An Automated Calls Retrieval System

(ACRS)

By

GC ABDUL RAUF

GC SHAHEER MEHMOOD

GC NAQEEBULLAH

GC SANAULLAH

Supervised by:

Dr. Shibli Nisar

Submitted to the faculty of Department of Electrical Engineering,
Military College of Signals, National University of Sciences and Technology,Islamabad, in partial fulfillment for the requirements of B.E Degree in Electrical (Telecom) Engineering.

June 2022

In the name of ALLAH, the Most benevolent, the Most Courteous

CERTIFICATE OF CORRECTNESS AND APPROVAL

This is to officially state that the thesis work contained in this report

"An Automated Calls Retrieval System"

is carried out by

GC ABDUL RAUF

GC SHAHEER MAHMOOD

GC NAQEEBULLAH

GC SANAULLAH

under my supervision and that in my judgement, it is fully ample, in scope and excellence, for the degree of Bachelor of Electrical (Telecom.) Engineering in Military College of Signals, National University of Sciences and Technology (NUST), Islamabad.

Approved by

Supervisor
Dr.Shibli Nisar
Department of EE, MCS

Date: _____

DECLARATION OF ORIGINALITY

We hereby declare that no portion of work presented in this thesis has been submitted in support of another award or qualification in either this institute or anywhere else.

ACKNOWLEDGEMENTS

Allah Subhan'Wa'Tala is the sole guidance in all domains.

Our parents, colleagues and most of all supervisor Dr. Shibli Nisar without your guidance.

The group members, who through all adversities worked steadfastly.

Plagiarism Certificate (Turnitin Report)

This thesis has 15% similarity index. Turnitin report endorsed by Supervisor is attached.

_____

GC Abdul Rauf

00000278748

_____

GC Shaheer Mahmood

00000278727

_____

GC Naqeebullah

00000278743

_____

GC Sanaullah

00000278740

_____

Signature of Supervisor

ABSTRACT

Nowadays LEAs manually retrieve calls of a specific user. It is impossible to do the same for millions of speakers. LEAs faced severe issues when retrieving call conversations manually from big databases. Speaker identification technologies are widely applied in voice authentication, security and surveillance,electronic voice eavesdropping, and identity verification.The proposed model will automate the entire process by recognizing the suspicious speaker and dig out all the audio conversations of target.To address problem, AI based solution is recommended .This project will mainly benefit the Military or LEAs for security purposes. Moreover. The proposed model can easily be implemented in any international or national military organization or for commercial purposes, which will help in assisting international collaborations

TABLE OF CONTENTS

Chapter 1

Introduction

1.1 Overview

ACRS is an inbound calling system as well as an automated outbound call management system. Inbound call systems allow you to record a unique audio message for distinct consumer pain points, allowing each caller to select from your menu options and solve a specific problem.

1.2 Problem statement

The suggested ACRS system will be able to extract questionable speaker calls from a vast data/repository. Using automated storage and retrieval systems instead of buying and storing products in your warehouse to maintain a healthy backlog is not only a viable option, but also a profitable one. Manually monitoring every voice call by law enforcement officials is a demanding task, and this complexity multiplies when terrorists utilise new codes for their operations [1]. Law enforcement organisations face a huge challenge in keeping up with new keywords and manually monitoring every calls. Researchers have made numerous attempts to use machine learning techniques to systematically digitise this procedure.

1.3 Approach

Automated calling services save you time and money by allowing you to make hundreds, if not thousands, of automated calls to a group or organisation at simultaneously. Any telephone system that interacts with callers without human input other than the recipient is known as an automated phone system**.**

1.4 Scope

Currently, LEAs must manually retrieve calls made by a specific user. It would be impossible to do so for millions of people. When manually extracting call conversations from large databases, LEAs ran into serious problems. Authentication, security and surveillance, electronic voice eavesdropping, and identity verification all use speaker identification technology. The technology we propose will automate the entire retrieval procedure and assist LEAs in their operations.

1.5 Objectives

The proposed model will function in two stages: The goal of this project is to use a vast database to find calls from known speakers. This project will assist law enforcement authorities (LEAs) in locating questionable speakers' calls.

1.6 Deliverables

Sr. No	Tasks	Deliverables
1	Literature Review	Literature Survey
2	Requirements Specification	Software Requirements Specification document (SRS)
3	Detailed Design	Software Design Specification document (SDS)
4	Implementation	Project demonstration
5	Training	Deployment plan
6	Testing	Evaluation plan
7	Deployment	Complete desktop application with the necessary

1.7 Justification for Selection of Topic

For security purposes, this project will primarily help the military or LEAs. Furthermore, this will not only aid LEAs in their operations by collecting questionable speakers calls from a large database, but it will also be useful in commercial applications like as cellular providers. The gadget will not only improve security for our community, but it will also provide valuable research in the form of publishable research publications. The proposed paradigm is simple to apply in any international or national military organisation, as well as for commercial objectives, and will aid international cooperation.

1.8 Overview of the Document

This document explains how our application AACRS works in detail. It begins with a literature review, which highlights previous work in a similar field, a system requirement analysis, a system architecture that highlights the software modules and represents the system in the form of a component diagram, a Use Case Diagram, a Sequence Diagram, and the overall design of the system. The discussion will then move on to a full description of all of the components involved. The system's dependencies, its relationship with other goods, and its ability to be reused will also be explored. Finally, test scenarios and a recommendation for future work were provided.

1.9 Document Conventions

Headings are numbered in order of priority. Font used is Times New Roman. All the main headings are of size 16 and bold. All the second level sub-headings are of size 14 and bold. All the further sub-headings are of size 12 and bold. Where necessary references are provided in this document. However, where references are not provided, the meaning is self-explanatory.

1.10. Intended Audience

This document is intended for:

1. **Developers: (Project Group)**

To ensure that they are constructing the suitable enterprise that meets the requirements outlined in this report.

2. **Testers: (Project Group, Supervisor)**

To have a detailed list of the features and capabilities that must respond to requirements.

3. **Users:**

To be familiar with the task's potential and how to use/react in letdown situations, as well as to suggest additional features that would make it significantly more valuable.

4. **Documentation writers: (Project Group)**

Recognize which characteristics and how they should be clarified. What innovations are required, how will the framework react in each client's activity, what potential framework disappointments may emerge, and what are the solutions to each of those disappointments, and so on.

5. **Project Supervisors: (Dr. Shibli Nisar)**

The project supervisor will utilise this document to ensure that all needs have been understood and, in the end, that the requirements have been implemented correctly and thoroughly.

Chapter 2

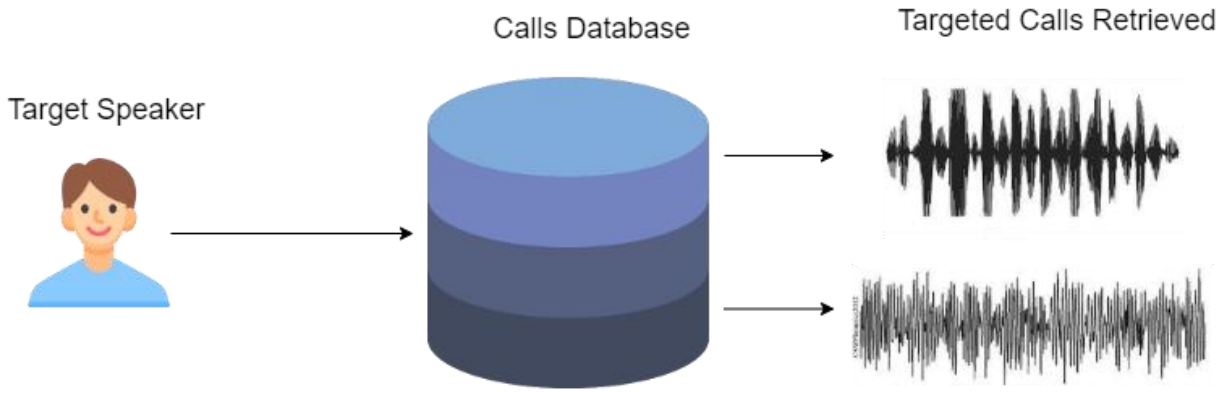
Literature Review

2.1 What is call?

In finance, a call typically means one of two things:

1. A call option is a derivatives contract that gives the owner the right, but not the duty, to buy a predetermined amount of an underlying security at a predetermined price within a predetermined time frame.

2. In a call auction, buyers set a maximum acceptable price to buy and sellers set a minimum acceptable price to sell an asset on an exchange for a specified period of time. This practise of matching buyers and sellers promotes liquidity and lowers volatility. A call market is another name for the auction.



2.2 Automatic Call Retrieving System

An Automated Calls Retrieval System (ACRS) is a set of equipment and controls that handle and retrieve materials as needed with precision, accuracy, and speed, all while maintaining a certain level of automation. Smaller automated systems to bigger computer-controlled retrieval systems fully integrated into a manufacturing and distribution process are all available. A range of computer-controlled technologies for automatically depositing and recovering loads to and from predetermined storage sites are referred to as ACRS.

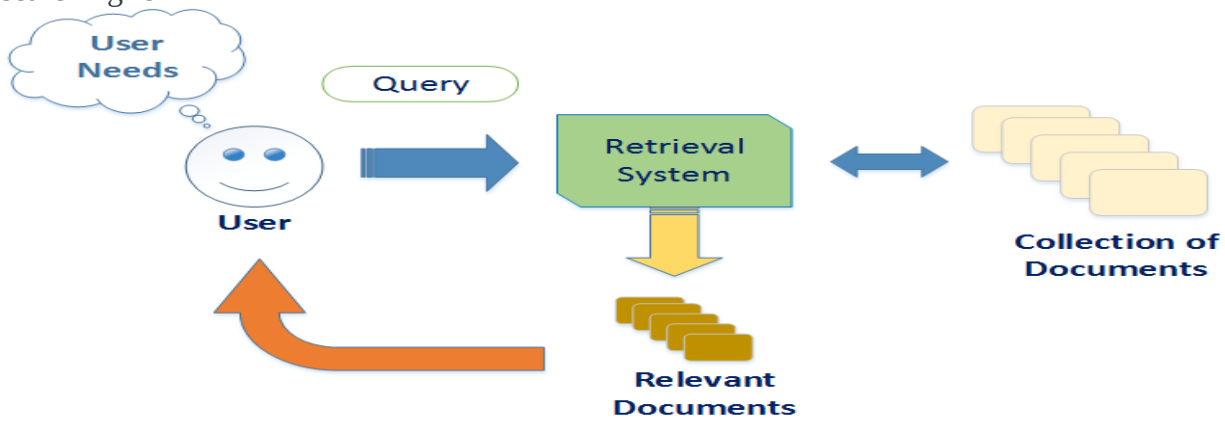
2.3 How does an IRS System works?

Although some contact centres continue to record calls on tapes or CDs that are not connected to their servers in any way, the vast majority of contact centres, whether they own their own technology or use a hosted technology platform, now record calls on network-attached storage (NAS) systems or storage area networks (SAN).

The former is essentially a single block of hard disc storage that links to the IT network, allowing authorised personnel to access data like digital call recordings stored in that storage device in

formats like.wav files. In the meantime, the latter is a platform that connects a variety of storage devices to the operating system. These unique storage systems could include tape libraries or optical jukeboxes, as well as robotic data storage devices that can automatically load and unload call records stored on CD or DVD.

They could also be independent storage platforms for digital call recordings in the form of.wav files. Calls that are digitally recorded and integrated with a private branch exchange (PBX) or computer telephony integration (CTI) server and customer relationship management (CRM) system can be retrieved more easily because the server captures certain associated data or metadata that can be 'tagged' to the call. Date, time, call length, agent extension number, caller line identity (CLI), and dialled number are examples of metadata. Most authorised contact centre staff can obtain call recordings very quickly using this metadata, given they know what they're searching fo



2.4 Previous work

This section summarises the previous research on retrieval systems. Information retrieval has been the subject of extensive research in the last decade.

1. Automatic discrimination between singing and speaking voices for a flexible music retrieval system

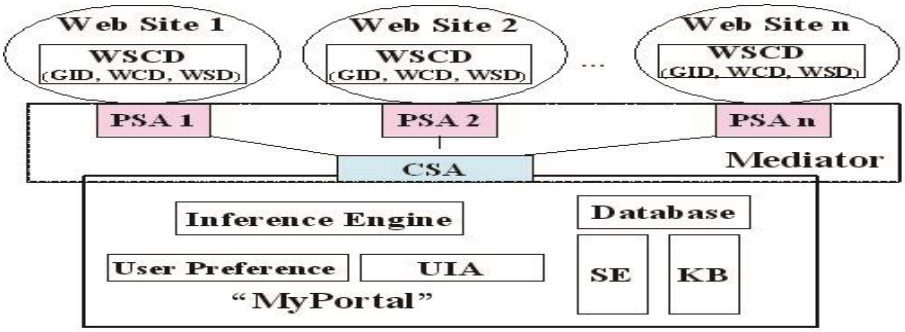
Yasunori Ohishi et al. (4) describe a music retrieval system that allows users to obtain a song using one of two methods: singing the melody or stating the title. A way of automatically differentiating between singing and speaking voices is required to allow the user to employ those methods without altering the voice input mode. We then combined two measures: MFCC and an F0 (voice pitch) contour to construct an automatic method of differentiating between singing and speaking voices based on these findings. Our technique achieved 68.1 percent accuracy for 200ms signals and 87.3 percent accuracy for 2s signals in experiments. We eventually constructed a music retrieval system based on this strategy that can take both singing and speaking voices for the melody and title.

2. Information retrieval based call classification

A fully automatic call classification system for customer service selection was reported by Jan Kneissler et al (2). Calls are classified based on one consumer response to a "How may I assist you?" prompt. We include two new aspects to our information retrieval-based call classifier that boost classification accuracy significantly: the application of a-priory word relevance based on class knowledge and classification confidence estimation. We discuss the spontaneous speech recognizer as well as the classifier, and look at how speech recognition and call categorization accuracy are related.

3. An Architecture for Personal Semantic Web Information Retrieval System

Semantic Web and Web service technologies, as demonstrated by Haibo Yu et al (2), have brought both new possibilities and problems for autonomous information processing. Web services offer a new way to access Web resources, but they've been maintained separately from standard Web content resources up until now. A conceptual architecture for a personal semantic Web information retrieval system is proposed in this work. It combines semantic Web, Web services, and multi-agent technologies to provide not only accurate positioning of Web resources, but also automatic or semi-automatic integration of hybrid Web contents and Web services.First, "all participants contribute consistently to the semantic description" in this research web site 1. Second, "web content integration with web services." Third, offering a portal to all relevant information for the user.



4. (An) automatic document retrieval system

The Korea Advanced Institute of Science and Technology describes an automatic document retrieval system that analyses the contents of documents and search requests expressed in natural language and returns answers to the search request the documents that appear to be most relevant to the request despite not being indexed by the exact terms of the request. A number of approaches for organizing indexing systems that can increase overall system performance are investigated. Functional word removal, stem dictionary and suffix list creation, word deconstruction, thesaurus development, and phrase generation are only a few examples.The system has four processing methods that can be used to not only imitate a real-world working environment, but also to evaluate the effectiveness of the various processing options. The system's retrieval performance is demonstrated, illustrating the utility of each method.

Chapter 3

ACRS Methodology

As the system was created to make the entire procedure as seamless and exact as possible, we've included the entire technique in figure 3.1. After initial processing, we send our speech signal to the system as an input to be examined further. To begin, the feature
The speech signal is used in the extraction process, which will be discussed later. For the
As a first priority, we used the CNN model for our system. Our model was trained using this.
We also employed SVM classifier, Nave Bayes Algorithm, and Random Forest models.
To test the accuracy of these models, use a forest classification model, for example. We will also provide explanations.the paper's specifics on these modelsbriefly. Almost 80% data was trained for each Machine Learning Algorithm, data was collected and accuracy was determined. The result The results of the algorithm were then displayed in our system as suspicious or non-suspicious word labels. interface for desktop applications

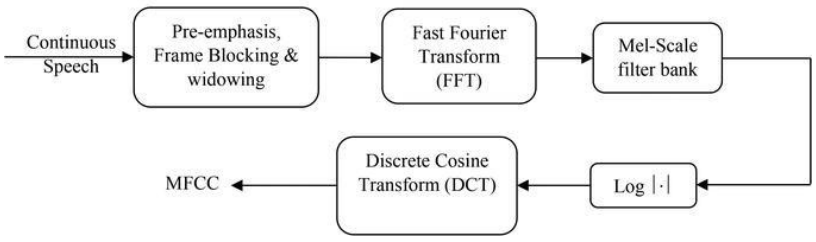
3.1 MFCC Feature Extraction

There is some extraneous data in the voice signals, which we identify as noise. Before extracting the features, this audio signal must be filtered to remove any unwanted noise. Feature extraction is used to evaluate a speech signal and depict it in a number of relevant components. This also gets rid of the unnecessary elements of speech.We typically utilize the mel-frequency cepstrum coefficient to depict the short-term power spectrum of a sound in sound processing (MFCC). The MFCCs (mel-frequency cepstral coefficients) are the coefficients that make up an MFC. These are obtained from the audio clip's nonlinear spectrum, SSTRUM. The MFCC feature extraction approach was also used

A. **Pre-emphasis:** The word "pre-emphasis" refers to the act of filtering a signal such that higher frequencies are highlighted. It is most typically employed to balance the spectrum of vocal sounds with a sharp high frequency roll off. With pre-emphasis, the glottal effects in the sound are erased. As a pre-emphasis filter, the following transfer function is widely employed.

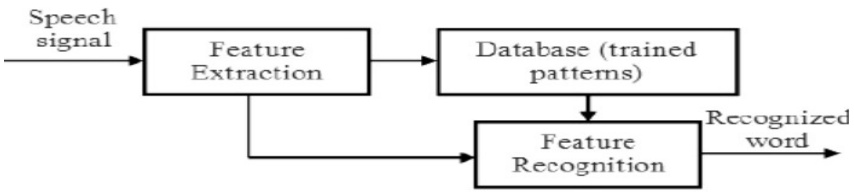
$$H(z) = 1 - bz{-}1$$

B.Windowing and Frame Blocking: The voice signal is a signal that changes slowly over time. The speech signal must be analyzed for steady acoustic properties over a short period of time. This is why speech analysis is performed on brief segments of speech that are static.
These short-term measurements are made in 20- or 10-second periods. This allows for the tracking of individual speech sounds with temporal features, and this window is large enough to provide adequate spectral resolution. During the Fourier transform, hamming windows are typically utilized to improve the harmonics and minimize the edge effect.



C. Fourier transform: By applying the Fourier transform to the windowed frames, the magnitude domain (time to frequency) is converted.

D. Mel spectrum: After applying a Mel filter bank to a Fourier converted signal, the Mel spectrum is the following stage. This is a unit of measurement based on the frequency perceived by the human ear. Below 1 kHz, the Mel scale has linear frequency spacing, and above 1 kHz, it has logarithmic frequency spacing.



E. The discrete cosine transform (DCT): Translates the log Mel spectrum into the time domain. The final MFCC characteristics that were required come out of this step. Acoustic vectors are the distinct number of various coefficients created by using DCT. For our system, 13 distinct coefficients were extracted.

3.2 Machine Learning Algorithms

We chose CNN as our core algorithm for our system, but we also looked at the behavior of other machine learning algorithms for comparison. We'll go over the fundamentals of each algorithm before diving into the analysis.

3.2.1. Convolutional Neural Network

Convolutional neural networks are a machine learning-based model that uses the convolutional mathematical process. In one of the layers, we employ convolution rather than general matrices multiplication. The input, output, and hidden layers of this neural network are designated accordingly.Because the activation function and convolution mask the inputs and outputs of this layer, the middle layer is the one that is hidden. Convolution is carried out by these hidden levels. The layer's inputs, in this case the MFCC features and the convolutional kernel, are taken as a dot product. The kernel is then slid across the input matrix, producing a feature map that serves as the input for the following layer. Pooling and normalizing layer processes, among others, are also carried out.

In addition to the convolutional layer, there is a pooling layer that minimizes data by mixing the data clusters of one layer's output into a distinct cluster for the next layer. The two most frequent methods of pooling are max pooling and average pooling, which take the maximum value of each cluster and the average value, respectively.

The next step is dropout, which prevents a neural network from over fitting by comparing different probability values for improved accuracy. In the following sections, we will describe the findings for our dataset once the convolutional neural network has been applied to it

$$Y(n) = \sum_{n=-\infty}^{\infty} x(n)e^{-i\omega n}$$

Figure 3.2.1(1)

activation_3 (Activation)	(None, 4000, 32)	0	['add_1[0][0]']
max_pooling1d_1 (MaxPooling1D)	(None, 2000, 32)	0	['activation_3[0][0]']
conv1d_7 (Conv1D)	(None, 2000, 64)	6208	['max_pooling1d_1[0][0]']
activation_4 (Activation)	(None, 2000, 64)	0	['conv1d_7[0][0]']
conv1d_8 (Conv1D)	(None, 2000, 64)	12352	['activation_4[0][0]']
activation_5 (Activation)	(None, 2000, 64)	0	['conv1d_8[0][0]']
conv1d_9 (Conv1D)	(None, 2000, 64)	12352	['activation_5[0][0]']
conv1d_6 (Conv1D)	(None, 2000, 64)	2112	['max_pooling1d_1[0][0]']
add_2 (Add)	(None, 2000, 64)	0	['conv1d_9[0][0]', 'conv1d_6[0][0]']
activation_6 (Activation)	(None, 2000, 64)	0	['add_2[0][0]']
max_pooling1d_2 (MaxPooling1D)	(None, 1000, 64)	0	['activation_6[0][0]']
conv1d_11 (Conv1D)	(None, 1000, 128)	24704	['max_pooling1d_2[0][0]']
activation_7 (Activation)	(None, 1000, 128)	0	['conv1d_11[0][0]']
conv1d_12 (Conv1D)	(None, 1000, 128)	49280	['activation_7[0][0]']
activation_8 (Activation)	(None, 1000, 128)	0	['conv1d_12[0][0]']
conv1d_13 (Conv1D)	(None, 1000, 128)	49280	['activation_8[0][0]']
conv1d_10 (Conv1D)	(None, 1000, 128)	8320	['max_pooling1d_2[0][0]']
add_3 (Add)	(None, 1000, 128)	0	['conv1d_13[0][0]', 'conv1d_10[0][0]']

Figure 3.2.1(2)

Layer (type)	Output Shape	Param #	Connected to
input (InputLayer)	[(None, 8000, 1)]	0	[]
conv1d_1 (Conv1D)	(None, 8000, 16)	64	['input[0][0]']
activation (Activation)	(None, 8000, 16)	0	['conv1d_1[0][0]']
conv1d_2 (Conv1D)	(None, 8000, 16)	784	['activation[0][0]']
conv1d (Conv1D)	(None, 8000, 16)	32	['input[0][0]']
add (Add)	(None, 8000, 16)	0	['conv1d_2[0][0]', 'conv1d[0][0]']
activation_1 (Activation)	(None, 8000, 16)	0	['add[0][0]']
max_pooling1d (MaxPooling1D)	(None, 4000, 16)	0	['activation_1[0][0]']
conv1d_4 (Conv1D)	(None, 4000, 32)	1568	['max_pooling1d[0][0]']
activation_2 (Activation)	(None, 4000, 32)	0	['conv1d_4[0][0]']
conv1d_5 (Conv1D)	(None, 4000, 32)	3104	['activation_2[0][0]']
conv1d_3 (Conv1D)	(None, 4000, 32)	544	['max_pooling1d[0][0]']
add_1 (Add)	(None, 4000, 32)	0	['conv1d_5[0][0]', 'conv1d_3[0][0]']

```
activation_9 (Activation)      (None, 1000, 128)    0           ['add_3[0][0]']

max_pooling1d_3 (MaxPooling1D)  (None, 500, 128)    0           ['activation_9[0][0]']

conv1d_15 (Conv1D)             (None, 500, 128)     49280       ['max_pooling1d_3[0][0]']

activation_10 (Activation)     (None, 500, 128)     0           ['conv1d_15[0][0]']

conv1d_16 (Conv1D)             (None, 500, 128)     49280       ['activation_10[0][0]']

activation_11 (Activation)     (None, 500, 128)     0           ['conv1d_16[0][0]']

conv1d_17 (Conv1D)             (None, 500, 128)     49280       ['activation_11[0][0]']

conv1d_14 (Conv1D)             (None, 500, 128)     16512       ['max_pooling1d_3[0][0]']

add_4 (Add)                    (None, 500, 128)     0           ['conv1d_17[0][0]',
                                                                 'conv1d_14[0][0]']

activation_12 (Activation)     (None, 500, 128)     0           ['add_4[0][0]']

max_pooling1d_4 (MaxPooling1D)  (None, 250, 128)    0           ['activation_12[0][0]']

average_pooling1d (AveragePool  (None, 83, 128)     0           ['max_pooling1d_4[0][0]']
ing1D)

flatten (Flatten)              (None, 10624)        0           ['average_pooling1d[0][0]']

dense (Dense)                  (None, 256)          2720000     ['flatten[0][0]']

dense_1 (Dense)                (None, 128)          32896       ['dense[0][0]']

output (Dense)                 (None, 8)            1032        ['dense_1[0][0]']

==================================================================================================
Total params: 3,088,984
Trainable params: 3,088,984
Non-trainable params: 0
```

Figure 3.2.1(3)

Model training:

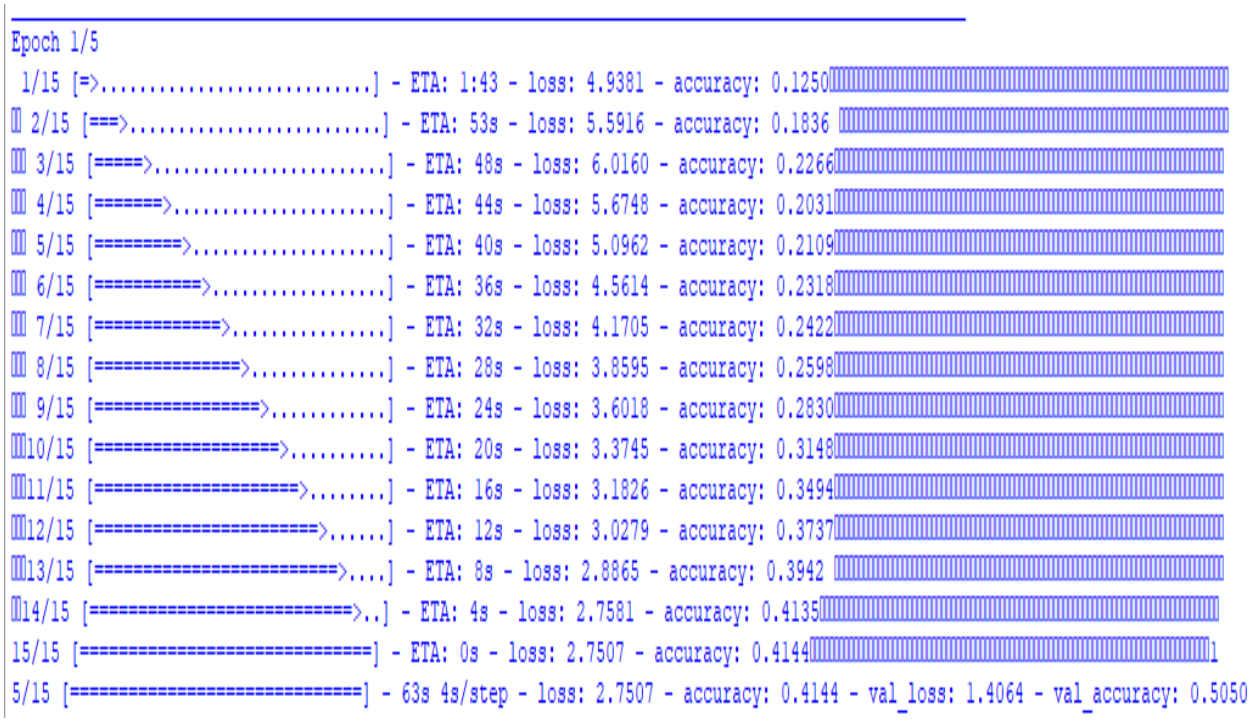


Figure 3.2.1(4)

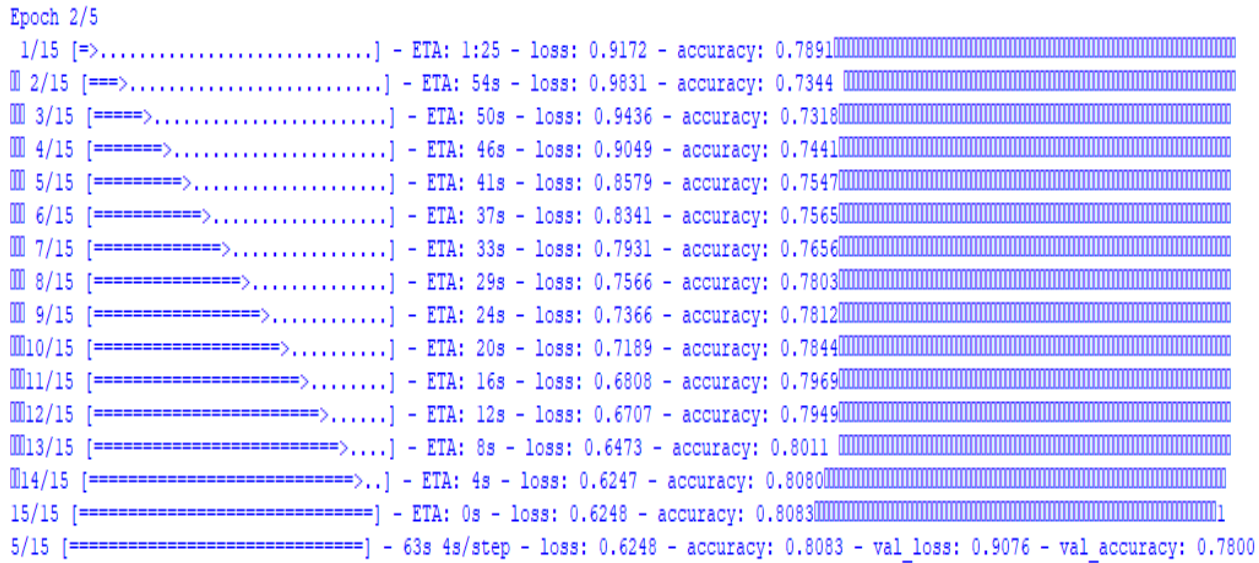


Figure 3.2.1(5)

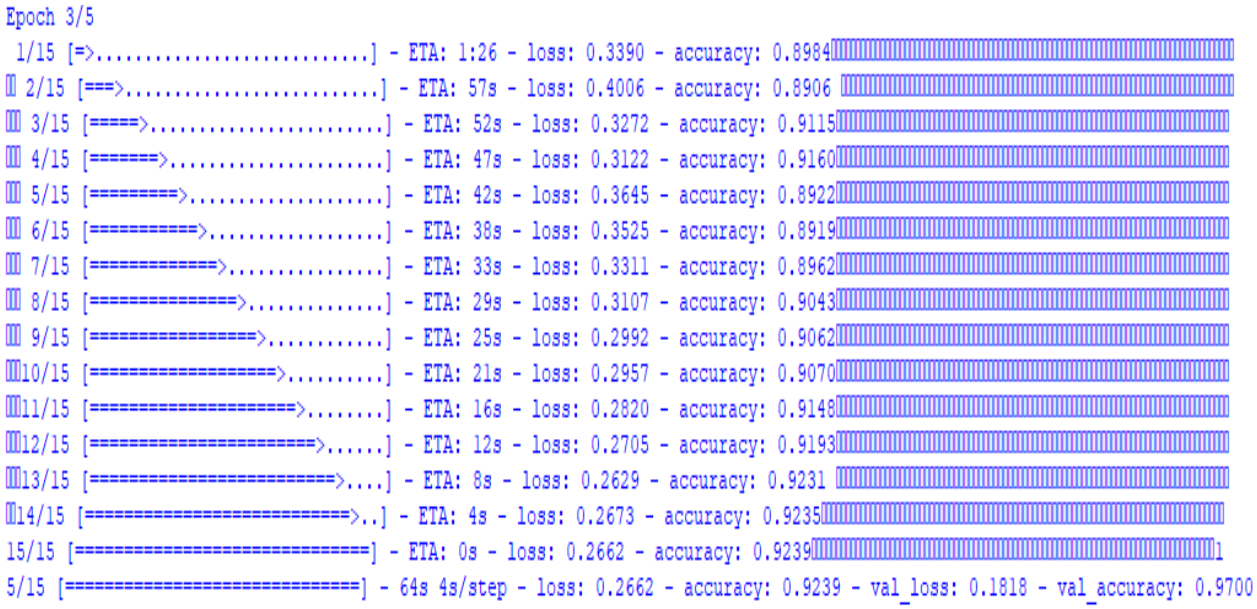


Figure 3.2.1(6)

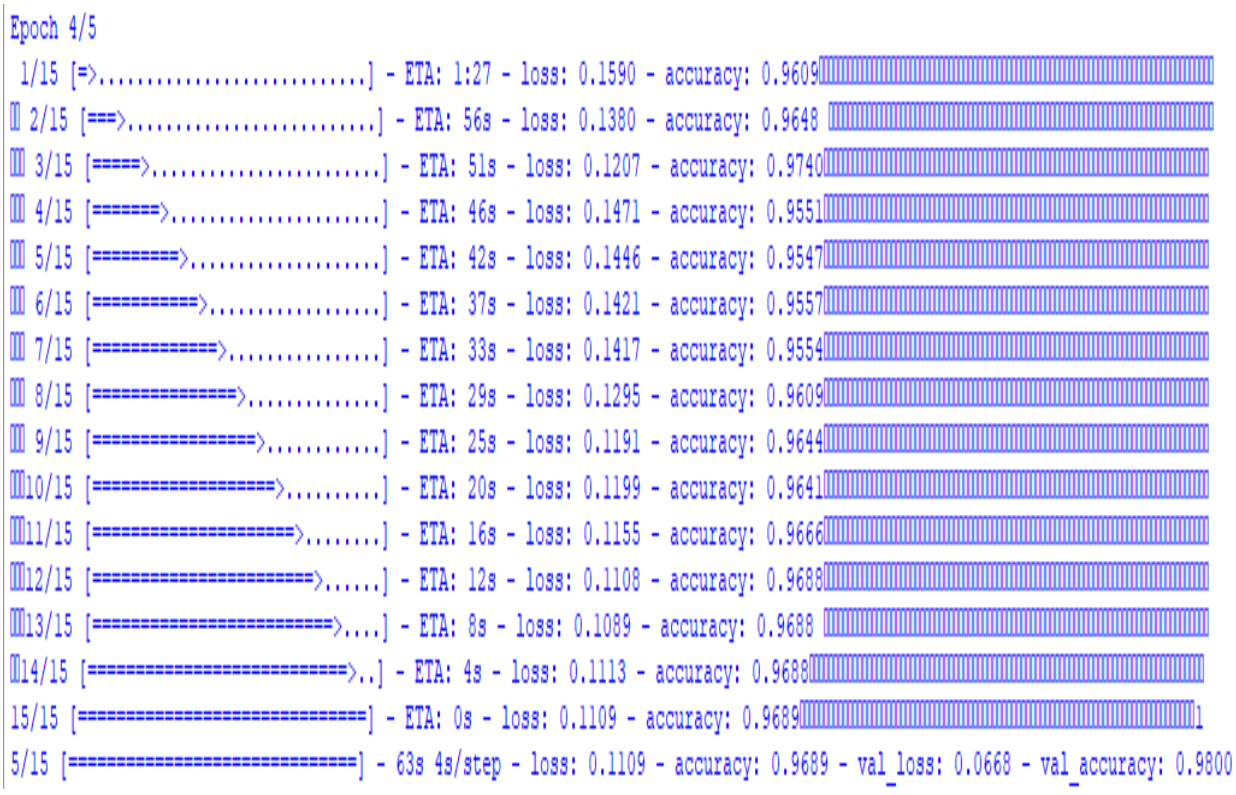


Figure 3.2.1(7)

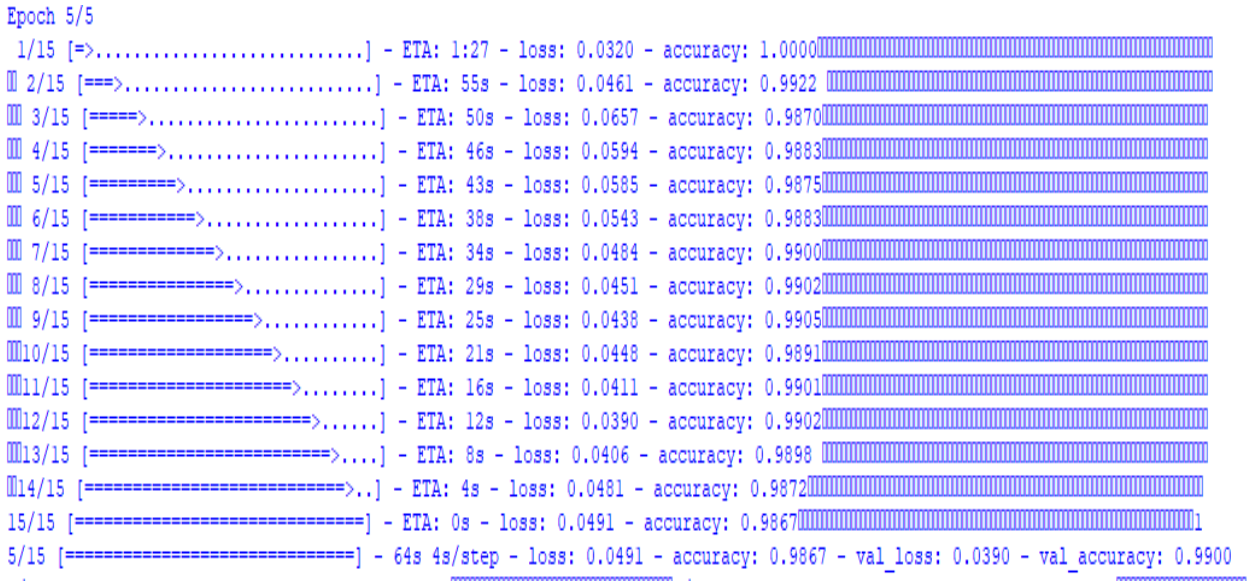


Figure 3.2.1(8)

Prediction:

The model has predicted the results accurately as attached , A speaker for example is a person Naqeeb who's voice is predicted successfully.

```
Speaker: naqeeb - Predicted: naqeeb
<IPython.lib.display.Audio object>
Speaker: rauf - Predicted: rauf
<IPython.lib.display.Audio object>
Speaker: moiz - Predicted: moiz
<IPython.lib.display.Audio object>
Speaker: howaiz - Predicted: howaiz
<IPython.lib.display.Audio object>
Speaker: naqeeb - Predicted: naqeeb
<IPython.lib.display.Audio object>
Speaker: sanaullah - Predicted: sanaullah
<IPython.lib.display.Audio object>
Speaker: naqeeb - Predicted: naqeeb
<IPython.lib.display.Audio object>
Speaker: moiz - Predicted: moiz
<IPython.lib.display.Audio object>
Speaker: moiz - Predicted: moiz
<IPython.lib.display.Audio object>
Speaker: muzammil - Predicted: muzammil
<IPython.lib.display.Audio object>
>>>
```

Figure 3.2.1(9)

Chapter 4: Software Requirement Engineering

4.1 Introduction

This chapter summarises the complete Software Requirement Specification document, including the system's purpose, scope, functional, and non-functional needs. The purpose of this article is to provide a full overview of the ACRS project, which intends to construct a suspicious speech tracking device that will automate the call tracking procedure. This paper contains the ACRS's specific requirements

4.1.1 Purpose

The goal of this chapter is to provide a full description of the software that does Suspicious Speech Tracking assisting in the automation of call tracing by agencies in order to prevent terrorist activity. This is the foundation for all software development guidelines.It will also help clients guarantee that all needs and specifications are met.

4.2 Overall Description

4.2.1 Product Functions

The following are the key properties of ACRS in this domain:

- The system keeps track of suspicious speakers.
- Users will supply phone audios to the system as input.
- ACRS will assess whether the speaker is genuine or suspicious.

A **specialized** desktop application will display the entire process

4.3 User Classes and Characteristics

The following section describes the types of users of ACRS.

4.3.1 Security Agencies

Once the prototype is in their hands, security agencies will have access to the system and will be able to enhance the dataset to meet their needs and resources.

4.3.2 Corporate sector

Using relevant datasets, many firms will be able to keep their conversations secure while protecting their privacy.

4.4 Operating Enviroment

4.4.1 Hardware

ACRS will have the following Hardware specifications:

- Rasberry Pie 4: For processing sounds as a standalone device
- High quality professional condenser microphone: In order to manually record the dataset/calls.

4.4.2 Software

ACRS will have the following Software specifications:

- Python IDE
- Anaconda

4.5 Design and implementation Constraints

- It will be capable of handling only one call at a time.
- The dataset will be confined to a few selected words and trained on those words, with the option to increase the dataset as needed.
- We'll have to meet those requirements because the system will require a lot of processing power.

4.6 User Documentation

The users will be given a user handbook with instructions on how to run ACRS as well as the limits that must be considered. Users will also have access to a project report outlining the software's features, capabilities, and procedures

4.7 Assumptions and Dependencies

- Constant electricity supply
- Users must be aware of the dataset's limits and that the system will need to be trained on a different dataset if they want it to be adaptable to their company.
- It will be necessary to consider the system's accuracy.

4.8 External Interface Requirements

4.8.1 User Interfaces

- Front-end software: UI based on Raspian OS/ Windows OS for training
- Back-end software: Python

4.8.2 Hardware Interfaces

- Windows Operating System(For training the machine)
- Rasbian OS for Rasberry Pie(For Implementation)

4.9 Communication Interfaces

Our system uses Python in the background to scan (for suspicious terms) every audio file delivered or listened to in real time, as well as a UI based on the Raspbian OS to provide realtime feedback. In order to secure the entire process and system, a login/signup option has been introduced.

4.10 System Features

ACRS will be providing following system features:

1. Signup

    - Description and Priority

To use all system functions, the user must first join up/register with ACRS It is of moderate importance.

    - Stimulus/Response Sequences

1. The software system will be launched by the user.

2. The user will then be taken to the register page.

3. To sign up for ACRS, the user will input registration information.

    - Functional Requirements

Req 1:Users must be able to visit the signup page.

Req 2: The user should be able to submit registration information into the appropriate forms.

Req 3: The system should be able to properly register new users

2. Login

    - Description and Priority

The user will first log on to ACRS to process audio files and detect suspicious calls. It is of moderate importance.

    - Stimulus/Response Sequences

1. The software system will be launched by the user.

2. The user will then be taken to the login page.

3.To access ACRS, the user must input login information.

- Functional Requirements

Req 1:Users must be able to view the login page.

Req 2: The user should be able to enter login information into the appropriate fields.

Req 3: Only authorised users should be able to access the system.

3. Receive Audio File

- Description and Priority

The ACRS feature allows the system to accept and upload various audio files for processing. It is critical since the system will be worthless without the audio file.

- Sequences of Stimulus/Response

1. The user will upload audio files that have already been captured from a memory drive.

Microsc or direct audio calls can be used as input for real-time analysis.

- Functional Specifications

Req 1: The audio file must be uploaded in a specified format (Wav.).

Req 2: Users should be able to submit audio files up to a certain size.

4. Determine whether the audio content/words are suspicious or not suspicious.

- Description and Priority

ACRS's major function or objective is to classify auditory words as suspicious or not. It is a high priority because the development of ACRS is solely for this reason.

- Sequences of Stimulus/Response

1. The audio files will be uploaded by the user.

2. The system will compare audio terms to suspicious words already saved in the dataset.

- Functional Specifications

Req1: The system should be able to match the supplied audio to the dataset that has already been provided.

Req 2: The system must be able to distinguish between suspicious and non-suspicious speakers.

5. Notify the user if there are any dubious calls.

- Priority and Description

If it detects any questionable terms in the audio file or call, ACRS will notify the user instantly.

It is a high priority since the user must be notified of questionable calls in order to take action to stop the ill/terrorist acts right away.

- Sequences of Stimulus/Response

1. Audio files will be uploaded by the user.

2. The machine will compare the audio words to the dataset's .

3. If suspicious speakers are detected, the user will receive a warning on the system.

- Requirements for Function

Req 1: The user should be able to receive appropriate alerts when questionable terms are entered.

Req 2: When the UI detects a questionable call, it should display a relevant message to the user.

6. Update the dataset/list of suspicious words

- Priority and Description

ACRS will be updated for new suspicious terms by the user. The priority is low since it is dependent on the user's needs and is not the most important criterion.

- Sequences of Stimulus/Response

1. More suspicious words will be added to the dataset by the user.

2. After that, the user will train the system to recognise new suspicious terms.

3. The user will put the system to the test by listening to the audio files for new suspicious terms.

- Requirements for Function

Req 1: Users should be able to add new suspicious words to the dataset with ease.

Req 2: The user should be able to test the system for new suspicious speaker**s**.

7. Sign out

- Priority and Description

When not in use, the user will logout of ACRS to prevent unwanted access. Because logging out is dependent on the needs of the user, it is a low priority.

- Sequences of Stimulus/Response

1. The user selects the logout option.

- Functional Specifications

Req 1: Users should have access to the logout button.

Req 2: The user should be able to logout without losing any data.

4.11Non-functional Requirements

4.11.1 Requirements for Performance

- The software's response time must not be very long, and the application must run on operating systems with high processing power.
- The system's accuracy should be sufficient to make it a trusted system.

4.11.2 Requirements for Safety

- Any user's information must be handled carefully by the application.
- Users must register with accurate information so that if an error arises, the service can assist him as much as possible.
- User credentials and personal information are not to be shared with other users.

4.11.3 Requirements for Security

- Only authorized users have access to the system, which prevents unauthorized alterations.
- Unauthorized persons shall not have access to the system.

4.11.4Attributes of Software Quality

- Availability: If the user meets the hardware and software requirements, he can access the software whenever he wants.
- Maintainability: New suspicious terms should be added to the dataset to keep it up to date.
- Reusability: The system's components must be written in such a way that they can be reused.

Chapter 5

Analysis

ACRS was created with the goal of making automatic speech recognition for suspicious speech tracking more user-friendly. At the same time, accuracy must be considered in such systems because it is of paramount importance. To achieve the best possible accuracy, a rigorous quality and accuracy check was performed on our dataset on a regular basis. The training and testing sections of the dataset were split into two pieces. After extracting features with the MFCC approach, machine-learning algorithms were applied to the data.

Chapter 6

Future Work

ACRS can be further developed to automate speech recognition on a big scale. To reach as many individuals as possible, we will endeavour to provide more data to our project. The technology could attain greater accuracy with more data. Furthermore, we must cover all available datasets of our desired folks in order to rely on it as an accurate gadget. This database will be shared with researchers who desire to work on any such system for the greater good. We've developed a system application so far, but we'll strive to convert it to a standalone device so that security agencies may deploy it quickly.

6.1 Increasing Accuracy

Our algorithm now uses 2500 words from 10 distinct speakers. Our system's accuracy is based on this CNN-based approach. However, more data is required to improve its accuracy. By expanding the number of speakers per word, we will be able to improve its accuracy.

6.2 Enhancing Dataset

We created a prototype of our concept using a 2500-word dataset. However, in order for it to function in a real-time context, the greater number of voice samples from various individuals must be covered. We'll concentrate on covering all of the individuals in the organisation and establishing a dataset for them.

6.3 Standalone Device

Currently, the ACRS interface is built in PyQt5, making it a very user-friendly application. Because we intend to use this application primarily for security agencies, a standalone device will be preferred for direct deployment to their systems. We'll work on making a proper system using Raspberry Pi 4 to make it a standalone device.

Conclusion

The project "An Automated Call Retrieval System" is a prototype for automating the process of call tracking in order to detect any suspicious talk among terrorists. It will be an efficient way to complete the task while also saving manpower, making it very cost effective. Our work is distinct in that it is not language dependent. According to our best knowledge, it was a novel concept that was one of a kind. This database will be shared with researchers who want to work on any such system for the greater good. Our product has a very user-friendly interface and a quick response time. The ACRS system can be used in a variety of security agencies, including the police, the intelligence bureau, military intelligence, and the ISI, by simply adding a dataset that corresponds to the job requirements.

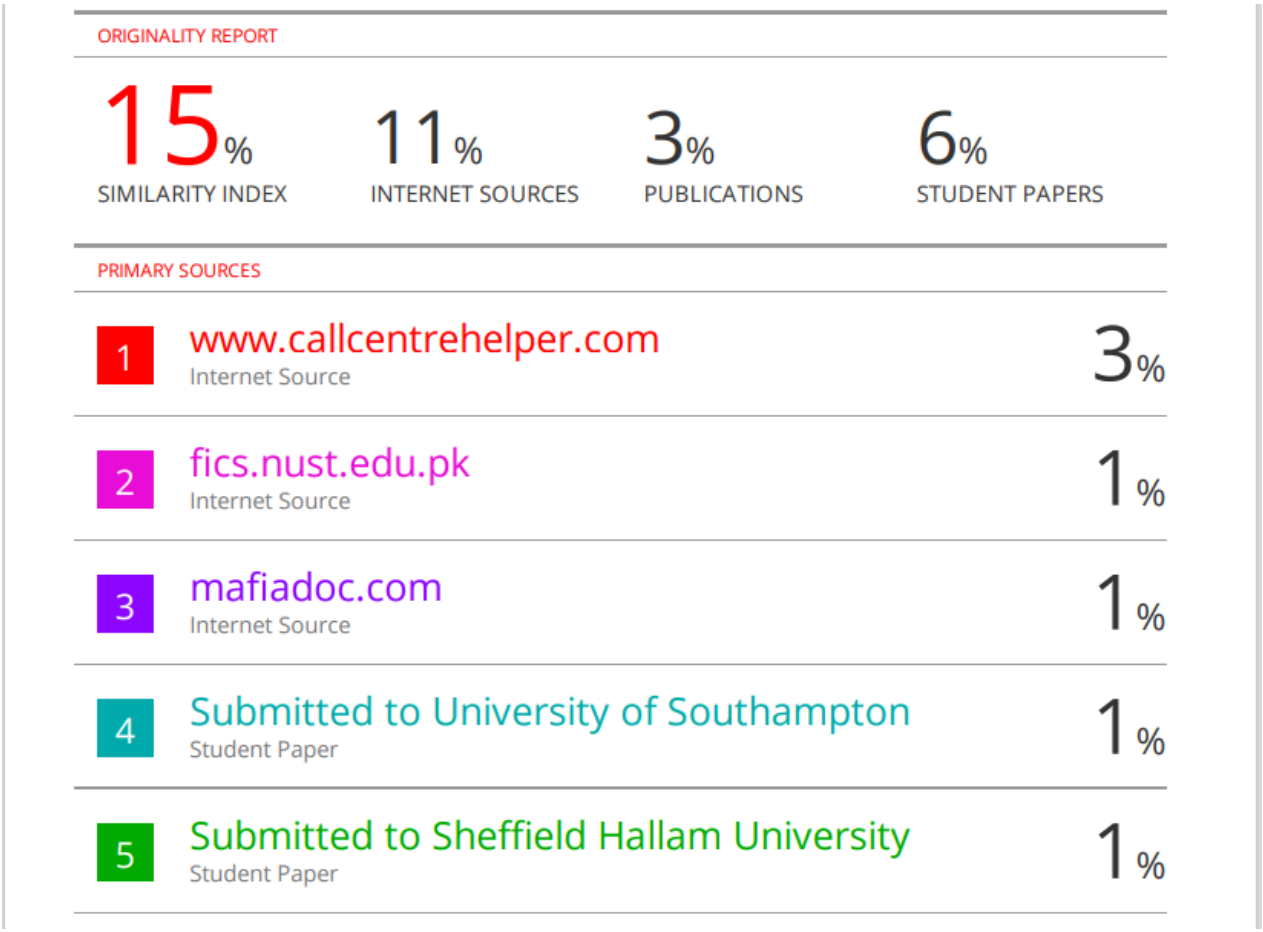
References

1. [1] Dong Yu and Li Deng. Automatic Speech Recognition. Springer, 2016.
2. [2] Zhongxin Bai and Xiao-Lei Zhang. Speaker recognition based on deep learning: An overview. Neural Networks, 2021.
3. [3] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In 2015 IEEE international conference on acoustics, speech and signal processing (ICASSP), pages 5206–5210. IEEE, 2015.
4. [4] Dong Wang and Xuewei Zhang. Thchs-30: A free chinese speech corpus. arXiv preprint arXiv:1512.01882, 2015.
5. [5] Hui Bu, Jiayu Du, Xingyu Na, Bengu Wu, and Hao Zheng. Aishell-1: An open-source mandarin speech corpus and a speech recognition baseline. In 2017 20th Conference of the Oriental Chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (O-COCOSDA), pages 1–5. IEEE, 2017.
6. [6] Jiayu Du, Xingyu Na, Xuechen Liu, and Hui Bu. Aishell-2: Transforming mandarin asr research into industrial scale. arXiv preprint arXiv:1808.10583, 2018.
7. [7] Guoguo Chen, Shuzhou Chai, Guanbo Wang, Jiayu Du, Wei-Qiang Zhang, Chao Weng, Dan Su, Daniel Povey, Jan Trmal, Junbo Zhang, et al. Gigaspeech: An evolving, multi-domain asr corpus with 10,000 hours of transcribed audio. arXiv preprint arXiv:2106.06909, 2021.
8. [8] Daniel Galvez, Greg Diamos, Juan Manuel Ciro Torres, Keith Achorn, Anjali Gopi, David Kanter, Max Lam, Mark Mazumder, and Vijay Janapa Reddi. The people's speech: A large-scale diverse english speech recognition dataset for commercial usage. In Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1), 2021. 1031
9. [9] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. Voxceleb: a large-scale speaker identification dataset. arXiv preprint arXiv:1706.08612, 2017.
10. [10] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. Voxceleb2: Deep speaker recognition. arXiv preprint arXiv:1806.05622, 2018.
11. [11] Yue Fan, JW Kang, LT Li, KC Li, HL Chen, ST Cheng, PY Zhang, ZY Zhou, YQ Cai, and Dong Wang. Cn-celeb: a challenging chinese speaker recognition dataset. In ICASSP 2020- 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 7604–7608. IEEE, 2020.
12. [12] Yi Liu, Pascale Fung, Yongsheng Yang, Christopher Cieri, Shudong Huang, and David Graff. Hkust/mts: A very large scale mandarin telephone speech corpus. In International Symposium on Chinese Spoken Language Processing, pages 724–735. Springer, 2006.
13. [13] Tingwei Guo, Cheng Wen, Dongwei Jiang, Ne Luo, Ruixiong Zhang, Shuaijiang Zhao, Wubo Li, Cheng Gong, Wei Zou, Kun Han, et al. Didispeech: A large scale

mandarin speech corpus. In ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 6968–6972. IEEE, 2021.

14. [14] Chinese Academy of Social Sciences and City University of Hong Kong. Language Atlas of China. The Commercial Press, 2012.
15. [15] Jeffrey Weng. What is mandarin? the social project of language standardization in early republican china. The Journal of Asian Studies, 77(3):611–633, 2018.
16. [16] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al. The kaldi speech recognition toolkit. In IEEE 2011 workshop on automatic speech recognition and understanding, number CONF. IEEE Signal Processing Society, 2011.
17. [17] Liang Xu, Hai Hu, Xuanwei Zhang, Lu Li, Chenjie Cao, Yudong Li, Yechen Xu, Kai Sun, Dian Yu, Cong Yu, Yin Tian, Qianqian Dong, Weitang Liu, Bo Shi, Yiming Cui, Junyi Li, Jun Zeng, Rongzhao Wang, Weijian Xie, Yanting Li, Yina Patterson, Zuoyu Tian, Yiwen Zhang, He Zhou, Shaoweihua Liu, Zhe Zhao, Qipeng Zhao, Cong Yue, Xinrui Zhang, Zhengliang Yang, Kyle Richardson, and Zhenzhong Lan. CLUE: A Chinese language understanding evaluation benchmark. In Proceedings of the 28th International Conference on Computational Linguistics, pages 4762–4772, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics.
18. [18] Seyed Omid Sadjadi, Timothee Kheyrkhah, Audrey Tong, Craig S Greenberg, Douglas A Reynolds, Elliot Singer, Lisa P Mason, Jaime Hernandez-Cordero, et al. The 2017 nist language recognition evaluation. In Odyssey, pages 82–89, 2018.
19. [19] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. Journal of machine learning research, 9(11), 2008.
20. [20] Shengkui Zhao, Hao Wang, Trung Hieu Nguyen, and Bin Ma. Towards natural and controllable cross-lingual voice conversion based on neural tts model and phonetic posteriorgram. In ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 5969–5973. IEEE, 2021.

APPENDIX A



ORIGINALITY REPORT

15%	11%	3%	6%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.callcentrehelper.com Internet Source	3%
2	fics.nust.edu.pk Internet Source	1%
3	mafiadoc.com Internet Source	1%
4	Submitted to University of Southampton Student Paper	1%
5	Submitted to Sheffield Hallam University Student Paper	1%

6	Submitted to University of Strathclyde Student Paper	1 %
7	www.investopedia.com Internet Source	1 %
8	Submitted to University of Hertfordshire Student Paper	1 %
9	Submitted to The Chartered Insurance Institute Student Paper	1 %
10	www.researchgate.net Internet Source	1 %
11	dl.acm.org Internet Source	1 %

12	Gerard Salton. "The evaluation of automatic retrieval procedures— selected test results using the SMART system", American Documentation, 07/1965 Publication	<1 %
13	Haibo Yu, Tsunenori Mine, Makoto Amamiya. "An architecture for personal semantic web information retrieval system", Special interest tracks and posters of the 14th international conference on World Wide Web - WWW '05, 2005 Publication	<1 %
14	umpir.ump.edu.my Internet Source	<1 %
15	Submitted to colorado-technical-university Student Paper	<1 %