectral and temporal information must be extracted. The process involves:

- **Applying Short-Time Fourier Transform (STFT)**: Converts the one-dimensional audio signal into a two-dimensional spectrogram.
- **Mel-Scale Conversion**: Applies triangular filter banks to map frequencies to the Mel scale, emphasizing human-relevant frequency regions.
- **Computing MFCCs**: Extracts cepstral coefficients from the Mel-scaled spectrogram, capturing key features of human speech.
- **Calculating Delta and Delta-Delta Coefficients**: Captures the dynamic behavior of speech features over time.
- **Visualization**: Optionally, the extracted features can be visualized to verify the time-frequency structure before feeding into the CNN model.

This feature extraction process ensures that the input to the CNN is rich in both spectral and temporal information, enabling effective keyword detection.

5.5.1.1 Preprocessing of the Audio Signal

Before feature extraction, the input audio signal underwent a preprocessing phase to ensure consistency and enhance the quality of the extracted features. This step involved two main operations:

- **Normalization**: The amplitude of the signal was normalized by dividing each sample by the maximum absolute amplitude. This process ensures that variations in loudness do not affect the extracted features and prevents amplitude-dependent distortions.
- **Silence Removal**: Silent and near-zero amplitude segments were removed to focus on the informative portions of the signal. This step reduces computational complexity and enhances the robustness of the extracted features.

Following preprocessing, the audio signal was ready for spectral transformation using the Short-Time Fourier Transform (STFT) as part of the MFCC computation process.

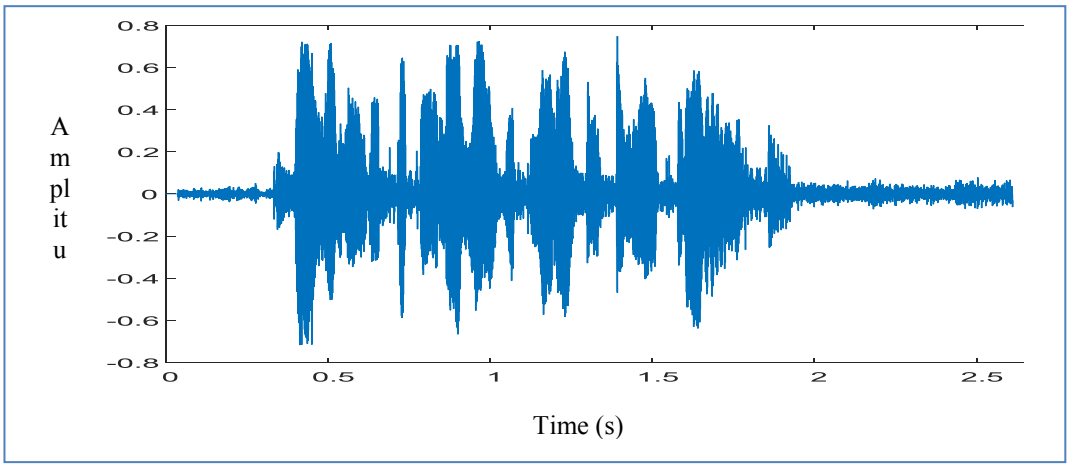


Figure 5. Speech signal: normalization and silence removal for MFCC feature extraction.

The time-domain plot of the input audio signal after preprocessing. The waveform has been normalized, and silence has been removed to enhance the feature extraction process

5.5.1.2 Time-Domain Visualization of the Preprocessed Signal

After applying normalization and silence removal during preprocessing, the time-domain waveform of the input audio signal was visualized. This ensures that only the informative speech segments were preserved, enhancing the quality of the features to be extracted.

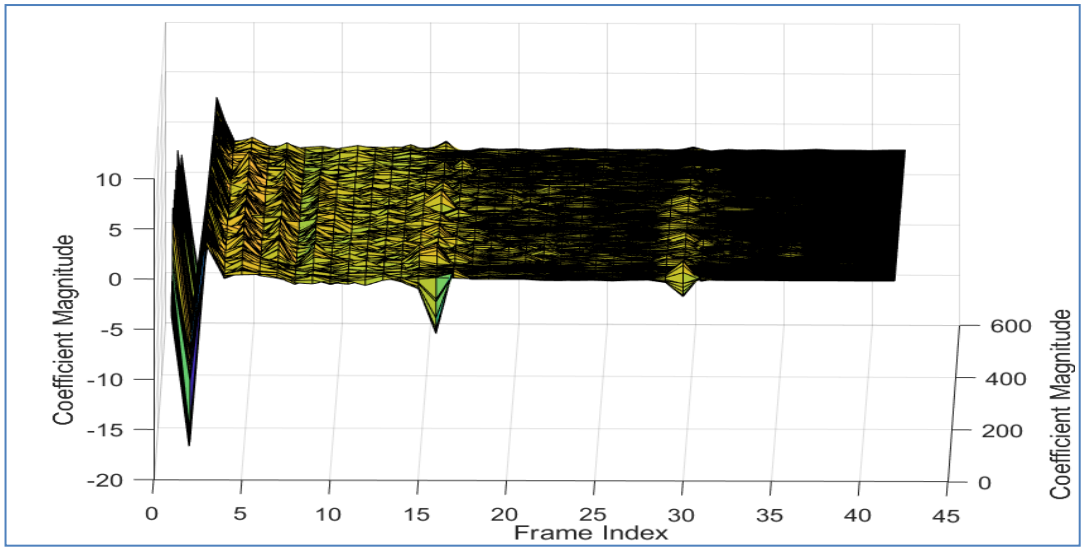


Figure 6. Visualization of MFCC and their derivatives for one audio segment, showing time-frequency dynamics.

To better understand the extracted features, graphical representations were generated. A 2D plot was used to analyze the variation of MFCC coefficients over time, providing insights into the changing spectral properties of the signal. Additionally, a 3D surface plot was constructed, offering a more detailed visualization of the temporal and spectral structure of the extracted features. These visualizations confirmed that the MFCC-based representation effectively captures the relevant characteristics of the signal for further processing.

A3D surface plot offered a more detailed view of the time-frequency structure of the extracted features

5.5.2 Preparing Features for CNN Input

With the MFCC features extracted and structured, the next step was preparing them into a suitable input representation for the CNN model. Since CNNs are designed to process two-dimensional structured data, the extracted MFCC matrices were treated as image-like inputs. The combination of static coefficients, delta coefficients, and delta-delta coefficients ensured that the network could learn both spectral and temporal patterns from the input data.

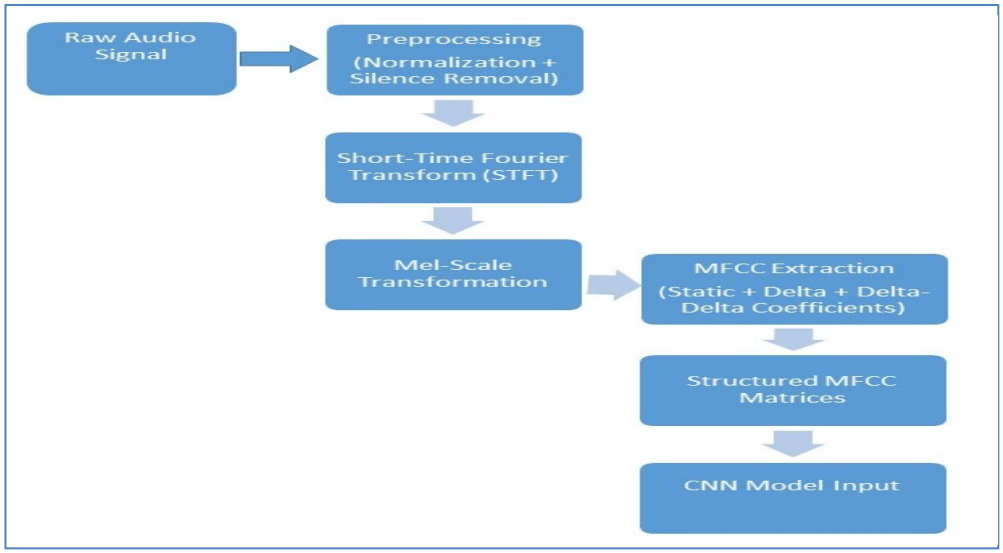
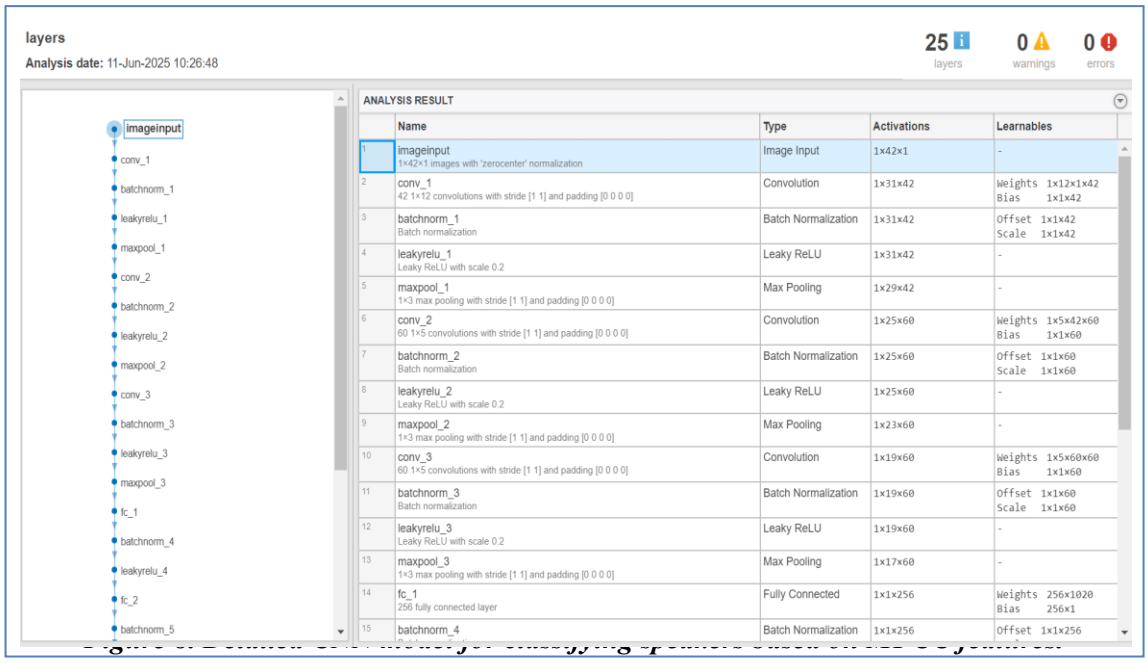


Figure 7. Overview of the preprocessing and feature extraction steps used to train the CNN.

5.6 Speaker Identification Method

To identify speakers, a CNN-based speaker recognition system was developed using Mel-Frequency Cepstral Coefficients (MFCCs) along with their Delta and Delta-Delta coefficients as input features. The system architecture is designed as a supervised classification model that directly classifies speakers from speech frames.

- **Feature Extraction:** MFCCs with Delta and Delta-Delta coefficients were extracted from each speech frame to capture both spectral and temporal dynamics of the speaker's voice.
- **CNN Architecture:** The extracted features were fed into a Convolutional Neural Network composed of convolutional layers for local pattern extraction, max-pooling layers for downsampling, and fully connected dense layers for classification.



Output Layer: A fully connected layer with Softmax activation was used to classify input speech frames into one of the known speaker classes.

This approach enables the system to associate detected keywords with specific speakers, enhancing the overall security and identification capability of the application.

5.6.1 Model Training and Evaluation Metrics

5.6.1.1 Training Process

- **Data Splitting:** The dataset was split into 70% training, 15% validation, and 15% testing.
- **Optimizer:** Adam optimizer was used with categorical cross-entropy loss function.
- **Learning Rate:** Initialized at 0.001 with adaptive decay during training.
- **Regularization:** Dropout layers were applied to prevent overfitting.
- **Early Stopping:** Training was monitored using validation loss to save the best-performing model and avoid overfitting.

5.6.1.2 Evaluation Metrics

- **Accuracy:** Percentage of correctly classified speaker samples.

- **Confusion Matrix:** Used to visualize classification performance across different speakers.
- **Precision, Recall, and F1-Score:** Computed to evaluate classification robustness, especially important in cases of class imbalance.

Note: Unlike verification-based speaker recognition systems, this model directly performs classification without embedding generation or cosine similarity measures.

6 Results and Discussion

6.1 Keyword Detection Model

The CNN-based model for keyword detection was trained using MFCC features with delta and delta-delta derivatives. The model was trained for 4 epochs over 832 iterations using the Adam optimizer and a constant learning rate of 0.001.

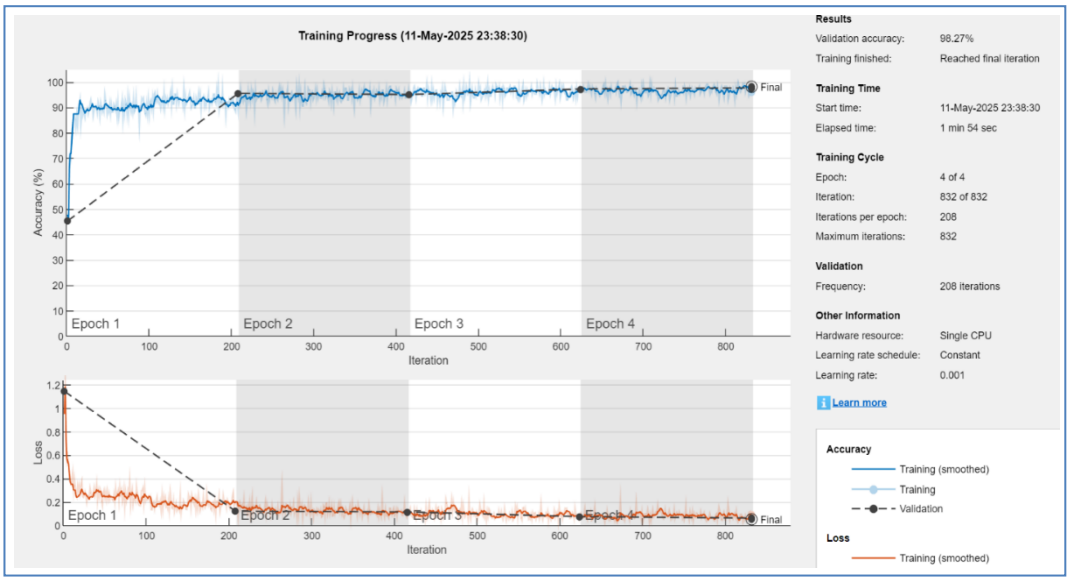


Figure 9. Training progress for Keyword Detection – Drug Contexts.

The final validation accuracy achieved was **98.27%**. The training process showed rapid convergence in the first epoch, and accuracy stabilized above 95% in the following epochs. The loss curve also dropped steadily, indicating proper learning without overfitting.

This figure illustrates the CNN training performance for detecting drug-related keywords. The model shows rapid convergence and high accuracy by the final epoch, demonstrating effective feature learning for the keyword *الحشيش*.

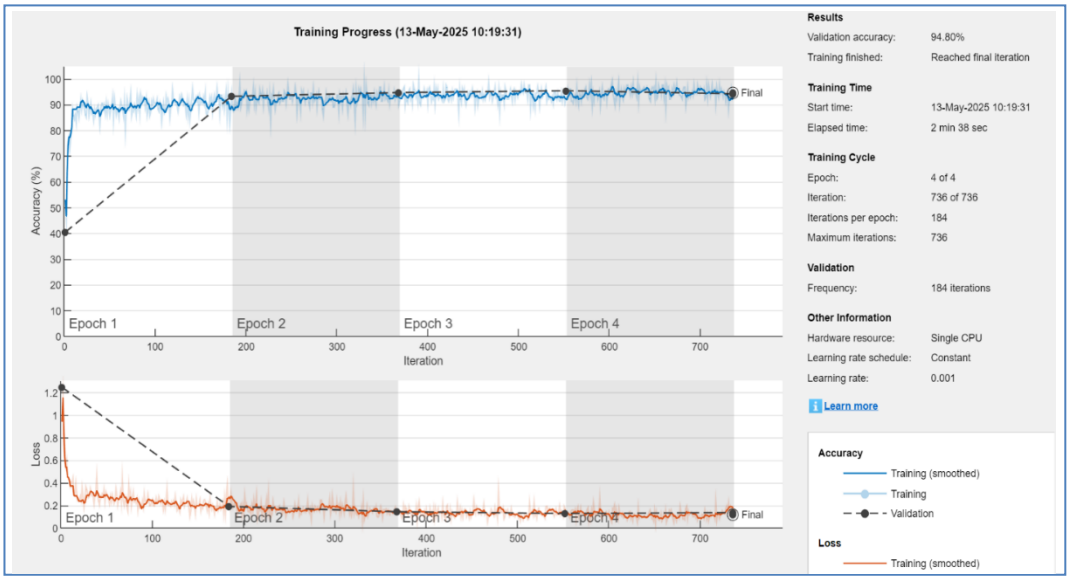


Figure 10. Training progress for Keyword Detection – Theft Context.

CNN training accuracy over epochs for theft-related keyword السرقة. The model maintains consistent learning and achieves high validation accuracy, confirming its effectiveness in recognizing theft-associated speech patterns.

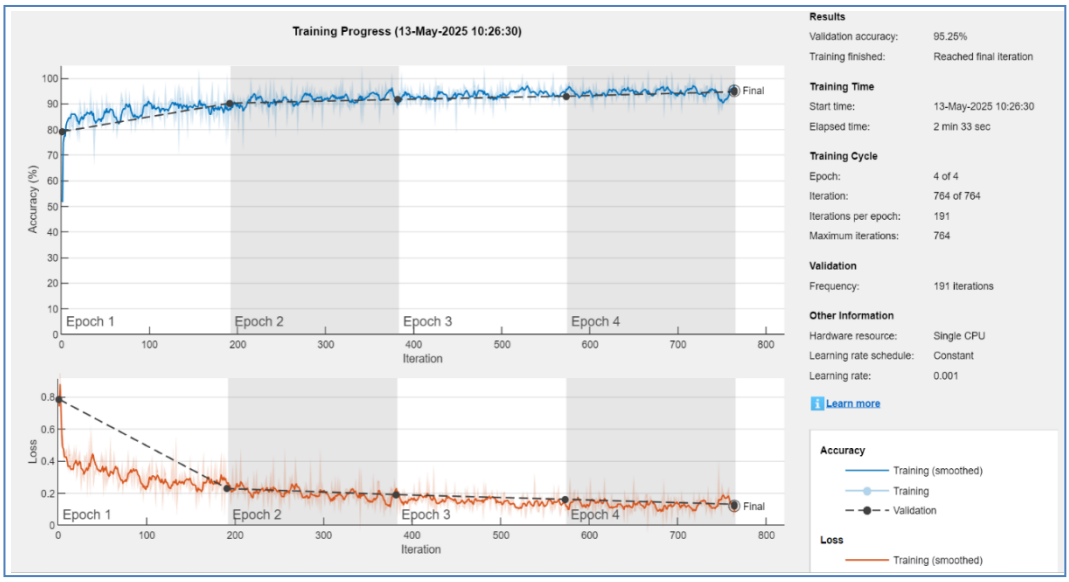


Figure 11. Training progress for Keyword Detection – Bribery Context.

Validation accuracy progression for detecting bribery-related keyword تشيبا. The model exhibits strong learning behavior with stable improvements and minimal overfitting, indicating robust generalization for this category.

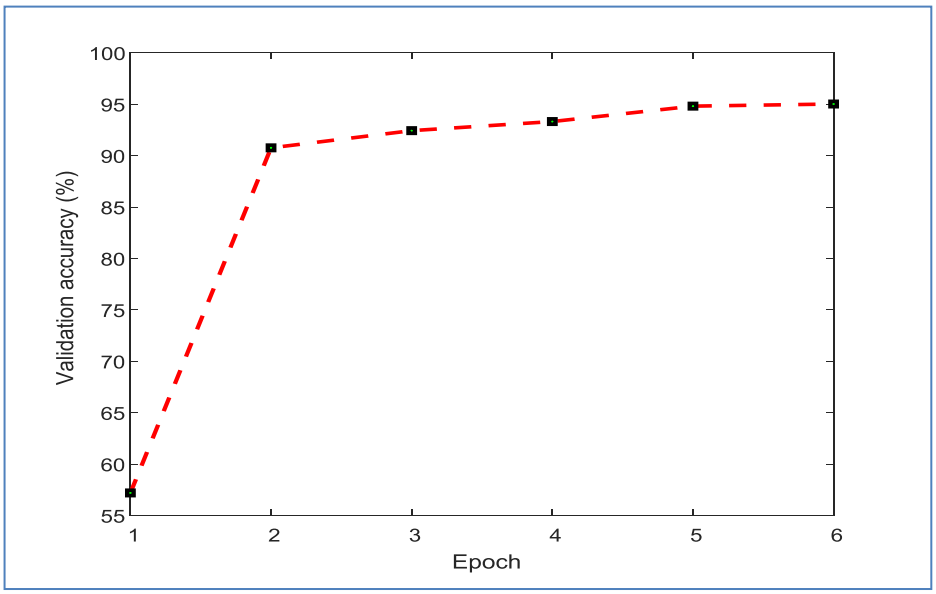


Figure 12. Average Keyword Detection Model Performance.

Validation accuracy curve averaged over all keyword classes during training. The graph shows consistent improvement across 5 epochs, with accuracy stabilizing above 94%, indicating strong convergence of the CNN model and generalization across different crime-related keywords.

Table 6: Epoch-wise Validation Accuracy of the Keyword Detection Model

Epoch	Validation accuracy %
0	57.1877
1	90.7546
2	92.4265
3	93.3118
4	94.8104
5	95.0174

This table presents the progression of validation accuracy over training epochs for the CNN-based keyword detection model. The sharp increase in accuracy after the first epoch, followed by steady improvement, indicates rapid convergence and strong learning performance across all keyword classes.

6.2 Speaker Identification and Gender Classification Models

The SEERIUM system integrates two critical modules for speaker-related analysis: one for identifying known speakers already present in the database, and another for predicting speaker attributes—notably gender—when no direct match is found. Both modules leverage convolutional neural networks (CNNs) trained on MFCC features, along with

their delta and delta-delta coefficients, which provide a rich representation of the temporal and spectral characteristics of speech.

6.2.1 Speaker Identification Model

The speaker identification model was designed to classify voice recordings based on a predefined list of known individuals. Using supervised learning, the model learns to associate specific vocal patterns with individual speaker identities. The CNN was trained using a dataset of annotated recordings that varied by speaker age, gender, and dialect to ensure robustness.

Three separate training sessions were conducted to evaluate consistency and generalization. The results are shown in the following figures:

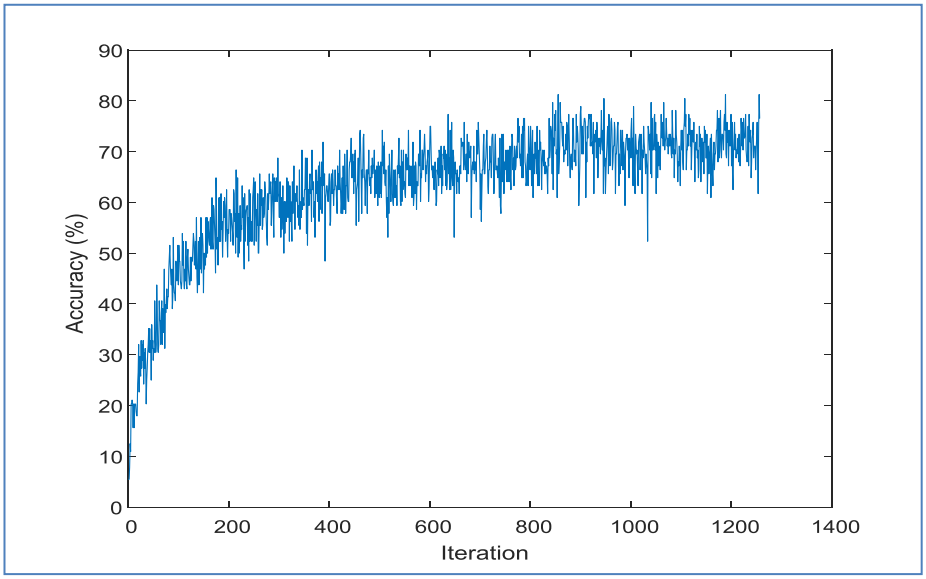


Figure 13. Training progress for speaker identification (Accuracy).

Training accuracy curve for the speaker identification model. The graph shows consistent improvement across epochs, reaching a stable accuracy of approximately 71%, indicating effective learning from the speaker data.

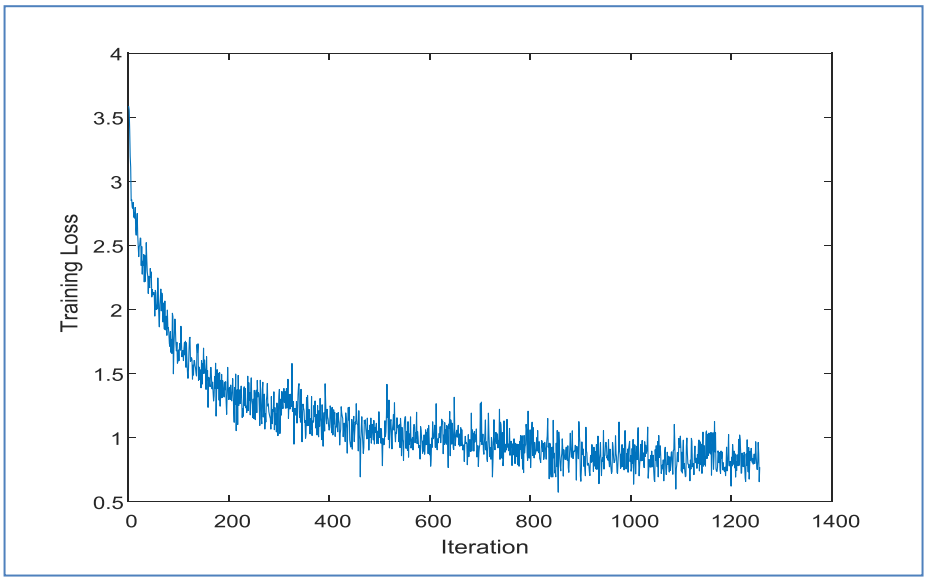


Figure 14. Training loss for speaker identification (Accuracy).

Loss curve during the training phase of the speaker identification model. The decreasing loss confirms convergence and the model's ability to minimize classification error over time.

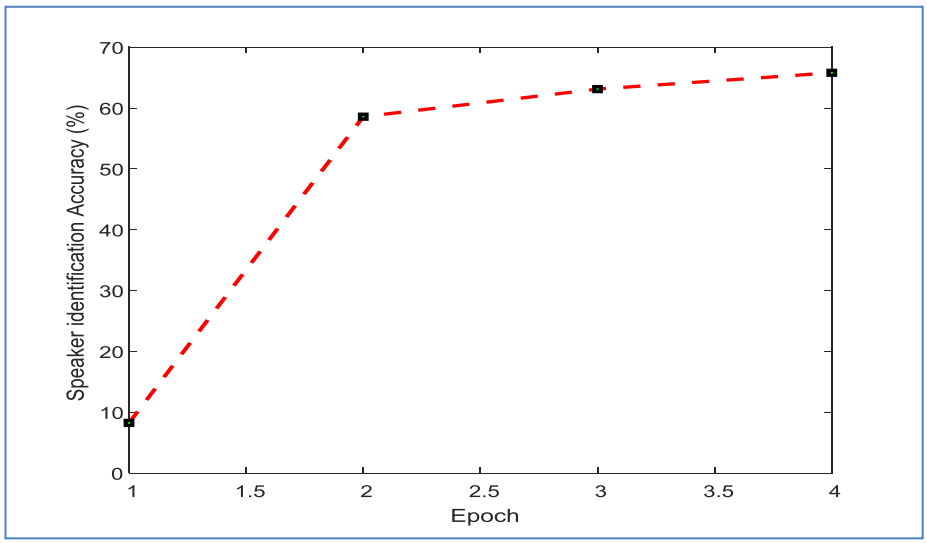


Figure 15. Average speaker identification (Accuracy) Model Performance

Summary of model performance metrics (Accuracy, Precision, and Recall, for speaker identification. The results confirm consistent classification across multiple speaker classes with satisfactory precision.

6.2.2 Speaker prediction Model (Gender Classification)

To enhance SEERIUM's functionality when the speaker is not recognized, a gender classification module was implemented. This model approximates whether the speaker is male or female based solely on their voice characteristics. It serves as a fallback mechanism when speaker identification fails, offering basic but valuable profiling information.

As with speaker identification, the gender classification model was trained over three sessions. Accuracy results are presented in the following figures:

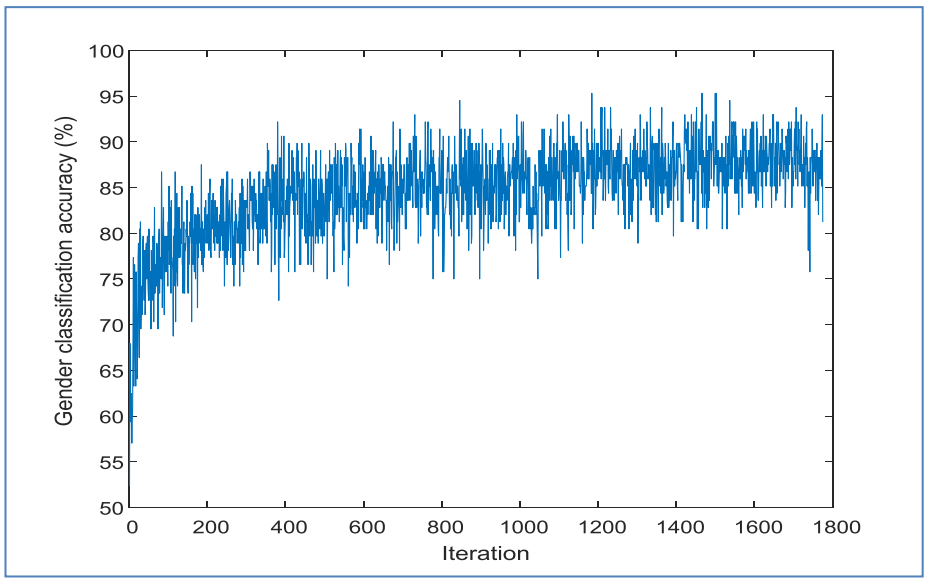


Figure 16. Training progress for Speaker prediction Model (Gender Classification).

Accuracy curve during training of the gender classification model. The graph indicates strong and steady learning, with final validation accuracy approaching 81%, suitable for real-world speaker profiling.

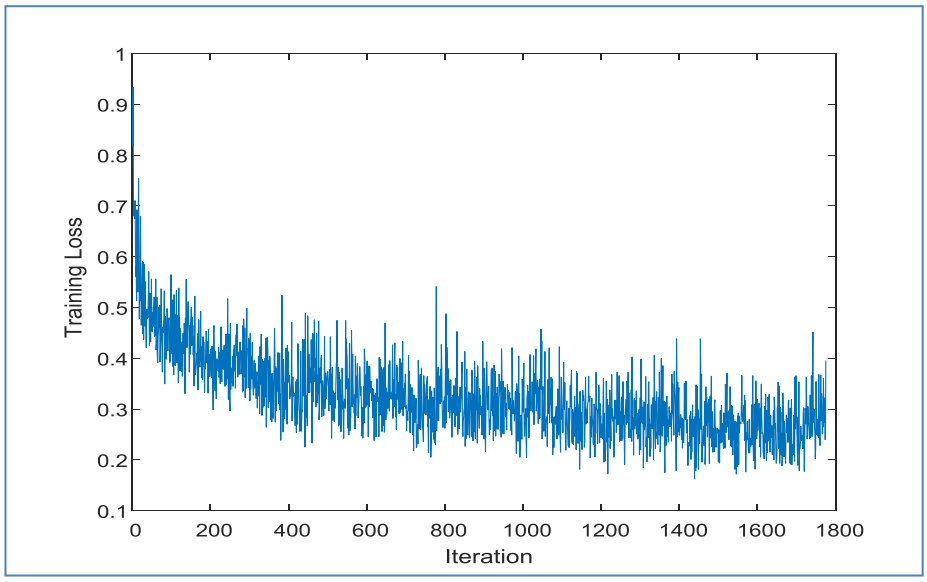


Figure 17. Training loss for Speaker prediction Model (Gender Classification).

Gender classification model loss during training. The downward trend shows successful learning with reduced error across epochs, suggesting good generalization and minimal overfitting.

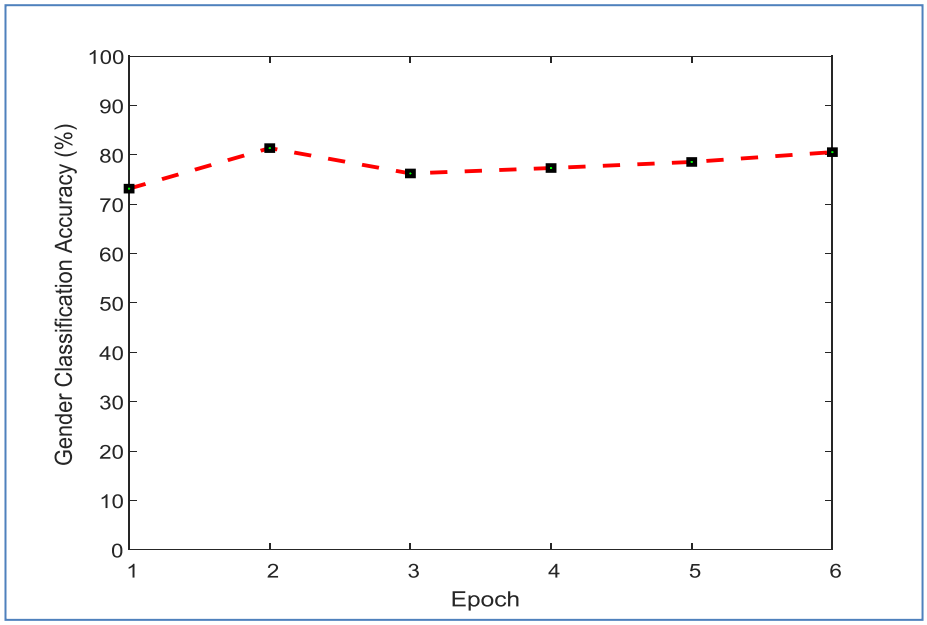


Figure 18. Average Training progress for speaker prediction Model Performance.

Overview of gender classification performance averaged across all sessions. This figure highlights the model's capability to accurately distinguish male and female speakers using vocal characteristics.

7 Realization of the System via GUI: SEERIUM

After finalizing the training and evaluation of the neural networks and validating the performance of the keyword detection, speaker identification, and speaker prediction models, we developed a fully functional GUI using MATLAB App Designer to:

- Provide an interactive platform to test and demonstrate the SEERIUM system in real-time.
- Bridge the gap between academic theory and a usable prototype.
- Facilitate future real-world deployment or integration into law enforcement tools.

Figure 19 shows the Real-time user interface for keyword detection and speaker analysis built in MATLAB.

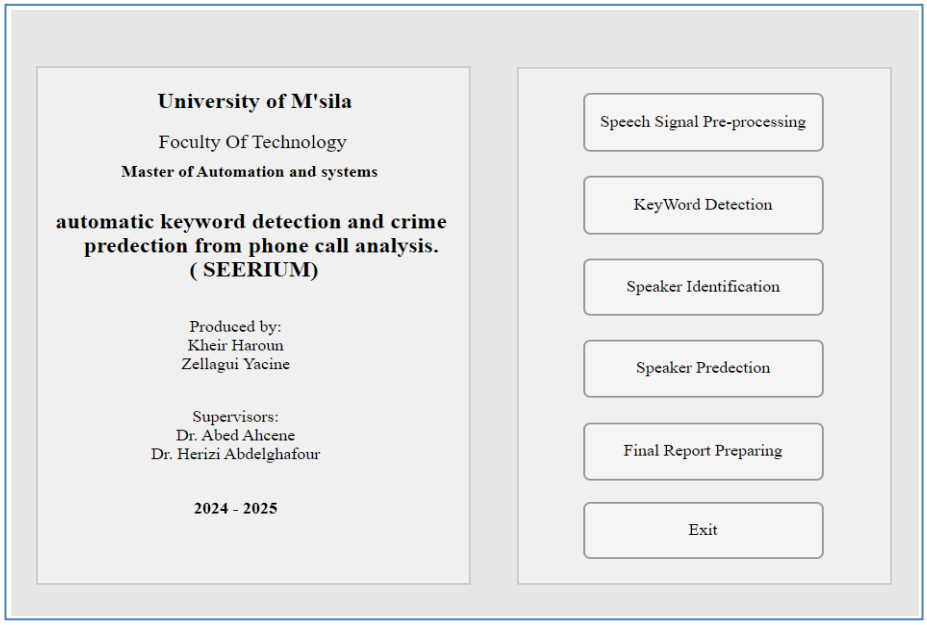


Figure 19. SEERIUM: GUI for Speech Preprocessing, Keyword Detection, and Speaker Analysis.

7.1 GUI Functionalities

The Audio Analysis Tool includes the following components:

7.1.1 Speech Signal Pre-processing

Applies filtering, normalization, and feature extraction (MFCC, spectrograms, etc.)

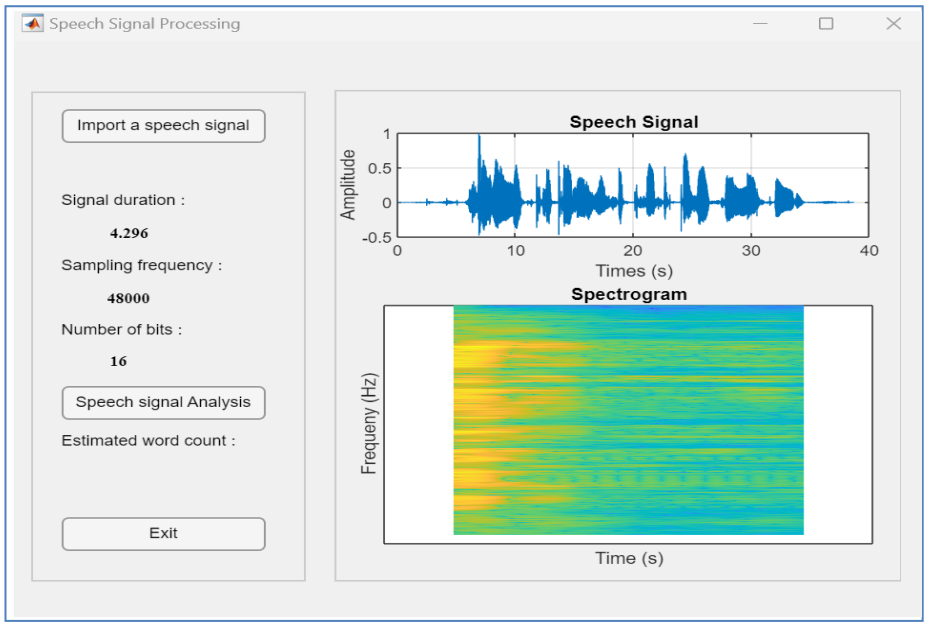


Figure 20. Speech Signal Preprocessing Interface – SEERIUM.

This interface allows the user to import a raw speech signal and visualize key signal properties such as duration, sampling frequency, and bit depth. It also displays the waveform and the corresponding spectrogram for time-frequency analysis. This step is essential for enhancing speech quality and preparing the signal for the further tasks.

7.1.2 Keyword Detection

Classifies whether dangerous or suspicious keywords exist in the voice file.

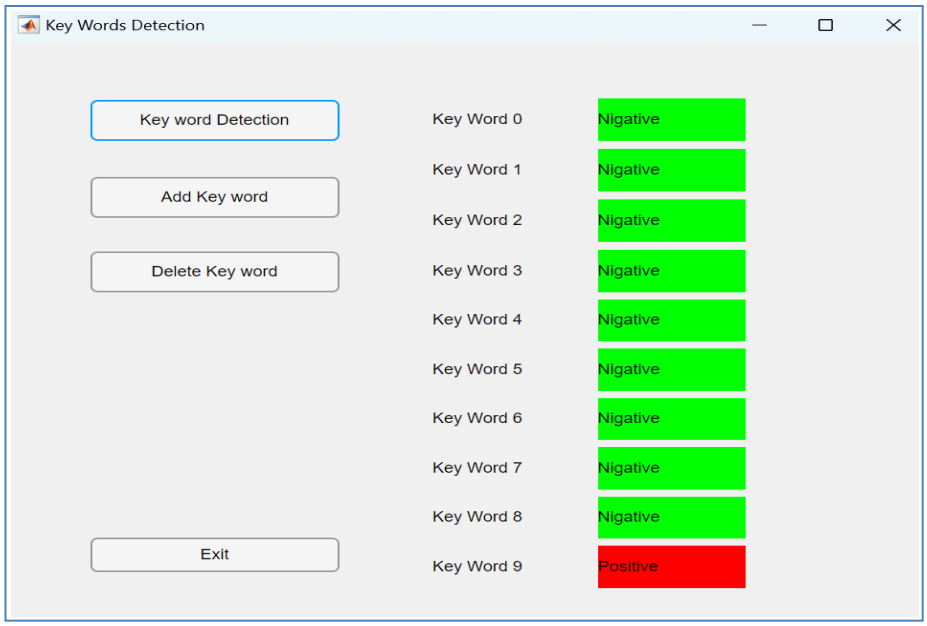


Figure 21. Keyword Detection Interface - SEERIUM.

This interface enables detection of predefined keywords within the speech signal. Users can add or delete keywords and initiate detection through a simple GUI. Each keyword is evaluated, and its detection status is displayed as either “Positive” (red) or “Negative” (green), helping highlight the presence of specific terms in the processed audio. This module is crucial for identifying suspicious or predefined phrases in real-time voice analysis.

7.1.3 Speaker Identification

Identifies the speaker if already present in the voice database and Displays his profile (Name, Age, Gender, ID, etc.)

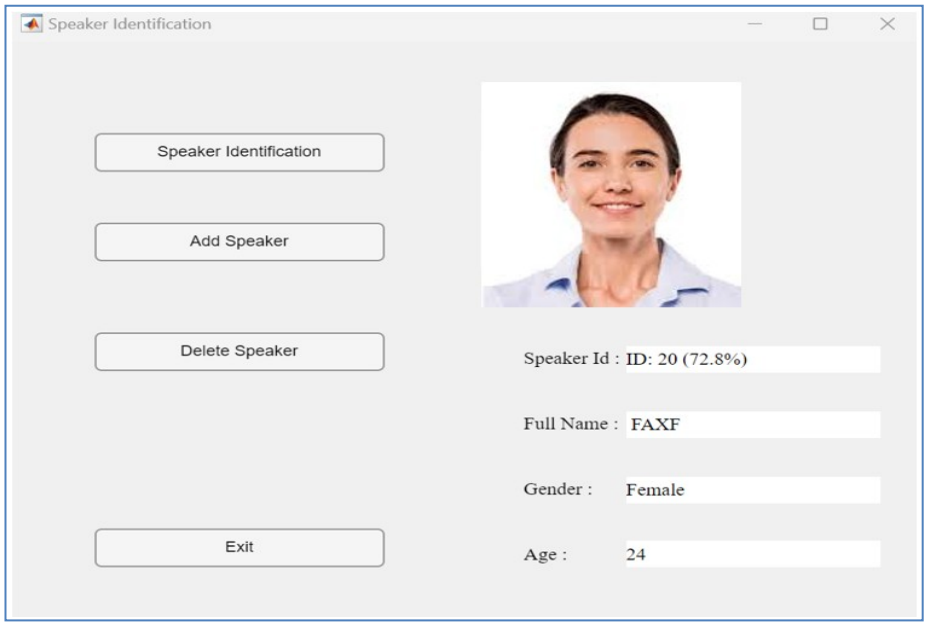


Figure 22. Speaker Identification Interface

This interface allows for the identification and management of speakers based on their voice input. Upon detection, the system displays the speaker’s ID, match confidence percentage, full name, gender, and age, along with a profile image. Users can also add or remove speaker entries. This feature is essential for recognizing known individuals in audio recordings and associating vocal data with identity attributes.

7.1.4 Speaker Prediction

In this case the systm predicts speaker identity based on voice characteristics, even if they are not in the database. It utilizes approximation models and clustering.

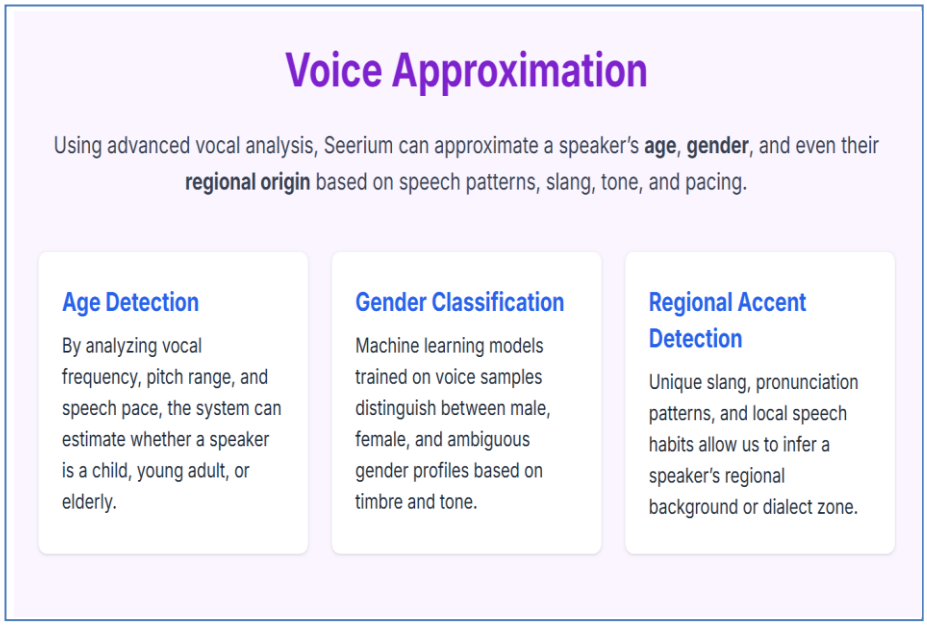


Figure 23. Voice Approximation Module.

This interface estimates a speaker's age, gender, and regional origin when no direct match is found in the database.

- Age Detection uses pitch, frequency, and speech rate.
- Gender Classification analyzes vocal tone to identify male, female, or ambiguous profiles.
- Regional Accent Detection relies on speech habits and slang to infer geographic background.

At this stage, more data is still needed, especially for improving the accuracy of regional origin detection.

7.1.5 Final Report Preparing

Seerium generates a clear report in Excel format summarizing the analysis. It includes detected keywords with timestamps, prediction confidence scores, and speaker ID details (name, age, gender, photo). Users can export this as a PDF or text file for easy sharing and review. Optional notes can be added before exporting.

7.2 Test Scenarios

- Uploaded or recorded voice samples were used for real-time processing.
- Test dataset included:
    - Voices with predefined keywords (e.g., drugs, theft …).
    - Known and unknown speakers to test both identification and prediction.

Result accuracy matched training metrics:

- Keyword detection: ~95%
- Speaker identification: ~71%
- Speaker prediction: ~81%

7.3 Web-Based Interface

To complement the MATLAB-based GUI, we also developed a lightweight, responsive web version of SEERIUM. This online platform allows users to upload or record voice samples directly from a browser and receive keyword detection and speaker prediction results in real-time. The web interface mirrors the core functionalities of the desktop GUI, including waveform visualization, result display, and report generation. It enhances accessibility, allowing institutions and users to test SEERIUM on any device without needing MATLAB.

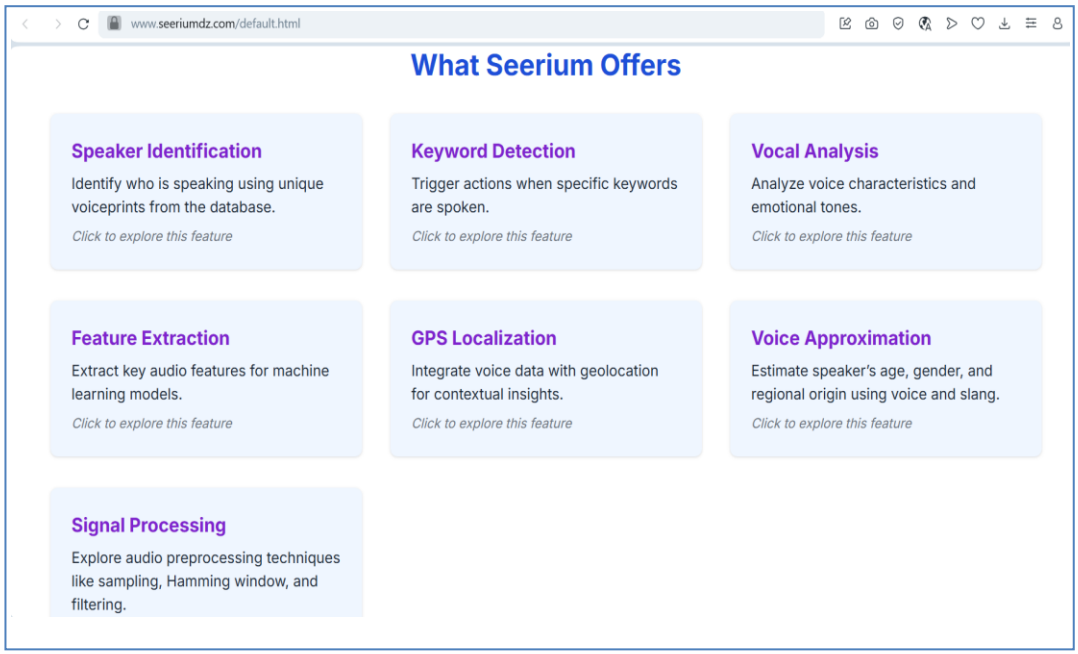


Figure 24. SEERIUM Web Interface – Feature Dashboard.

A screenshot of the SEERIUM web platform's homepage, showcasing the main features available to users, including speaker identification, keyword detection, voice approximation,and signal processing.

7.3.1 Benefits of This Approach

- Brings SEERIUM one step closer to a commercial or institutional prototype.
- Makes testing easier for stakeholders like law enforcement, researchers, and potential investors.
- Demonstrates your capacity to translate AI research into a functional product.

8 General Conclusion

This thesis presented the design, implementation, and evaluation of SEERIUM, an intelligent voice analysis system capable of performing automatic keyword detection, speaker identification, and crime prediction from phone call audio. The system was developed to overcome limitations of traditional voice recognition methods, particularly their vulnerability to noise, spoofing, and linguistic variability.

By using frequency-based features such as Mel-Frequency Cepstral Coefficients (MFCCs) and deep learning models like Convolutional Neural Networks (CNNs), SEERIUM achieved high accuracy in real-world scenarios. A custom dataset was created with crime-related keywords in Algerian dialect, enabling the system to recognize context-specific vocabulary and speaker traits with improved reliability.

The system achieved:

- **95% accuracy** in keyword detection,
- **81% accuracy** in speaker prediction,
- **71% accuracy** in speaker identification.

In addition to a fully functional MATLAB-based GUI, a responsive web version of SEERIUM was developed to allow users to upload or record audio from any device and receive real-time analysis. This platform enhances accessibility and demonstrates the system's potential for broader deployment in law enforcement, security, and forensic applications.

SEERIUM contributes a practical and adaptable solution to voice-based crime prediction. It combines biometric identification with speech content analysis and proves the value of deep learning in developing smart, ethical, and locally adapted surveillance tools. With continued development and real-world testing, SEERIUM has the potential to become a powerful platform for proactive crime detection and digital audio intelligence.

Abstract: Traditional voice recognition systems predominantly rely on anatomical features such as vocal tract geometry and articulatory patterns. This study introduces an alternative approach centered on the acoustic characteristics of speech, with an emphasis on frequency-domain features. The proposed system integrates spectral analysis, Mel-Frequency Cepstral Coefficients (MFCCs), and deep learning architectures to perform three tasks: keyword detection, gender classification, and speaker identification. Using a custom dataset comprising real-world, noise-contaminated voice recordings, the model achieved an accuracy of 94% for keyword detection, 80.5% for gender classification, and 71% for speaker identification. These results underscore the robustness of frequency-based features in non-ideal conditions and highlight their applicability in privacy-sensitive and locally processed voice recognition systems. Future work will explore advanced neural architectures and signal enhancement techniques to further improve performance across diverse environments. A web-based platform was also developed to allow users to test the system via voice uploads and receive immediate analysis results without needing MATLAB.

Keywords: Voice recognition, Frequency-domain features, Deep learning, Keyword detection, Speaker identification, MFCCs, Noisy data

References

[1] Md. Sahidullah, S. Sigtia, and G. E. Henter, "Comparison of feature representations for robust speech recognition," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 28, pp. 2185–2199, 2020.

[2] D. Jurafsky and J. H. Martin, Speech and Language Processing, 3rd ed., Draft. [Online]. Available: https://web.stanford.edu/~jurafsky/slp3/, 2023.

[3] D. Yu and L. Deng, Automatic Speech Recognition: A Deep Learning Approach, Springer, 2016.

[4] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in Proc. IEEE ICASSP, pp. 6645–6649, 2013.

[5] H. Dubey and S. Bhattacharya, "A perceptual approach to speech denoising using deep neural networks," Neurocomputing, vol. 314, pp. 200–212, 2018.

[6] Y. Zhang, Z. Chen, and M. Yu, "End-to-End Models for Speech Recognition: A Review," IEEE Access, vol. 8, pp. 142344–142359, 2020.

[7] R. Tang and J. Lin, "Deep residual learning for small-footprint keyword spotting," in Proc. IEEE ICASSP, 2018, pp. 5484–5488.