CONTRIVER®, Mysore

Department of product designing and development

INTERNSHIP TRAINING REPORT

Submitted in partial fulfilment of the requirements for the certification of

30 days internship training program

SUBMITTED BY

SHASHANK GOWDA R
(1DB20CS098)

Under the Guidance of

Ms. Afreen
Bachelor of Engineering,
web development trainer

Department of Programming and Development

M/S CONTRIVER®

402 B, Mysore Rd, Opp Gopalan Mall, Rajarajeshwari Nagar,
Bengaluru, 560039,
Karnataka, India

2023 - 2024

CONTRIVER®

#127/1, Chamalapura Street, Nanjangud, Mysore 571301.

Department of Programming and Development

TRAINING CERTIFICATE

This is to certify that ***Sri. SHASHANK GOWDA R(1DB20CS098).*** *bonafide students of* ***DON BOSCO INSTITUTE OF TECHNOLOGY*** *in partial fulfillment for the "Training Certificate" award in* the ***Department of Programming and Development*** of the ***CONTRIVER, Bangalore*** *during the year* 2023-2024. *It is certified that he has undergone internship during the time period from 14/08/2023 to 16/09/2023 of all working days corrections/suggestions indicated for internal validation have been incorporated in the report deposited to the guide and trainer. The training report has been approved as it satisfies the organizational requirements with respect to the Internship training prescribed for the said qualification.*

Ms. Afreen **Bachelor of Engineering.** **Web Development Trainer**	**Sangeeta A Kambali** **B.E.** **Sr. Design Engineer, Chief Assistant data analyst**	**Shri. SANJAY B** **DMT, B.E.** **Sr. Production Head and Chief Executive Officer**

ACKNOWLEDGEMENT

It is our privilege to express gratitude to all those who inspired us and guided us to complete the internship training program. This work has remained incomplete without the direct and indirect help of many people who have guided us in the success of this internship. We are grateful to them.

We would like to express our sincere gratitude to Contriver for giving us the chance to complete our internship with the company. We would like to thank Mr. Sanjay B for providing us with excellent supervision during our internship. Throughout the training sessions, his encouragement and support were vital. What should have been difficult chores were made into an invaluable learning experience under Mr. Sanjay B's direction. His perceptive viewpoints made a substantial contribution to our successful completion of the required assignment on schedule. He has tremendously inspired us throughout our internship, and we are incredibly appreciative of his outstanding leadership.

As our supervisors and mentors, Ms. Sangeeta K and Ms. Afreen, we also like to recognize and appreciate their contributions. Their perseverance and commitment to guiding us were crucial to our growth. We particularly appreciate their method of simplifying difficult work so that we could understand them and then gradually hone our talents. They gave us difficult homework, which helped us learn more and gain more self-assurance. We sincerely thank them for their invaluable assistance and guidance throughout our training sessions.

Date:

Place: Bangalore

- SHASHANK GOWDA R

RESUME

SHASHANK GOWDA R
COMPUTER SCIENCE AND ENGINEER

CONTACT INFORMATION

ADDRESS:
S/O L Raja,
#11 Hegganahalli Cross,
Sunkadakatte,
Bangalore - 560091.
EMAIL ID : shashankgowdar02@gmail.com
CONTACT NO : 6361296303

OBJECTIVE

I am passionate about exploring new horizons, constantly learning and adapting to emerging technologies, and eagerly embracing challenges. My goal is to stay at the forefront of innovation and knowledge, actively seeking opportunities that allow me to expand my skill set and tackle complex problems. By doing so, I aim to contribute meaningfully to any organization I work with, ultimately driving progress and success in a rapidly evolving professional landscape.

ACADEMIC INFORMATION

EDUCATION QUALIFICATIONS:

COURSE/EXAM	INSTITUTION	YEAR OF PASSING	MARKS OBTAINED IN %
B.E. in CSE	*Don Bosco Institute of Technology, Bangalore.*	*2024*	*90.6*
Pre-University	*Mangalore Independent PU College, Bangalore.*	*2020*	*90.5*
S.S.L.C	*Apollo High School, Bangalore*	*2018*	*95.52*

TRAINING/ INTERNSHIP

INTERNSHIP	*Contriver*

COMPUTER SKILLS

- Packages : MS Office, MS PowerPoint, MS Excel
- Engineering Tools : Python, C/C++, HTML, CSS, JS, ReactJS, Flask, IoT,Java.

PROJECT DETAILS

ENGINEERING PROJECT: Speech Emotion Recognition.

Abstract: This project introduces a deep learning-based Speech Emotion Recognition system employing RNN with LSTM, effectively capturing emotional cues from spoken language. The system's applications span human-computer interaction, mental health monitoring, education, and more, promising a significant advancement in emotionally aware technology.

PERSONAL STRENGTH

- Hardworking, dedicated, responsible, self confident.

PERSONAL PROFILE

Name	: Shashank Gowda R
Father's name	: L Raja
DOB	: 02/06/2002
Marital Status	: single
Nationality	: Indian
Languages Known	: English, Kannada, Hindi.
Personal address	: S/O L Raja, #11 Hegganahalli Cross, Sunkadakatte Bangalore - 560091.

DECLARATION

I hereby declare that all the information is correct and true to the best of my knowledge and belief.

DATE: Yours Sincerely,

Place: Bangalore

(SHASHANK GOWDA R)

TAKEAWAY TOPICS FROM TRAINING

I had an opportunity to immerse myself in all facets of machine learning throughout my internship at Contrived, which spanned from August 14, 2023. The workshop provided pupils with a solid grounding in Python, with a focus on machine learning. The topics that followed were the main topics to take away from this internship:

➢ Python Fundamentals

My internship at Contriver started out with a thorough dive into the fundamentals of Python programming. In creating a solid foundation for my foray into the fields of data science and machine learning, this phase was crucial. The Python Basics module gave me a thorough understanding of Python's practical uses in addition to introducing me to its grammar and structure.

Important Elements of Python Basics Module:

Variables and Data Types: A thorough examination of Python variables and data types opened the module. I learned how to declare and use variables to hold several kinds of data, including texts, integers, and floating-point numbers. It is essential to understand data types since they serve as the foundation for data modification and analysis.

Control Structures: Control structures were presented and thoroughly rehearsed. These included conditional statements (if, elif, else) and loops (for and while). These building blocks enable decisions and repeats, which are essential for resolving complicated problems, and allow for the logical flow of a Python program.

Functions: The idea of functions, a fundamental building element in Python, was explained in great detail. Along with learning how to design and use functions, modular programming's importance was also made clear to me. Functions make complex programs manageable by enabling code reuse and organization.

Practical Scripting: I had the chance to use my skills in actual situations throughout the Python Basics module. Writing Python scripts to carry out numerous tasks was a hurdle I faced during practical exercises and assignments. Basic processes like data input/output and mathematical calculations were among these jobs, as well as more complex ones like processing files and manipulating data.

The Python Basics Module's Importance

This challenging course was essential in fostering a sense of assurance and mastery of Python programming. The practical application-focused, hands-on approach was particularly successful in bridging the gap between theory and practice. By the end of this phase, writing Python scripts to solve issues, modify data, and automate processes had become second nature.

Furthermore, a solid grasp of Python's foundational concepts served as the basis for the training's succeeding courses. This core knowledge continually served as a reference point for understanding and applying increasingly complicated data science and machine learning approaches as I went into more difficult ideas.

In conclusion, the Python Basics module served as the foundation for my internship experience and gave me the fundamental knowledge and abilities I needed to navigate Python programming. This foundation was crucial to my complete internship experience at Contriver since it enabled me to expand on my expertise in later training sessions.

➢ Basics of Numpy and Pandas

The training at Contrived proceeded with a vital focus on two crucial Python libraries: NumPy and Pandas, building on the solid foundation of Python basics. In the field of machine learning, these libraries serve as the cornerstone for data manipulation and analysis. My grasp of these libraries and their practical use in streamlining data processing and converting unstructured data into structured representations has deepened thanks to the "Numpy and Pandas Basics" session.

The Numpy and Pandas Basics Module's

An extensive introduction to NumPy, the Numerical Python library, was given at the beginning of the module. I gained knowledge of NumPy arrays, which form the foundation of my library. These arrays are crucial data structures in data science because they make it possible to store and manipulate enormous datasets efficiently. By learning about ideas like array generation, indexing, slicing, and reshaping, I was able to work with data in a more structured and effective way.

Data Manipulation with Pandas: Pandas, a flexible data manipulation library, was introduced as an important subject for the following section. I gained knowledge of working with Data Frames and Pandas Series, two crucial data structures for data analysis. Data cleansing, data transformation, filtering, and aggregation were just a few of the different data manipulation processes addressed in the subject. Gaining useful abilities for real-world data work required a lot of practical experience.

Preprocessing and Data Cleaning: The significance of high-quality data was emphasized heavily. I gained knowledge on how to spot and manage outliers, duplicate records, and missing data. To ensure that data was in the best possible condition for analysis and modeling, methods for data pretreatment, such as data normalization and standardization, were investigated.

Introduction to AIML (Artificial Intelligence and Machine Learning)

A thorough introduction to artificial intelligence and machine learning was given in the AIML curriculum. The fundamental principles and methods employed in the creation of intelligent systems were clarified during this training session. The differences between artificial intelligence and machine learning, supervised and unsupervised learning, and the ethics of AI were among the key ideas. My understanding of the ramifications and uses of AI in numerous fields has been widened by this information.

Regression and Classifiers: I studied classifiers and regression methods within the field of machine learning during the training's final stage. Supervised learning, classification algorithms (such as Decision Trees), and regression analysis (such as Linear Regression, Random Forests) were all topics covered. I gained practical expertise in model training, evaluation, and selection through hands-on exercises and projects. My path to mastering machine learning required me to fully grasp the subtleties of categorization and regression.

HTML (Hyper Text Markup Language)

The core of web development, HTML, was covered in-depth during the internship. I now possess the knowledge and abilities I need to produce web content that is both structured and semantically meaningful as a result of this lesson. The main ideas to remember from this section were:

Markup Structure: I discovered how to organize web material using headers, paragraphs, lists, and links in HTML. For material to be ordered and readable, it is essential to understand how a web page is put together.

Form and Input Element: The topic of the module was the development of online forms and input components, which allowed for user data to be gathered. I improved my ability to make submit buttons, radio buttons, checkboxes, and input fields.

Navigation and Hyperlinks: I mastered the art of building linkages between websites. Effective website navigation requires having this information.

CSS (Cascading Style Sheet)

By emphasizing the styling and display of web information, the CSS module enhanced the HTML instruction. I learned how to create web pages that are both aesthetically pleasing and user-friendly thanks to this segment:

Layout and Positioning: The subject went in-depth on how to position items on web pages and create responsive layouts. This information is essential for creating web pages that adjust to various screen sizes and devices.

CSS Selectors: I became proficient at utilizing CSS selectors to style only certain elements. This gives designers of web pages detailed control.

JavaScript

Another crucial component of the training was JavaScript, a dynamic scripting language that gives web pages interactivity and capability. What you should remember most from the JavaScript module was:

Variables and Data Types: Through this course, I developed a thorough understanding of JavaScript variables and data types, which are crucial for scripting and manipulating data.

Control Structures: This subject addressed control structures, such as conditionals (if statements) and loops (for and while), which allow web applications to make dynamic decisions and perform repetitive activities.

In conclusion, my internship at Contriver gave me a strong foundation in Python programming, experience using NumPy and Pandas to manipulate data, knowledge of machine learning and artificial intelligence, and experience with classifiers and regression. And also HTML, CSS, and JavaScript into the internship training broadened my skill set and reinforced my understanding of web development. These topics are foundational to the creation of dynamic, responsive, and visually appealing web applications. I now have the knowledge and abilities required to succeed in the fields of data science and machine learning thanks to these key takeaways. I am appreciative of the mentorship I received and the priceless experience I gained during my internship.

TAKEAWAY TOPICS FROM THE GUEST LECTURER

I had the pleasure of attending a guest lecture by Shri. SANJAY B. DMT, B.E., who is a seasoned expert in the field of web development and served as the Senior Production Head and Chief Executive Officer, during the duration of my internship at Contrived. I gained a deeper grasp of web development, with a concentration on WordPress and website construction, thanks to Shri. SANJAY B's insights and knowledge.

Understanding the Basics of Websites and Webpages

The lecture began with a thorough introduction to webpages and the foundational ideas of websites by Shri. SANJAY B. He highlighted that webpages are the foundation of the internet and went into detail on their creation, upkeep, and storage on servers. The following crucial ideas were discussed:

Webpage Creation: Shri. SANJAY B explained the process of developing websites, emphasizing the significance of HTML, CSS, and JavaScript as the main languages used in website building. He explained how these tools combine to organize, style, and interactively enhance content.

Storage of Web Pages: The presentation gave information about how Web pages are kept on web servers. This included topics like server setups, web hosting, and the function of content management systems (CMS) like WordPress in handling website content.

Webpage Access: Shri. Sanjay B described how users access webpages using web browsers. He went into detail about how HTTP/HTTPS protocols and URLs (Uniform Resource Locators) work to fetch web content from servers to users' devices.

Domain Administration: The definition of domains and domain name systems (DNS) was clarified. Shri. SANJAY B addressed the importance of picking a suitable domain name and how it affects a website's usability and branding.

WordPress from the Ground Up

The guest lecture included a substantial section on WordPress, a well-liked and adaptable CMS used for website construction. In order to learn and use WordPress, from its installation to creating a fully functional website, Shri. SANJAY B gave a step-by-step tutorial.

WordPress Installation: Shri. SANJAY B led us through the procedure while emphasizing the significance of a trustworthy hosting platform and database configuration.

Web design that is responsive: Shri. SANJAY B emphasized the significance of responsive web design. He clarified methods for ensuring that websites effortlessly adjust to different screen sizes and devices, improving the user experience.

Creating the Home Page of the Contriver Website

The final activity of the lecture was to build the homepage of the Contriver website. In a hands-on session, Shri. SANJAY B explained how to organize material, add images, text, and multimedia components, and set up the layout to build a captivating and responsive site.

SEO (search engine optimization)

In addition to providing in-depth knowledge of WordPress and web development, Shri. SANJAY B. DMT also devoted a significant amount of the guest lecture to the important subject of Search Engine Optimization (SEO). In order to increase a website's exposure and accessibility on search engines, SEO is a crucial component of web development and digital marketing.

Finally, the guest lecture by Shri. SANJAY B was a rewarding experience that expanded my knowledge of WordPress and web development. It gave a strong foundation for building websites, responsive design, and the rules of good web administration. The hands-on experience I received while creating the homepage for the Contriver website was not only instructive but also directly related to my internship work. These insights will surely aid in my development as a multifaceted web developer and contributor to the technology industry.

FEEDBACK/OPINION OF THE INTERNSHIP

Innovative topics/Methods:

The Contriver internship program offered a novel perspective through exploring cutting-edge subjects and techniques, considerably enhancing my educational experience. The study of Python, NumPy, Pandas, WordPress site development, and Search Engine Optimization (SEO) provided a thorough understanding of the technical world. My understanding was firmly established thanks to the practical tasks that were included along with the theoretical lessons. This participatory method promoted creativity and demanded participation in the subject matter.

The industrial significance of the topics:

The emphasis on the industrial relevance of the subjects covered during this internship was one of its standout features. In order to be in line with practical applications and market demands, the training courses were carefully chosen. This practical focus was really helpful since it made it clear to me how what I was learning directly related to methods used in the industry. I am well-equipped to contribute meaningfully in a professional setting thanks to the directly transferrable knowledge and skills I have obtained.

Syllabus/Concepts that can be included/recommended in engineering curriculum (Academics):

The internship program was extensive and interesting, but there is room for growth, specifically when it comes to the curriculum's inclusion of deeper discussions of machine learning (ML) and deep learning (DL). An extended focus on ML and DL for engineering students could be very advantageous given the growing importance of these topics in numerous businesses. I suggest incorporating cutting-edge ML and DL ideas into the academic curriculum to better train aspiring engineers for the rapidly changing technological environment.

Area of improvements/Drawbacks in the internship program:

Two crucial areas where the internship program could be improved are:

- Integration of ML and DL: It is strongly advised to add more complex Machine Learning and Deep Learning topics to the curriculum. Giving interns a deeper understanding of these subjects would be beneficial because they are rapidly changing and have major industrial importance.

- Timing of Sessions: Extensive discussion of the topics might be possible with longer sessions. Longer sessions would allow for more time for practical activities, conversations, and practical projects, encouraging a deeper degree of understanding and creativity.

Opinion of the internship:

I had a really favorable overall impression of my internship at Contrived. The curriculum promoted a warm, welcoming environment that promoted participation in class and social interaction. The guest lecturer's knowledge and the mentors' availability were both exceptional. The topics covered by the guest lecturer, particularly those concerning web development and SEO, were great additions to the curriculum and improved the internship's all-encompassing nature. Along with enhancing my expertise, the program has motivated me to learn more about data science and technology.

CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

1.1 Introduction

The goal of the branch of study known as Speech Emotion Recognition (SER) is to identify and analyze the emotional content of human speech. Speech-based emotions, such as happiness, sadness, anger, fear, disgust, surprise, and neutrality, are essential for human interaction and communication. Numerous practical applications, such as in human-computer interaction, customer service, mental health diagnostics, and entertainment, can be made by comprehending and automating the recognition of these emotions.

Traditionally, feature engineering and machine learning algorithms that were created by hand were used in SER. However recent developments in deep learning, especially Recurrent Neural Networks (RNNs) like Long Short-Term Memory (LSTM) networks, have completely changed the area. Deep learning models are ideally suited for SER tasks because they automatically learn essential features from unprocessed audio input.

In this project, we investigate how deep learning, and more specifically, an LSTM-based model, can be used to recognize speech emotions. Creating a model that can precisely categorize audio recordings into predefined emotion categories is the aim. We use a dataset of audio samples that have been labeled with emotional states to accomplish this. Our goal is to use deep learning to identify subtle speech patterns and nuances that are indicative of various emotions.

By the project's conclusion, we hope to show how deep learning works in the SER space and contribute to the creation of systems that can recognize and react to emotional speech through analysis. The sections that follow will give a thorough rundown of our dataset, approach, findings, and discussions surrounding them.

1.2 Motivation

One of the most fundamental and adaptable forms of human communication is speech. In addition to words and information, it also carries a complex tapestry of feelings, intentions, and

nuance. It's not just fascinating to try to identify and comprehend these emotional cues in speech; it also has a lot of practical relevance and potential. Some of the main reasons for this:

- Human-computer interaction

  A greater need for natural and emotionally intelligent human-computer interfaces is emerging in an increasingly digital world. Chatbots and virtual assistants that are emotionally intelligent can engage users in more interesting and sympathetic interactions.

- Enhancing client relations

  Analyzing customer interactions is essential for determining issues and understanding customer satisfaction in sectors like customer service and call centers. Speech emotion recognition can assist in automating the process of determining callers' emotional states, leading to quicker problem-solving and improved client experiences.

- The improvement of mental health diagnosis and monitoring:

  Emotions play a crucial role in mental health, and keeping track of patients' emotional states can help with early diagnosis and treatment. The use of automated speech analysis can help mental health professionals by giving them more information about a patient's emotional health.

- Personalized entertainment and content:

  Real-time comprehension of audience emotions can help in the creation of entertainment and content by enabling the personalization of that content. For instance, altering a video game's plot or a playlist's music choices in response to the player's or listener's emotional state.

- Interaction between humans and machines in autonomous systems

  Recognizing the emotional state of users or passengers in autonomous vehicles, robots, and other AI-driven systems can improve safety and user experience. Based on the recognized emotions, autonomous systems can modify their actions or reactions.

1.3 PROBLEM STATEMENT

Going beyond conventional approaches and utilizing cutting-edge technologies is crucial for enhancing security measures in a variety of security applications, including surveillance, access control, and authentication. Through an analysis of people's spoken words, emotion detection from speech has the potential to provide an additional layer of security.

The main goal of this project is to create an effective emotion detection system that analyzes speech to determine people's emotional states. The system aims to enhance security measures and decision-making in various contexts by understanding and recognizing emotions expressed through speech.

1.4 Objective

The main goal of the project is to develop and put into use a state-of-the-art Speech Emotion Recognition (SER) system using cutting-edge deep learning methods, notably Recurrent Neural Networks (RNN) with Long Short-Term Memory (LSTM) units. The system aspires to the following objectives:

Emotion Recognition: The main goal of the research is to create a system that can reliably identify and categorize human emotions as they are presented in spoken language. The system will be able to recognize emotions like joy, sadness, anger, fear, and more by examining the acoustic characteristics, prosody, and grammatical content of speech.

Real-time Processing: The project's main goal is to make it possible for the system to recognize emotions in real-time and give users immediate feedback based on those states during conversations or other interactions. For applications like virtual assistants, customer support, and educational platforms, this real-time functionality is essential.

Deep Learning Approach: The system's foundation will be deep learning, specifically the use of RNNs with LSTM architecture. In order to comprehend the dynamic nature of emotional expression in language, LSTM networks are well-suited for collecting temporal relationships in speech data.

Enhanced User Experience: The technology being developed aims to radically improve the user experience in a variety of settings. It can produce more sympathetic and aware-of-context interactions by identifying and responding to user emotions. This is particularly useful in circumstances involving human-computer interaction when user engagement and happiness are crucial.

Applications across a Wide Range: The project recognizes the numerous uses for voice emotion recognition. These applications range from giving emotional support in mental health monitoring to improving customer service interactions by adjusting replies to the client's emotional state. One important component of the project's goal is the system's adaptability.

Research Contribution: The project seeks to significantly advance the fields of emotional computing and deep learning in addition to its practical applications. It aims to improve our comprehension of speech data and emotion recognition, opening the door for further study and advancement.

1.5 Proposed Solution

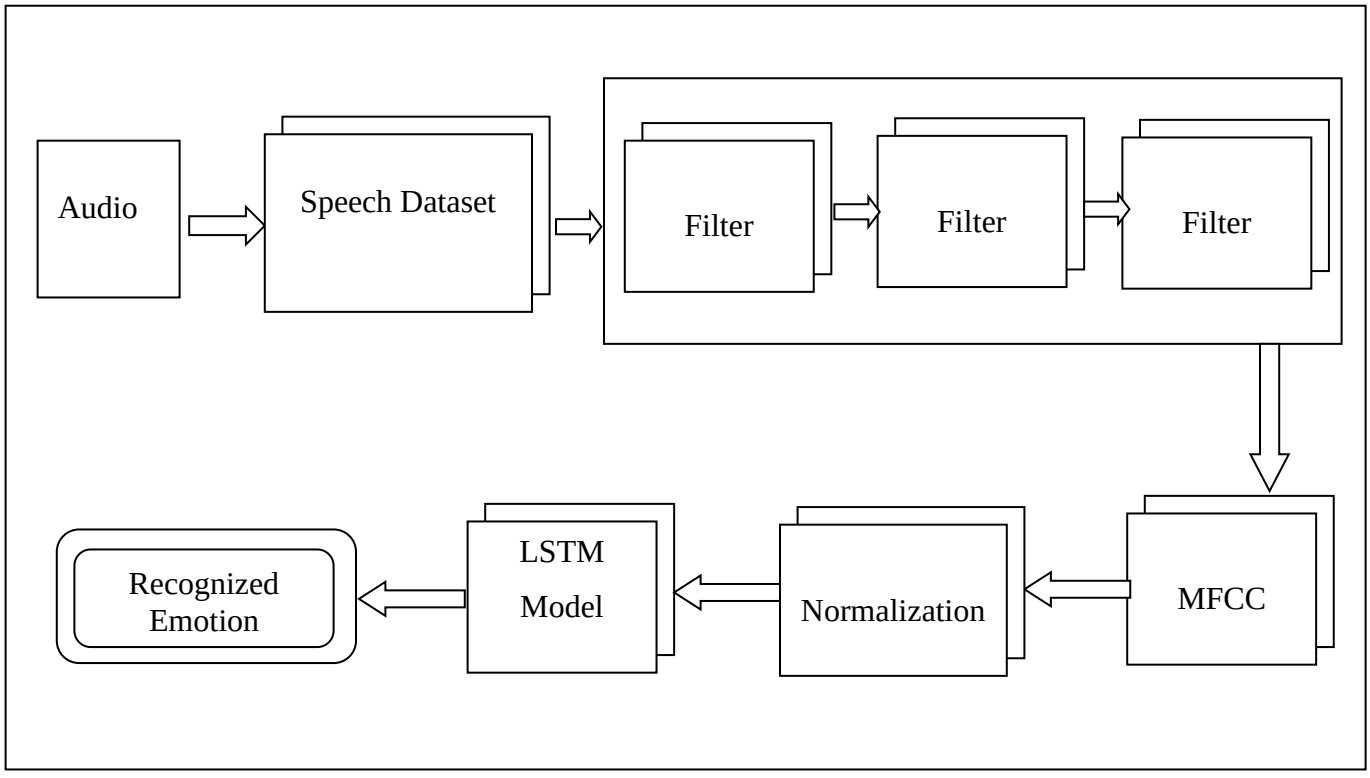

Fig:1.1 Block Diagram of Proposed Solution

CHAPTER 2
LITERATURE SURVEY

[1] Mohammad Amaz Uddin, Mohammad Salah Uddin Chowdury, Mayeen Uddin Khandaker, Nissren Tamam and Abdelmoneim Sulieman.

Human speech indirectly represents the mental state or emotion of others. The use of Artificial Intelligence (AI)-based techniques may bring revolution in this modern era by recognizing emotion from speech. In this study, we introduced a robust method for emotion recognition from human speech using a well-performed preprocessing technique together with the deep learning-based mixed model consisting of Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN). About 2800 audio files were extracted from the Toronto Emotional Speech set (TESS) database for this study. A high pass and Savitzky Golay Filter have been used to obtain noise-free as well as smooth audio data. A total of seven types of emotions; Angry, Disgust, Fear, Happy, Neutral, Pleasant-surprise, and Sad were used in this study. Energy, Fundamental frequency, and Mel Frequency Cepstral Coefficient (MFCC) have been used to extract the emotion features, and these features resulted in 97.5% accuracy in the mixed LSTM + CNN model. This mixed model is found to perform better than the usual state-of-the-art models in emotion recognition from speech. It also indicates that this mixed model could be effectively utilized in advanced research dealing with sound processing.

[2] Leila Kerkeni1, Youssef Serrestou, Mohamed Mbarki, Kosai Raoof, and Mohamed Ali Mahjoub

In this paper, we compare different approaches for emotion recognition tasks, and we propose an efficient solution based on a combination of these approaches. Recurrent neural network (RNN) classifier is used to classify seven emotions found in the Berlin and Spanish databases. Its performances are compared to Multivariate linear regression (MLR) and Support vector machine (SVM) classifiers. The explored features included: mel frequency cepstrum coefficients (MFCC) and modulation spectral features (MSFs). Finally results for different combinations of the features and on different databases are compared and explained. The overall experimental results reveal that the feature combination of MFCC and MS has the highest accuracy rate on both the Spanish emotional database using RNN classifier 90,05% and the Berlin emotional database using MLR 82,41%.

[3] Ruhul Amin Khalil, Edward Jones, Mohammad Inayatullah Babar, Tariq Ullah Jan, Mohammad Haseeb Zafar, and Thamer Alhussain.

Emotion recognition from speech signals is an important but challenging component of Human-Computer Interaction (HCI). In the literature on speech emotion recognition (SER), many techniques have been utilized to extract emotions from signals, including many well-established speech analysis and classification techniques. Deep Learning techniques have been recently proposed as an alternative to traditional techniques in SER. This paper presents an overview of Deep Learning techniques and discusses some recent literature where these methods are utilized for speech-based emotion recognition. The review covers databases used, emotions extracted, contributions made toward speech emotion recognition, and limitations related to it.

Chapter 3

System Requirements Specification

In particular, Recurrent Neural Networks (RNN) with Long Short-Term Memory (LSTM) architecture will be used in this paper to outline the system requirements for the construction of the Speech Emotion Recognition (SER) system.

3.1 Scope

This paper outlines the functional and non-functional requirements, restrictions, and dependencies and provides guidance for the creation, testing, and deployment of the Speech Emotion Recognition (SER) system. It acts as a thorough reference for all project participants, guaranteeing alignment with project objectives and user requirements. The article discusses limits like time and financial restrictions and emphasizes reliance on deep learning frameworks and data resources. It gives the entire project lifetime a systematic framework.

3.2 Definitions, abbreviations, and acronyms

SER (Speech Emotion Recognition): The act of identifying and classifying human emotions presented in spoken language is known as SER (Speech Emotion Recognition). To categorize emotions like happiness, sadness, anger, fear, and more in audio data, acoustic properties, and language content must be examined.

RNN (Recurrent Neural Network): Recurrent neural networks are a subset of artificial neural networks that are made to process sequential data. It is a crucial part of the SER system for collecting temporal components of emotional expression in speech since it works effectively for tasks that require dependencies over time.

Long Short-Term Memory (LSTM): LSTM is a particular kind of RNN architecture that excels at simulating long-term data dependencies. By preserving and applying information over long periods of time, LSTM units are employed in SER to efficiently record and understand emotional cues in speech data.

3.3 Functional Requirements

3.3.1 User Input

The capability of the system to accept audio data and the user's interaction with the User Interface are the functional requirements for user input:

- Acceptance of Audio Data: The SER system must be able to accept audio data in widely used and standardized file formats such as WAV (Waveform Audio File Format) and MP3. This criterion makes the system compatible with a variety of audio sources and streamlines user involvement.

- User-Initiated Data Upload: Using the User Interface, users will be able to upload audio data. This functionality, which is crucial for user engagement, is often implemented through an aspect of the user interface that is simple to use and enables users to submit audio samples for emotion recognition with ease.

3.3.2 Data Preprocessing

The Speech Emotion Recognition (SER) system's functionality and feature extraction depend critically on the data preparation requirements:

Preprocessing for Enhanced Analysis: To prepare the incoming audio data for further analysis, the system must preprocess it. During this preprocessing stage, tasks like noise reduction, resampling, and normalization are performed. These steps guarantee that the audio data is in a format that is appropriate for precise emotion recognition.

Feature Extraction: The extraction of pertinent features from the preprocessed audio data is a crucial necessity. Techniques like Mel-frequency cepstral coefficients (MFCC) extraction shall be used by the system. MFCCs are essential for identifying emotional cues in the audio since they are a tried-and-true method for collecting crucial acoustic properties in speech. The system's capacity to glean valuable information from the audio input is highlighted by this requirement.

3.3.3 Emotion Recognition

The Speech Emotion Recognition (SER) system's main functionality is highlighted by the functional requirements for emotion recognition:

Real-time emotion recognition using features acquired during the data preprocessing stage is the core function of the SER system. Utilizing a Recurrent Neural Network (RNN) with Long Short-Term Memory (LSTM) units, this task is accomplished. The RNN-LSTM model analyzes consecutive audio data and detects spoken language emotional cues. Real-time recognition makes sure that the system can respond to users in a timely manner regardless of their emotional state at any given time.

3.3.4 User Interface

To ensure a user-centric and accessible experience, the functional criteria for the user interface (UI) are crucial:

Facilitating User Interactions: The Speech Emotion Recognition (SER) system's User Interface (UI) will be the main point of contact for users. Its job is to give people an easy-to-use interface so they can interact with the system. This contains tools that make it simple for users to upload their audio data.

Displaying Emotion Predictions: The presentation of emotion predictions to users is a crucial aspect of the UI's function. The system's recognition results must be presented in an understandable manner, including the identified emotion and any associated confidence scores. This makes sure that users can readily comprehend and interpret the responses provided by the system.

User-Friendly Presentation: In order to improve the user experience, the UI must deliver emotion forecasts in an approachable and user-friendly manner. Clarity, readability, and simplicity of navigation should be given top priority in the UI's design and layout to make it easier for users to understand and engage with the system.

3.4 Non-Functional Requirements

3.4.1 Performance:

For the system to be effective and efficient, certain performance requirements are essential:

Real-Time Emotion forecasts: According to the system's performance mandate, real-time emotion forecasts must be provided. This implies that after receiving user input, the system should respond in less than one second. For systems that require quick feedback and flawless user interactions, real-time predictions are crucial.

Accuracy of 99%: The system's performance should reach an accuracy of 99% in emotion recognition in order to meet the requirements for dependable emotion recognition. By maintaining a certain level of accuracy, the system may continually provide users with high-quality emotional analysis, fostering user confidence and system dependability.

3.4.2 Scalability

The system's ability to respond to shifting demands is ensured by scalability requirements:

Support for Increasing User Demand: The system must be scalable, which means it must be able to handle growing user demand without noticeably degrading performance. The system's resources should be scaled properly as the number of users increases to maintain responsiveness and accuracy.

Adaptability to numerous Languages and Accents: The system must exhibit adaptability to numerous languages and accents in addition to meet rising user demand. This feature guarantees that the system can detect and react to various linguistic and cultural inputs in an effective manner, hence increasing its applicability and usefulness in worldwide situations.

3.5 Dataset

The Toronto Emotional Speech Set (TESS) is a well-known and painstakingly controlled dataset created for studies on speech analysis and emotion recognition. The TESS collection of emotionally expressive speech recordings includes a wide spectrum of emotions, including joy, sorrow, rage, and fear. The dataset is distinctive in that it contains a number of actors,

guaranteeing a range of accents and vocal attributes, expanding its applicability in cross-cultural and cross-linguistic investigations.

Each TESS recording is meticulously annotated, offering useful details about the intended emotional content and serving as a useful training and testing tool for emotion identification algorithms. TESS has emerged as a pillar for academics and developers looking to improve the fields of affective computing, human-computer interaction, and speech analysis because to its high-quality audio recordings and well-documented emotional context. Because of its accessibility, understanding and interpreting emotional signs in human speech has advanced significantly.

- Link for the Dataset

3.6 System Requirements

Hardware Requirements:

Processor	Multicore Processor
Hard Disk	500 GB
RAM	16 GB (To support model Training)
GPU	NVIDIA GPU with CUDA

Software Requirements:

Operating System	Windows, Linux, or MacOS
Programming Language	Python (3.11.5)
IDE	VS code, Pycharm , Jupyter Notebook,
Package and Library	numpy, pandas, os, seaborn, matplotlib, librosa, Audio, joblib, keras, streamlit, tensorflow

Chapter 4

System Analysis and Design

4.1 System Analysis

4.1.1 Requirements Gathering

This important step entails gathering and storing all specifications that govern the behavior and functionality of the system. Functional requirements are very important since they specify the unique abilities and characteristics the Speech Emotion Recognition (SER) system must have. Critical components including user input, data preparation, emotion identification, and the user interface are all included in these requirements. Non-functional requirements, which concentrate on the system's performance, scalability, security, and privacy elements, are as significant. They offer the standards for judging the system's overall efficacy and quality.

4.1.2 Data Collection and Analysis

The creation of a solid SER system depends on the collection and analysis of data. An essential component of this stage is collecting a varied and representative audio dataset. The emotion recognition model is trained using this dataset as its basis. The features of this data must be understood through data analysis. It assists in determining the essential auditory characteristics and their significance in recording emotional cues. The choice of these variables will determine how well the system can identify and categorize spoken emotions.

4.1.3 Technology Assessment

To ensure the system's technological viability, it is crucial to choose the right technologies. This step entails a thorough assessment of the chosen deep learning framework, such as TensorFlow. Additionally, selecting the right libraries is crucial, with programs like librosa being important in the processing of audio data. In addition, the optimal method for developing a user-friendly interface is evaluated based on the evaluation of UI development tools. The technology evaluation makes sure that the frameworks and tools selected are compatible with

the project's objectives and the technological requirements necessary for effective SER.

4.1.4 Architecture Design

The system's architecture is highlighted in this subsection. Data flow, the various parts of the SER system, and their interactions are all covered by the design. The architecture of the RNN-LSTM model must be defined in specific detail during this stage. For real-time emotion recognition, it specifies the number of layers, units, and activation functions. From user input to emotion prediction, a clearly defined architecture makes sure that data flows smoothly through the system, ensuring precise and effective outcomes.

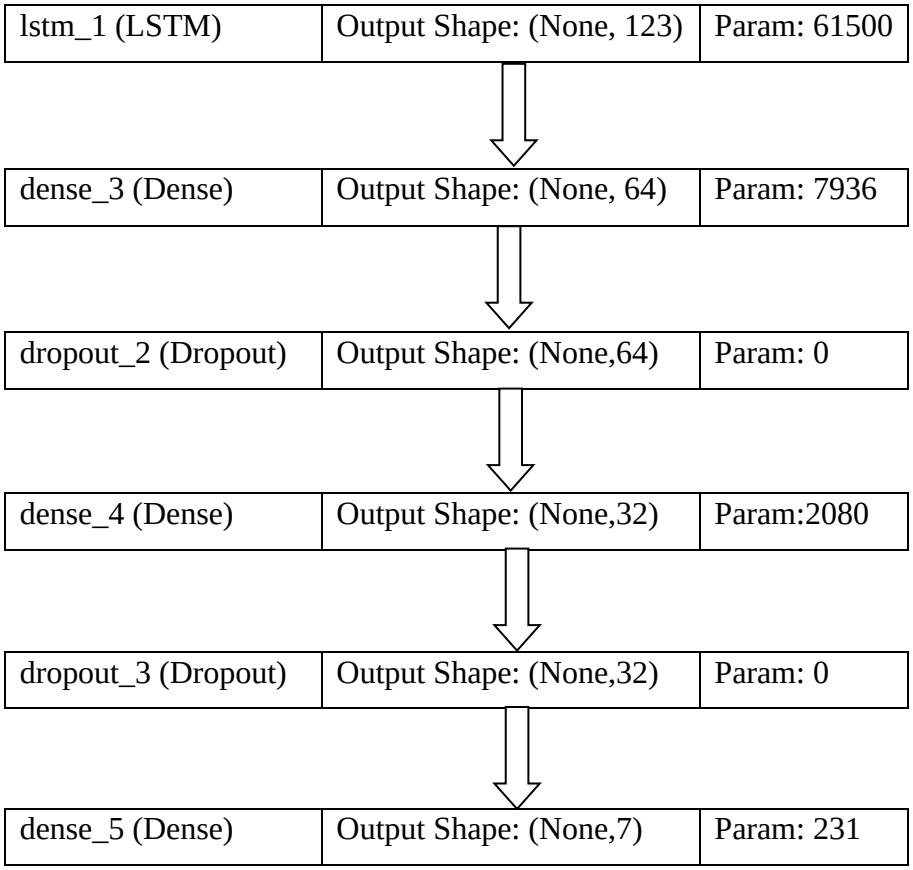

Figure 4.1: LSTM network architecture using MFCC features.

4.2 System Design

4.2.1 Data Preprocessing

In order to optimize audio data for emotion recognition, a detailed design for data preprocessing must be completed during this phase:

Enhancement Steps: The plan outlines a sequence of actions meant to improve the caliber of the audio data. These procedures include noise reduction to eliminate undesired background noise, resampling to guarantee that audio data has a constant sample rate, and normalization to uniformize the amplitude of the audio. Together, these steps get the audio ready for efficient feature extraction and emotion analysis.

Feature Extraction Techniques: Mel-frequency cepstral coefficients (MFCC) are heavily highlighted in the design's description of the unique feature extraction methods. The design specifies how these coefficients will be determined from the audio data that has already been processed. A key component in identifying emotional cues in speech, MFCC is a method for capturing pertinent acoustic properties that have been around for a while.

4.2.2 Emotion Recognition Model

The architecture and operational features of the design for the emotion recognition model are highlighted in this subsection:

Neural Network Architecture: The RNN-LSTM model's architecture, including the number of layers, the units inside each layer, and the activation functions used, is specified by the design. Real-time emotion recognition has special needs, and the neural network architecture is designed to meet those criteria.

Optimization Algorithm: The optimization methods that will be utilized to train the model are defined by the design. This covers information on the learning rate, loss function, and any regularization strategies used to enhance the model's performance.

Model Parameters: The design specifies model parameters such as the initial weights and biases. For the model to correctly understand emotions and make precise predictions, these parameters are essential.

Chapter 5

Implementation

5.1 Working Methodology

There are several processes involved in implementing a Speech Emotion Recognition (SER) system employing Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM). The RNN-LSTM model for SER is implemented in Python in the section below. The prerequisites for this implementation are the required libraries, a labeled dataset, and a trained model.

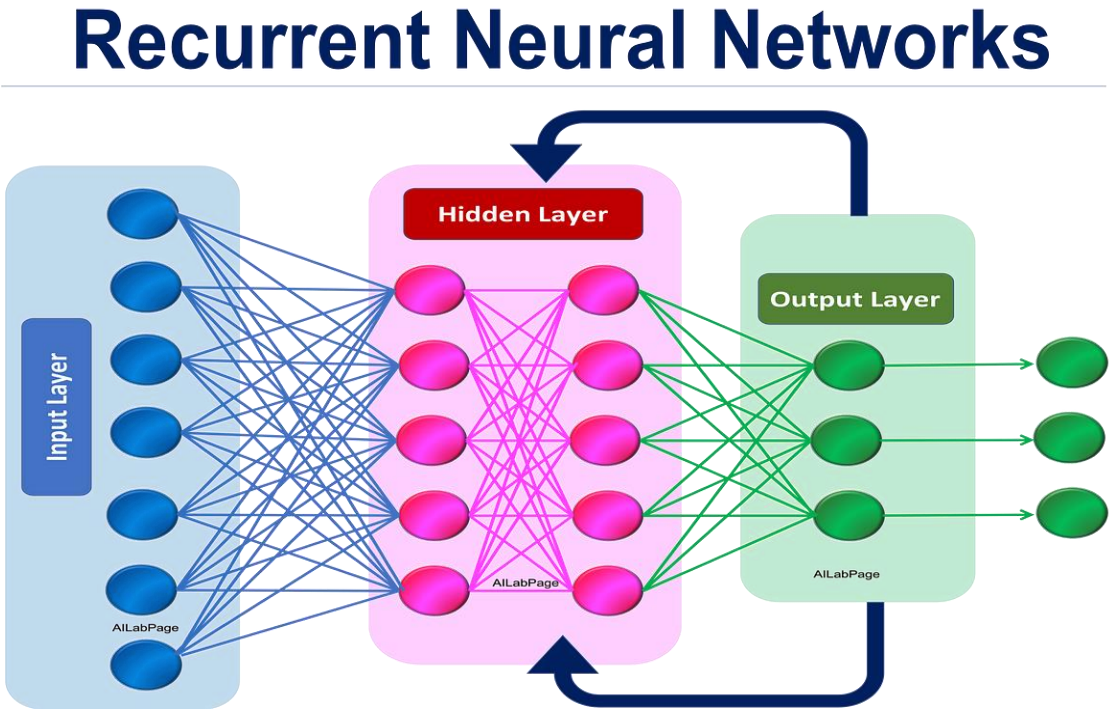

Figure 5.1: Overview of Recurrent Neural Network

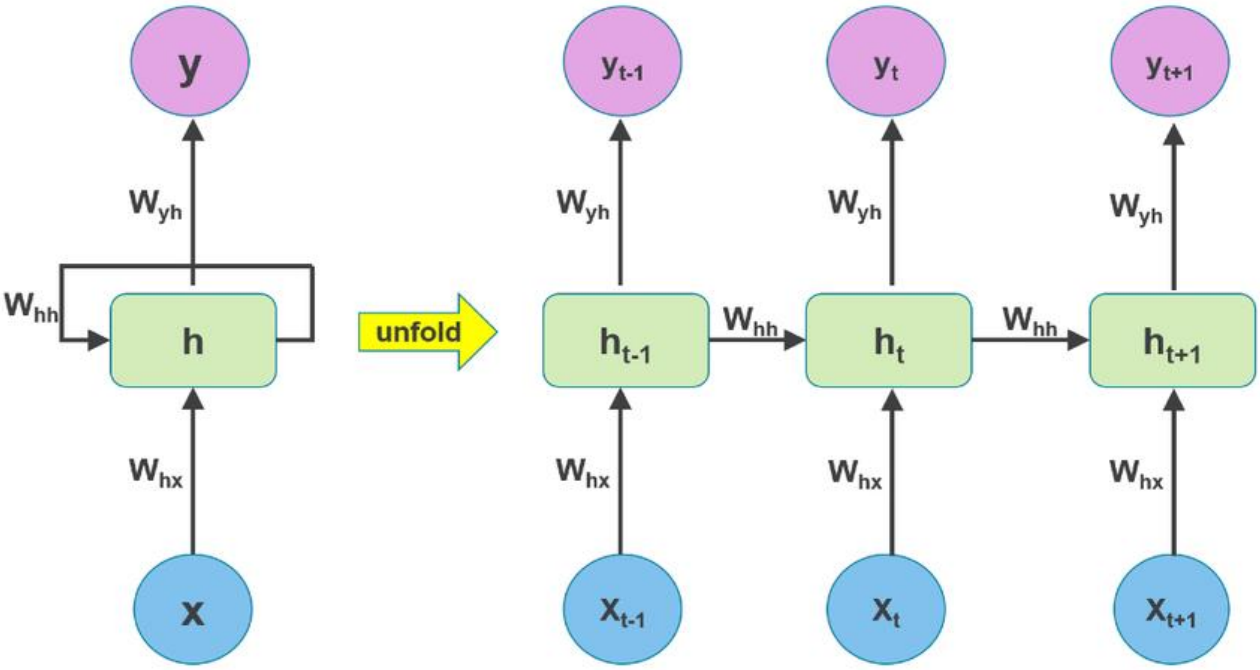

Figure 5.2: Overview of working of Neural Network

5.2 Implemented Code

5.2.1 Model Building

```
# Import Modules
import pandas as pd
import numpy as np
import os #to deal with files
import seaborn as sns # for visualization
import matplotlib.pyplot as plt # for visualization
import librosa  # for audio libraries
import librosa.display
from IPython.display import Audio #for audio playing
import warnings
warnings.filterwarnings("ignore")

# **Loading the Datasets**
paths = []
labels = []
for dirname, _, filenames in os.walk('Data'):
    for filename in filenames:
        paths.append(os.path.join(dirname,filename))
        label = filename.split('_')[-1]
        label = label.split('.')[0]
        labels.append(label.lower())
print("Data set is loaded")
```

```
# **Creating and loading the Dataframe**
df = pd.DataFrame()
df['speech'] = paths
df['label'] = labels
df.head()

print(df.info())
df['label'].value_counts()

# **Exploratory Data Analysis**
sns.countplot(x=df['label'])

def waveplot(data,sr,emotion):
    plt.figure(figsize=(5, 5))
    plt.title(emotion,size=20)
    librosa.display.waveshow(data,sr=sr)
    plt.show()


def spectogram(data,sr,emotion):
    x = librosa.stft(data)
    xdb = librosa.amplitude_to_db(abs(x))
    plt.figure(figsize=(5, 5))
    plt.title(emotion,size=20)
    librosa.display.specshow(xdb,sr=sr,x_axis='time',y_axis='hz')
    plt.colorbar()

emotion = 'fear'
path = np.array(df['speech'][df['label']==emotion])[0]
data,smapling_rate = librosa.load(path)
waveplot(data,smapling_rate,emotion)
spectogram(data,smapling_rate,emotion)
Audio(path)

# Similary Exploratory Data Analysis will be done for all the emotions

# **Feature Extraction**

def extract_mfcc(file_name):
    y,sr = librosa.load(file_name,duration=3,offset=0.5)
    mfcc =np.mean(librosa.feature.mfcc(y=y,sr=sr,n_mfcc=40).T,axis=0)
    return mfcc

extract_mfcc(df['speech'][0])

x_mfcc = df['speech'].apply(lambda x:extract_mfcc(x))
x_mfcc
X = [x for x in x_mfcc]
X = np.array(X)
X.shape
```

```
X = np.expand_dims(X,-1)
X.shape

from sklearn.preprocessing import OneHotEncoder
encoder = OneHotEncoder()
y=encoder.fit_transform(df[['label']])
y=y.toarray()
y.shape

# **Creating the LSTM model**

from keras.models import Sequential
from keras.layers import Dense, LSTM, Dropout

model = Sequential([
    LSTM(123,return_sequences=False, input_shape=(40,1)),
    Dense(64,activation='relu'),
    Dropout(0.2),
    Dense(32,activation='relu'),
    Dropout(0.2),
    Dense(7,activation='softmax'),
])

model.compile(loss='categorical_crossentropy',optimizer = 'adam',metrics=['accuracy'])
model.summary()

# Train the model
history = model.fit(X,y,validation_split=0.2,epochs=100,batch_size=512,shuffle=True)
# Saving the model
model.save('model.h5')

# **Plot the Results**
epochs = list(range(100))
acc = history.history['accuracy']
val_acc = history.history['val_accuracy']
plt.plot(epochs,acc,label="Training accuracy")
plt.plot(epochs,val_acc,label="Validation accuracy")
plt.xlabel("epochs")
plt.ylabel("Accuracy")
plt.legend()
plt.show()

loss = history.history['loss']
val_loss = history.history['val_loss']
plt.plot(epochs,loss,label="Training loss")
plt.plot(epochs,val_loss,label="Validation loss")
plt.xlabel("epochs")
plt.ylabel("loss")
plt.legend()
plt.show()
```

Chapter 6

Results and Snapshots

6.1 Graphical Analysis of Data

Figure 6.1: Bargraph indicating the number of datasets in each emotion

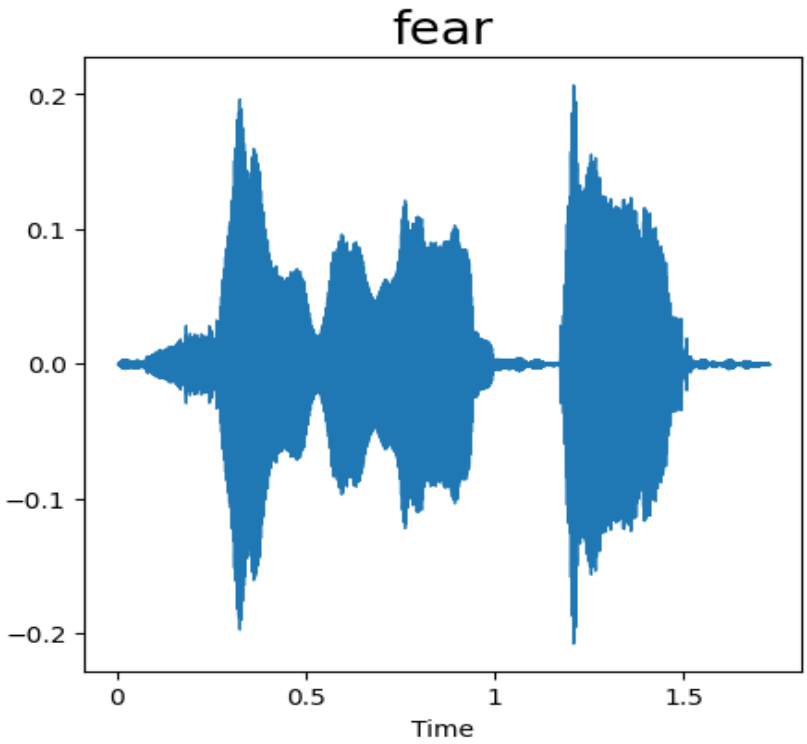

Figure 6.2: Wave show graph for the frequency of the audio

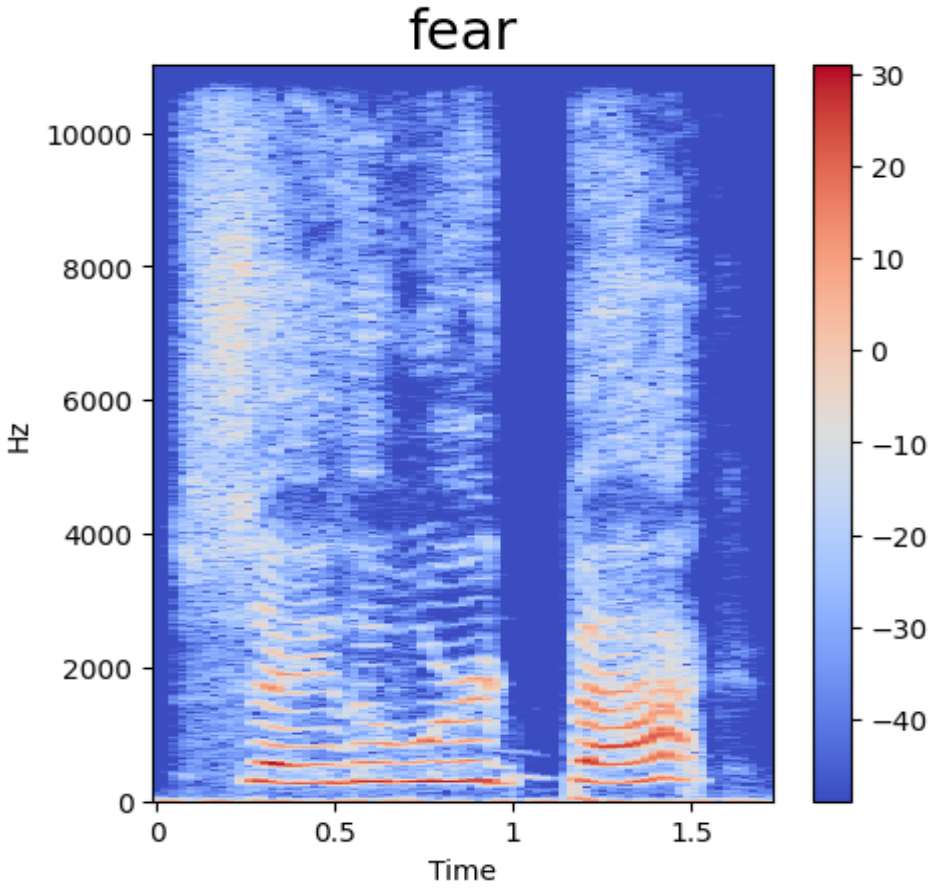

Figure 6.3: Spectrogram graph for the frequency of the audio

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm_1 (LSTM)               (None, 123)               61500

 dense_3 (Dense)             (None, 64)                7936

 dropout_2 (Dropout)         (None, 64)                0

 dense_4 (Dense)             (None, 32)                2080

 dropout_3 (Dropout)         (None, 32)                0

 dense_5 (Dense)             (None, 7)                 231

=================================================================
Total params: 71747 (280.26 KB)
Trainable params: 71747 (280.26 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Figure 6.4: Sequential trained model overview

```
Epoch 1/100
9/9 [==============================] - 5s 333ms/step - loss: 1.8369 - accuracy: 0.2583 - val_loss: 1.9436 - val_accuracy: 0.0661
Epoch 2/100
9/9 [==============================] - 2s 244ms/step - loss: 1.5779 - accuracy: 0.4069 - val_loss: 1.6972 - val_accuracy: 0.4402
Epoch 3/100
9/9 [==============================] - 3s 301ms/step - loss: 1.2654 - accuracy: 0.5536 - val_loss: 1.0860 - val_accuracy: 0.6107
Epoch 4/100
9/9 [==============================] - 2s 279ms/step - loss: 1.0107 - accuracy: 0.6058 - val_loss: 0.8476 - val_accuracy: 0.7116
Epoch 5/100
9/9 [==============================] - 2s 269ms/step - loss: 0.8070 - accuracy: 0.6975 - val_loss: 0.5776 - val_accuracy: 0.8821
Epoch 6/100
9/9 [==============================] - 2s 277ms/step - loss: 0.6298 - accuracy: 0.7940 - val_loss: 0.3730 - val_accuracy: 0.9366
Epoch 7/100
9/9 [==============================] - 2s 273ms/step - loss: 0.4551 - accuracy: 0.8589 - val_loss: 0.2202 - val_accuracy: 0.9670
Epoch 8/100
9/9 [==============================] - 2s 273ms/step - loss: 0.3399 - accuracy: 0.8958 - val_loss: 0.1383 - val_accuracy: 0.9732
Epoch 9/100
9/9 [==============================] - 2s 271ms/step - loss: 0.2788 - accuracy: 0.9145 - val_loss: 0.0956 - val_accuracy: 0.9768
Epoch 10/100
9/9 [==============================] - 2s 277ms/step - loss: 0.2159 - accuracy: 0.9371 - val_loss: 0.0832 - val_accuracy: 0.9795
Epoch 11/100
9/9 [==============================] - 2s 270ms/step - loss: 0.1875 - accuracy: 0.9473 - val_loss: 0.0736 - val_accuracy: 0.9804
Epoch 12/100
9/9 [==============================] - 2s 273ms/step - loss: 0.1684 - accuracy: 0.9527 - val_loss: 0.0427 - val_accuracy: 0.9902
Epoch 13/100
...
Epoch 99/100
9/9 [==============================] - 2s 232ms/step - loss: 0.0025 - accuracy: 0.9998 - val_loss: 0.0013 - val_accuracy: 0.9991
Epoch 100/100
9/9 [==============================] - 2s 231ms/step - loss: 0.0030 - accuracy: 0.9993 - val_loss: 0.0035 - val_accuracy: 0.9982
```

Figure 6.5: Training the model process

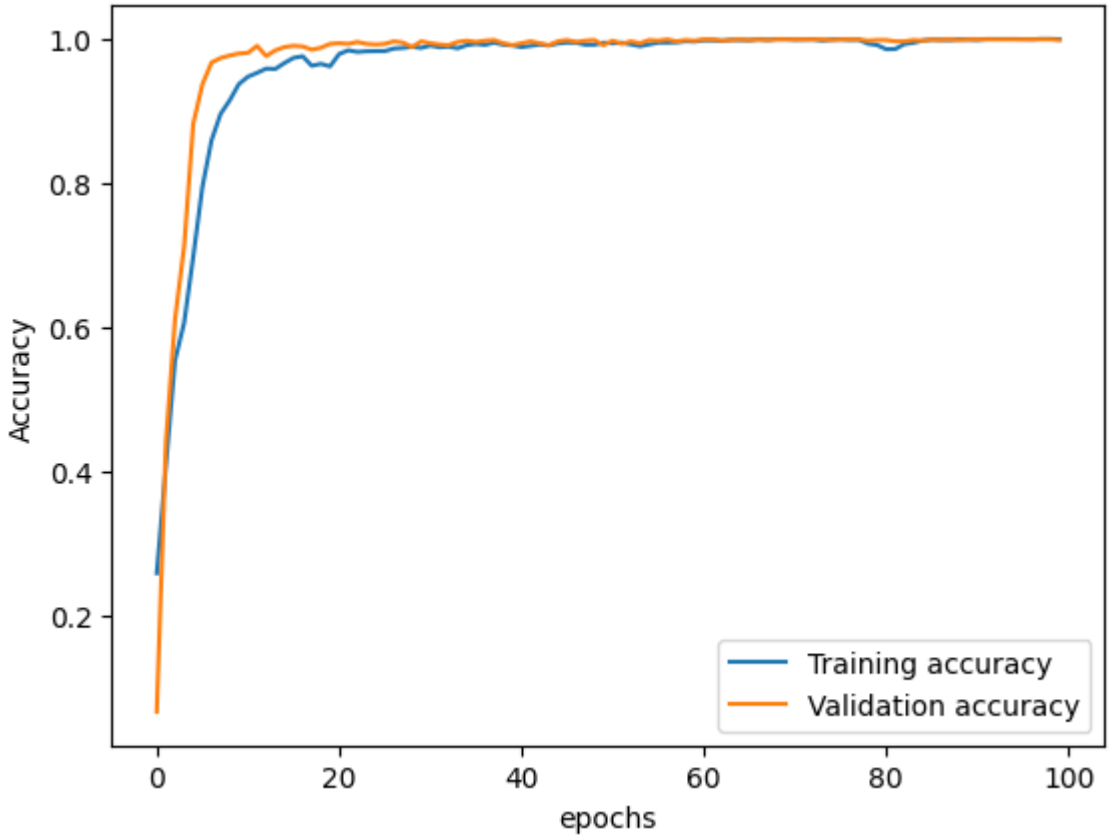

Figure 6.6: Model Training Accuracy

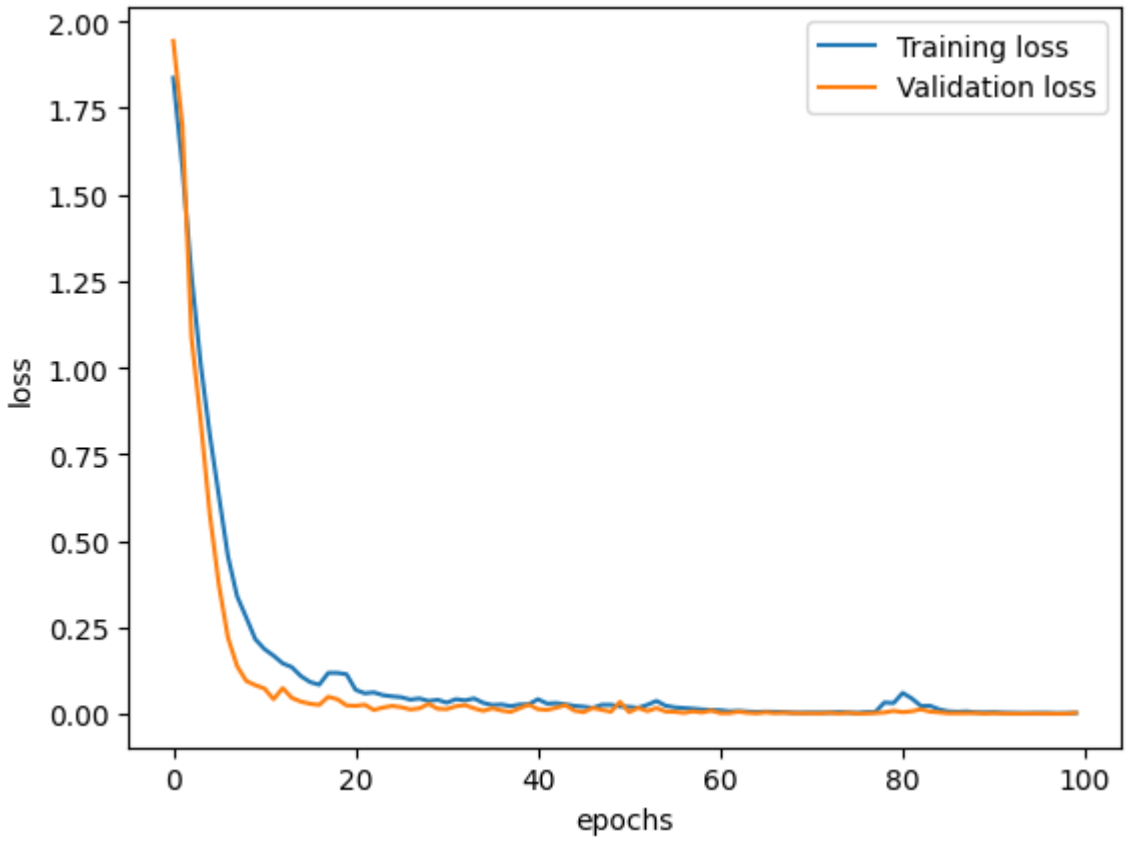

Figure 6.7: Model Testing Accuracy

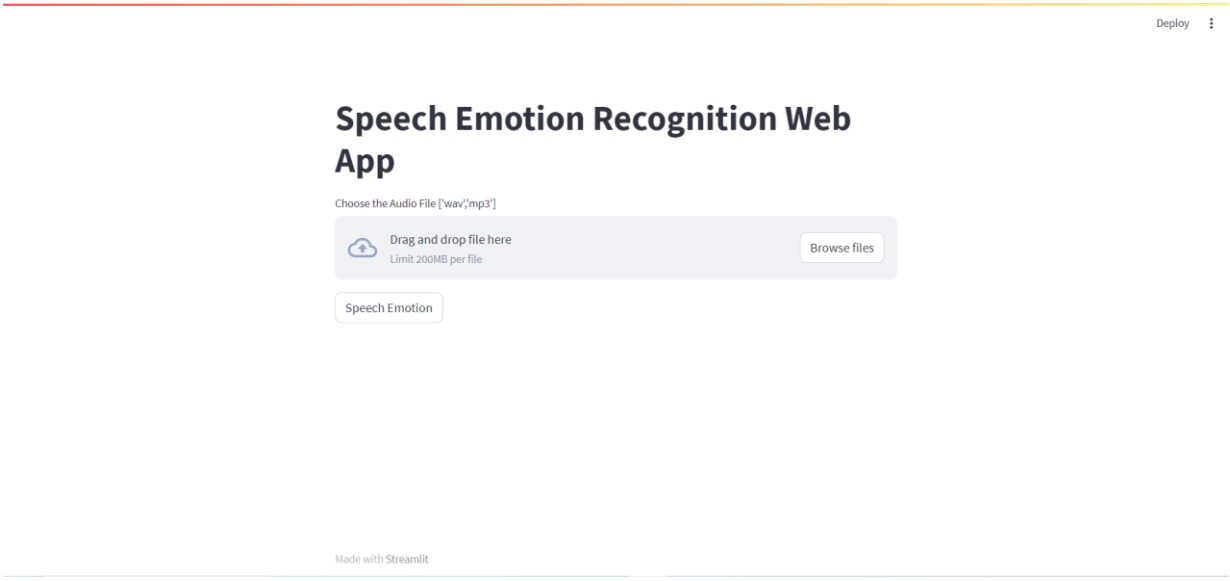

Figure 6.8: Webpage interface

Chapter 7
Conclusion and Future Enhancement

7.1 Conclusion

In this study, I have successfully constructed a voice Emotion Recognition (SER) system for evaluating and categorizing emotions in voice data using Recurrent Neural Networks (RNN) with Long Short-Term Memory (LSTM). The following are the project's main highlights and conclusions:

Accurately Recognizing Emotions: Our SER system's RNN-LSTM technology has enabled it to recognize and classify emotions within speech data with astounding precision. This precision demonstrates the system's capacity to comprehend and distinguish between the numerous emotional states expressed in speech.

Real-time emotional evaluation: The system can rapidly assess and identify the emotional content of incoming voice data because it is built to deliver real-time emotion analysis. It is very adaptable and ideal for a variety of applications thanks to its real-time capacity. Monitoring emotional states during real-time interactions, such as customer service calls or responses from a virtual assistant, are use cases for real-time emotion analysis. It can also be used in circumstances that need for quick input on emotional content.

Various Applications

The capabilities of the SER system make a wide range of application possibilities possible. These consist of:

Emotion-Aware Virtual Assistants: Adding emotional intelligence to virtual assistants to improve empathy and responsiveness in interactions.

Customer Feedback Sentiment Analysis: Determining customer sentiment by analyzing reviews of items or customer feedback left in call centers.

Creating instruments to help mental health providers track patients' emotional states using voice analysis.

7.2 Future Enhancements

Recognition of Multiple Emotions: The combination of several modalities, such as text analysis and facial expressions, with speech data represents one viable direction for future growth. Emotion recognition that is more precise and contextually aware can result from this multimodal approach. In circumstances where voice data alone may not be adequate, it can, for example, assist the system in understanding sarcasm or detecting emotions.

Various Datasets: increasing the system's capacity to distinguish a variety of emotions by exposing it to more varied and substantial voice datasets during training. This may enhance its capacity to recognize and classify emotions in various linguistic and cultural contexts.

Emotion Intensity: Future advances may include emotion intensity analysis, going beyond simple emotion categorization. Instead of simply categorizing different emotional states, this will provide people with a more complex knowledge of emotions.

Integration: The SER system can be integrated into a variety of current programs, services, and hardware. For instance, integrating it to add emotion-aware features to virtual assistants, customer support systems, or educational platforms.

Bibliography

- ML and DL course
- RNN Documentation
- LSTM Documentation
- Pandas Documentation
- Numpy Documentation
- Keras Documentation
- Librosa Documentation
- Joblib Documentation
- Seaborn Documentation

Reference Papers

- The Efficacy of Deep Learning-Based Mixed Model for Speech Emotion Recognition
- Speech Emotion Recognition: Methods and Case study
- Speech Emotion Recognition using Deep Learning Techniques: A Review
- Emotion Recognition from Speech