

SPEECH EMOTION RECOGNIZER USING LSTM MODEL

Submitted in partial fulfilment of the requirements for the award of
Bachelor of Engineering degree in Computer Science and Engineering

By

MADHUMITHA M V (Reg.No-40110689)

SIMRETHA R (Reg.No-40111231)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

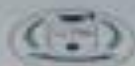
INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade “A++” by NAAC

JEPPIAR NAGAR, RAJIV GANDHI SALAI, CHENNAI – 600119

APRIL - 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Category - I University by UGC

Accredited "A++" by NAAC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **MADHUMITHA M V (Reg.No-40110689)** and **SIMRETHA R (Reg.No-40111231)** who carried out the project entitled "**SPEECH EMOTION RECOGNIZER USING LSTM MODEL**" under my supervision from November 2023 to April 2024.


Internal Guide

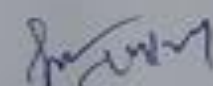
Dr.M.SREEKRISHNA M.E.,(Ph.D.)


Head of the Department

Dr. L. LAKSHMANAN

HEAD OF THE DEPARTMENT
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Jeppiaar Nagar, Rajiv Gandhi Salai,
Chennai - 600 119.

Submitted for Viva voce Examination held on 15 04.2024


Internal Examiner

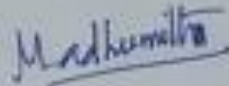

External Examiner

DECLARATION

We, MADHUMITHA M V (Reg.No-40110689), SIMRETHA R (Reg.No-40111231) hereby declare that the Project Report entitled "SPEECH EMOTION RECOGNIZER USING LSTM MODEL" done by us under the guidance of Dr.M.SREEKRISHNA M.E.,(Ph.D.), is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE: 15.04.2024

PLACE: CHENNAI



SIGNATURE OF CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to Board of Management of SATHYABAMA for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph.D.**, Dean, School of Computing, **Dr. L. Lakshmanan M.E., Ph.D.**, Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Ms. Sree Krishna M.E.,(Ph.D.)** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the Department of Computer Science and Engineering who were helpful in many ways for the completion of the project.

ABSTRACT

Emotion is a scorching topic in the recent years due to the critical unseen stress incurred during the pandemic and post-pandemic. This is worsening with the recent economy's inflation and increase of living cost, many employees are seriously affected and drawn forth many families saddened cases and tremendous drop of working performance. The increasing stress brings a lot of harm not only to the individual but to the company's and country's growth. To recognize emotion through a single model is less accurate, however, recruiting multiple-models may lead to latency in data processing and possibly misleading results if the input models data are not properly filtered and segmented. This paper will review, analyze and theoretically compare 15 facial expression methods and 17 voice methods of emotion recognition research works. It will outline the pros and cons of each method and discuss the accuracy of some of the standalone and hybrid emotion recognition methods. Some of the methods (such as CNN, KNN and SVM) can span over multiple-models, but reveal different level of strengths. This is very important to discover, so that one may replace or enhance the weaker level if applying the same method across the multiple-models. This paper will also illustrate different levels of popularity of the methods in each model for visual comparison in ease. Hopefully, it can cater the new researchers a quick identification on the most suitable method for recognizing the emotion through facial expression and/or voice.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO
	ABSTRACT	v
	LIST OF FIGURES	ix
	LIST OF ABBREVIATIONS	x
1	INTRODUCTIONS	1
	1.1 Background and Motivation	2
	1.2 Problem Statement	3
	1.3 Objectives and Scope	3
	1.4 Applications of Human Activity Recognition	4
	1.5 Organization of Thesis	5
2	LITERATURE SURVEY	6
	2.1 Overview of Existing Research and Literature	6
	2.2 Inferences from Literature Survey	11
	2.3 Open Problems in Existing System	11
3	REQUIREMENTS ANALYSIS	13
	3.1 Feasibility Studies of the Project	13
	3.1.1 Economic Feasibility	13
	3.1.2 Technical Feasibility	14
	3.1.3 Social Feasibility	14
	3.2 Requirements Specification	14
	3.2.1 Hardware Requirements	14
	3.2.2 Software Requirements	14
	3.2.3 Python	14
	3.2.4 Deep Learning	15
	3.2.5 Convolutional Neural Network	16
	3.2.6 Long Short-Term Memory	16
	3.2.7 Artificial Neural Network	17
	3.2.8 convLSTM	18
	3.2.9 LRCN	19
	3.2.10 Tensorflow	19

3.2.11	keras	20
3.3	System Use Case	20
3.3.1	System Description	20
3.3.2	System Use Case Scenario	21
4	DESCRIPTION OF PROPOSED SYSTEM	22
4.1	Description of Research Approach and Methods	22
4.1.1	Data Collection and Pre-processing	23
4.1.2	Feature Extraction	23
4.1.3	ConvLSTM and LRCN Models	24
4.1.4	Training and Testing	24
4.1.5	Evaluation Metrics	24
4.2	Architecture of Proposed System	25
4.3	Description of Software for Implementation and Testing Plan of the Proposed System	30
4.4	Project Management Plan	32
4.5	Financial Report on Estimated Costing	33
4.6	Software to Operations Plan	34
5	IMPLEMENTATION DETAILS	36
5.1	Development and Deployment Setup	36
5.2	Algorithms	37
5.2.1	Convolutional Neural Networks	37
5.2.2	Long Short Term Memory	37
5.3	Testing	38
6	OUTCOMES AND DISCUSSIONS	39
6.1	Introduction	39
6.2	IEEE Standards Followed in the Project	40
6.3	Constraints	40
6.4	Tradeoff in the Project	41
7	RESULTS AND DISCUSSIONS	43

7.1	Presentation of Findings	43
7.2	Data Analysis and Interpretation	43
8	CONCLUSION	45
8.1	Conclusion	45
8.2	Recommendations for Future Research	45
8.3	Research Issues	46
8.4	Implementation Issues	47
	REFERENCES	49
	APPENDIX	51
	A. SOURCE CODE	51
	B. SCREENSHOTS	65
	C. RESEARCH PAPER	68

LIST OF FIGURES

FIGURE NO	FIGURE NAME	PAGE NO
3.1	Structure of Convolutional Neural Network	16
3.2	Structure of Long Short-Term Memory	17
3.3	Structure of Artificial Neural Network	18
4.1	Methodology of Human Activity Recognition	22
4.2.1	Architecture Diagram	25
4.2.2	Data Flow Diagram	26
4.2.3	Class Diagram	27
4.2.4	Use Case Diagram	28
4.2.5	Activity Diagram	29
7.2.1	Calculating training and validation loss	43
7.2.2	Calculating training and validation accuracy	44
B.1	Welcome page	65
B.2	Audio and live cam module	65
B.3	Emotion prediction from audio	66
B.4	psychiatrist consultation	66
B.5	permission for camera access	67
B.6	detection of emotions	67

LIST OF ABBREVIATIONS

ABBREVIATION	EXPANSION
HAR	Human Activity Recognition
CNN	Convolutional Neural Network
LSTM	Long Short Term Memory
ConvLSTM	Convolutional Long Short Term Memory
LRCN	Long-term Recurrent Convolutional Network

CHAPTER 1

INTRODUCTION

Speech emotion recognition (SER) is an active area of research and a better way to communicate using among human–computer interaction (HCI). Speech signals play an important role in various real-time HCI applications, such as clinical studies, audio surveillance, lies detection, games, call centre, entertainment, and many more. However, the existing SER techniques still have some limitations, which include robust feature selection and advanced machine learning methods, for an efficient system. Thus, researchers are still working to find a significant solution in order to choose the right features and advance the artificial intelligence (AI) based classification techniques. Similarly, the background noise in a real-world voice could also be dramatically effective on the machine learning system. Nevertheless, the development of a decent speech-based emotion recognition system can easily increase the user experience in different areas with the HCI, such as AI cyber security and mobile health (mHealth).

The AI model has better potential than classical model to recognize the emotional state of the speaker from their signals during speech and shows a considerable impact on the SER in order to imitate these emotions. Nowadays, the researchers utilize the deep learning approaches in order to solve the recognition problems, such as voice recognition, emotion recognition, gesture recognition, face recognition, and image recognition. The main advantage of the utilization of deep learning approaches is the automatic selection of the features. The suggested model adjusts the weights in the convolution operation according to the input data, and it finds the important and the task-specific attributes.

Recently, the researchers have introduced a various number of deep neural networks (DNNs) techniques to model the emotions recognition in the speech data. These models are fundamentally different in nature.

For example, one group DNN model that detects significant cues from raw audio samples, and another group utilized a particular representation of an audio recording to provide input.

for the model. In order to make a robust and a significant model, the researchers utilized different types of feature combinations using diverse network strategies. Nowadays, deep learning and artificial intelligence methods are dominant toward extracting hidden cues and to recognize lines, curves, dots, shapes, and colors, because they utilize various types of convolution operations. Thus, the researchers have utilized the modest end-to-end models for the emotion recognition, which include convolution neural networks (CNNs), recurrent neural networks (RNNs), long short-term memory (LSTM), and deep belief networks (DBNs). These models extract high-level salient features from the speech signals that achieved a better recognition rate compared to the low-level features.

The researchers have used a deep learning model for an efficient SER, but the level of accuracy and the recognition rate are still quite low due to the data scariness and the worst model configuration. The current CNN approaches lack the ability to increase the accuracy of the emotion recognition in the SER domain. the Furthermore, researchers have utilized the RNN and the LSTM in order to learn the long-term dependencies and recognize the emotions. Nevertheless, these techniques have not revealed any significant changes in accuracy, but they increased the cost computation and training time of the whole model.

1.1 BACKGROUND AND MOTIVATION

Human activity recognition using LSTM and CNN is a popular research area in the field of deep learning. LSTM and CNN are two deep learning architectures that can be used together to recognize human activities from sensor data. LSTM is used to recognize time-sequential features, while CNN is used to extract features from signals. This technology has many potential applications, such as in healthcare, sports, and security. The use of deep learning models for human activity recognition has shown great progress in recent years. The combination of LSTM and CNN has been found to be effective in recognizing human activities with high accuracy. This technology has the potential to improve the quality of life for people by providing personalized healthcare and fitness monitoring.

1.2 PROBLEM STATEMENT

The problem statement for human activity recognition using LSTM and CNN is to accurately recognize and classify human activities from sensor data. The traditional pattern recognition methods have limitations in recognizing complex human activities. Therefore, deep learning models such as LSTM and CNN are used to overcome these limitations. The standards of designing the architecture, deciding the appropriate language for the effective architecture which could extract features from sensor data and recognize time-sequential features with high accuracy is followed as per the standards IEEE Std. 1016-1998 [22] Recommended Practice for Software Design Descriptions. The use of smartphones sensors data for human activity recognition is a popular research area. The goal is to develop a system that can recognize different forms of human activities in real-time with high accuracy. The system should be able to recognize activities such as walking, running, sitting, standing, and other complex activities.

1.3 OBJECTIVE AND SCOPE

Objective

The objective of human activity recognition using LSTM and CNN is to develop an accurate and efficient system that can recognize and classify human activities from sensor data. The system should be able to recognize different forms of human activities in real-time with high accuracy. The use of deep learning models such as LSTM and CNN can help to overcome the limitations of traditional pattern recognition methods. The system should be able to recognize activities such as walking, running, sitting, standing, and other complex activities. The goal is to develop a system that can be used in various applications such as healthcare, sports, and security. The system should be able to provide personalized healthcare and fitness monitoring. The use of smartphones sensors data for human activity recognition is a popular research area.

Scope

The scope of human activity recognition using LSTM and CNN is broad and has many potential applications in various fields. In healthcare, it can be used to monitor patients' activities and provide personalized healthcare. It can also be used in sports

to monitor athletes' activities and improve their performance. The system can be used in security to detect suspicious activities and prevent crimes. The use of smartphones sensors data for human activity recognition has made it possible to develop low-cost and portable systems that can be used in various applications. The combination of LSTM and CNN has shown promising results in recognizing human activities with high accuracy. The system can recognize activities such as walking, running, sitting, standing, and other complex activities. The use of deep learning models for human activity recognition has shown great progress in recent years.

1.4 APPLICATIONS OF HUMAN ACTIVITY RECOGNITION

In healthcare, HAR can be used to monitor the daily activities of elderly people or patients with chronic conditions. The system can alert caregivers if there is a deviation from normal activity patterns, indicating potential health issues or falls. HAR can also be used to monitor the physical activity levels of patients with chronic conditions, such as diabetes or cardiovascular disease, to assess their adherence to prescribed exercise regimens.

In security, HAR can be used for surveillance in public areas, such as airports or shopping malls, to detect suspicious behaviour or identify individuals engaged in criminal activities. HAR can also be used for crowd monitoring and control during large events, such as concerts or sporting events.

In sports analysis, HAR can be used to analyse the performance of athletes during training or competitions. The system can provide feedback on their technique and suggest improvements based on their activity patterns. HAR can also be used for injury prevention by monitoring the movement patterns of athletes and identifying potential sources of strain or injury.

In robotics, HAR can be used to enable robots to interact with humans more effectively by understanding their activities and intentions. HAR can also be used to enable robots to perform household tasks, such as cleaning or cooking, by recognizing and responding to human activities.

1.5 ORGANIZATION OF THESIS

The organization includes an introduction that provides an overview of the research problem, objectives, and significance. The literature review section discusses the existing research on human activity recognition using LSTM and CNN and the Limitations of the existing system. The methodology section describes the data collection process, pre-processing, and the deep learning models used. The results section presents the findings of the study, including the accuracy of the models and the comparison between Conv LSTM and LRCN. The discussion interprets the results, discusses the limitations of the study, and suggests research directions. Finally, the conclusion section summarizes the main findings and contributions of the study. The thesis also includes an abstract, acknowledgments, and references.

CHAPTER 2

LITERATURE SURVEY

Several studies on human activity recognition have been conducted in the recent years. This section summarizes all of the previous work on facial expression recognition.

2.1 OVERVIEW OF EXISTING RESEARCH AND LITERATURE

RUHUL AMIN KHALIL & EDWARD JONES's [1] paper emotion recognition from speech signals is an important but challenging component of Human-Computer Interaction (HCI). In the literature of 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.

ABDELAZIZ A. ABDELHAMID & EL-SAYED M. EL-KENAWY [2] one of the main challenges facing the current approaches of speech emotion recognition is the lack of a dataset large enough to train the currently available deep learning models properly. Therefore, this paper proposes a new data augmentation algorithm to enrich the speech emotions dataset with more department, College of Computing and through a careful addition of noise fractions. In addition, the hyperparameters of the currently available deep learning models are either handcrafted or adjusted during the training process. Therefore, we propose an optimized deep learning model in which the hyperparameters are optimized to find their best settings and thus achieve more recognition results.

AZAMAT MUKHAMEDIYA, S. L & SIAMAC FAZLI [3] On the Effect of Log-Mel Spectrogram Parameter Tuning for Deep Learning-Based Speech Emotion Recognition has become a major area of investigation in human computer

interaction. Conventionally, SER is formulated as a classification problem that follows a common methodology: (i) extracting features from speech signals; and (ii) constructing an emotion classifier using extracted features. With the advent of deep learning, however, the former stage is integrated into the latter. That is to say, deep neural networks (DNNs), which are trained using log-Mel spectrograms (LMS) of audio waveforms, extract discriminative features from LMS. A critical issue, and one that is often overlooked, is that this procedure is done without relating the choice of LMS parameters to the performance of the trained DNN classifiers.

PATRÍCIA J. BOTA & CHEN WANG [4] A Review, Current Challenges, and Future Possibilities on Emotion Recognition Using Machine Learning and Physiological Signals. The seminal work on Affective Computing in 1995 by Picard set the base for computing that relates to, arises from, or influences emotions. Affective computing is a multidisciplinary field of research spanning the areas of computer science, psychology, and cognitive science. Potential applications include automated driver assistance, healthcare, human-computer interaction, entertainment, marketing, teaching and many others. Thus, quickly, the field acquired high interest, with an enormous growth of the number of papers published on the topic since its inception.

TAIBA MAJID WANI & TEDDY SURYA GUNAWAN [5] A Comprehensive Review of Speech Emotion Recognition Systems. During the last decade, Speech Emotion Recognition (SER) has emerged as an integral component within Human-computer Interaction (HCI) and other high-end speech processing systems. Generally, an SER system targets the speaker's existence of varied emotions by extracting and classifying the prominent features from a pre processed speech signal. However, the way humans and machines recognize and correlate emotional aspects of speech signals are quite contrasting quantitatively and qualitatively, which present enormous difficulties in blending knowledge from interdisciplinary fields, particularly speech emotion recognition, applied psychology, and human-computer interface.

F. DIPL & T. VOGT [6] Real-time automatic emotion recognition from speech. It focuses on developing a system that can automatically recognize emotions from

speech signals in real-time. The authors propose a methodology that involves extracting relevant features from speech signals, such as pitch, intensity, spectral features, and prosodic features, which carry emotional information. Machine learning algorithms, such as Support Vector Machines (SVMs), Hidden Markov Models (HMMs), or Neural Networks, are then used to classify these features into different emotional categories.

J. SCHMIDHUBER [7] Deep Learning in neural networks: An overview Neural Networks. The paper starts by tracing the historical development of neural networks, highlighting key milestones such as the perceptron, backpropagation algorithm, and the resurgence of interest in neural networks in the 21st century. It discusses the concept of deep learning, which involves training neural networks with many layers (deep architectures) to learn hierarchical representations of data. This approach has been successful in various domains such as image recognition, natural language processing, and speech recognition.

ADAMS, T. A., & WHITE B. D [8] Real-Time Face and Speech Emotion Recognition in Occupational Stress Analysis. It may delve into techniques for analysing facial expressions in real-time, including computer vision algorithms, facial feature extraction, and emotion classification models. The chapter likely covers methods for analysing speech signals to extract emotional cues, such as prosodic features, speech intonation, and sentiment analysis techniques. It may discuss approaches for integrating data from facial expression analysis and speech emotion analysis to improve the accuracy and robustness of emotion recognition systems.

SMITH A. B., JOHNSON, C. D., & LEE, E. F [9] Real-time Emotion Recognition for Worker Stress Analysis: Integrating LSTM and CNN Models. The paper likely begins by discussing the significance of real-time emotion recognition in analysing worker stress and its impact on workplace productivity and well-being. It may provide a brief overview of LSTM and CNN models, highlighting their strengths in processing sequential data (such as speech signals) and spatial data (such as facial images), respectively. The paper likely describes how LSTM and CNN models are integrated to leverage their complementary strengths for more accurate and robust emotion recognition in real-time scenarios. It may discuss the preprocessing steps involved

in preparing input data (such as speech signals and facial images) for the LSTM and CNN models, as well as the feature extraction techniques used.

A. MEHRABIAN & S. R. FERRIS [10] 'Inference of attitudes from nonverbal communication in two channels. The paper likely introduces the concept of nonverbal communication and its role in conveying attitudes and emotions. It may discuss the two channels of nonverbal communication studied in the paper, which could include facial expressions, body language, gestures, vocal tone, and other nonverbal cues. The paper likely explores how observers infer attitudes from nonverbal cues, focusing on the accuracy and reliability of such inferences. It may describe the experimental setup used to study nonverbal communication and attitude inference, including the types of stimuli presented to participants and the methods of data collection and analysis.

S. FURUI, T. KIKUCHI, Y. SHINNAKA, & C. HORI [11], "Speech-to-Text and Speech-to-Speech Summarization. The paper may begin by discussing the importance of summarizing speech content, especially in scenarios where large volumes of spoken data need to be processed efficiently. It may delve into techniques for converting spoken speech into text and then summarizing the text representation using methods such as text summarization algorithms, natural language processing (NLP) techniques, and machine learning approaches. The paper may also explore methods for directly summarizing spoken speech without intermediate text representation. This could involve extracting key phrases or sentences, identifying important topics, and generating concise summaries.

M. EI AYADI, M. S. KAMEL & F. KARRAY [12] "Survey on speech emotion recognition. The paper likely begins by introducing the concept of SER and its importance in understanding and analysing human emotions conveyed through speech. It may discuss different types of features used in SER, such as acoustic features (e.g., pitch, energy, formants), prosodic features (e.g., intonation, rhythm), spectral features (e.g., Mel-frequency cepstral coefficients), and linguistic features (e.g., sentiment analysis, lexical features). It may cover existing databases and datasets used for training and evaluating SER systems, such as the Berlin

Emotional Speech Database (EMO-DB), the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS), and others.

Y. LECUN, Y. BENGIO, and G. HINTON [13] Deep learning. The paper likely starts by introducing the concept of deep learning, explaining its significance in the field of artificial intelligence and machine learning. It may discuss the historical development of deep learning, highlighting key milestones, breakthroughs, and advancements that have led to its current state. The paper likely covers various deep learning architectures, such as deep neural networks (DNNs), convolutional neural networks (CNNs), recurrent neural networks (RNNs), and their variants like Long Short-Term Memory (LSTM) networks and deep belief networks (DBNs).

J. AZCARRAGA & M. T. SUAREZ [14] Recognizing student emotions using brainwaves and mouse behaviour data. The paper likely introduces the importance of recognizing student emotions in educational settings and the potential benefits of using brainwave and mouse behaviour data for this purpose. It may discuss methods and technologies for collecting and analysing brainwave data, such as electroencephalography (EEG) techniques and signal processing algorithms. It may describe machine learning or pattern recognition models used to classify student emotions based on the analysed brainwave and mouse behaviour data, potentially including techniques like support vector machines (SVM), neural networks, or decision trees.

J. A. R. ELIOT & A. HIRUMI [15] Emotion theory in education research practice. The paper likely begins with an introduction to the role of emotions in education, emphasizing the importance of understanding and integrating emotion theory into educational research and practice. It may explore emotion theory from interdisciplinary perspectives, drawing insights from fields such as psychology, neuroscience, educational psychology, sociology, and cognitive science. It may discuss the impact of emotions on learning processes, including emotion regulation strategies, emotional arousal, motivation, attention, memory, and decision-making in educational settings.

2.2 INFERENCES FROM LITERATURE SURVEY

After going through the previous works we inferred some points. In Human activity recognition system based on the Android platform, they used the clustered approach to enhance the performance accuracy, and execution time but the classification times are dependent on the device models and capabilities. Real-time HAR model based on the smartphone inertial sensors used six different machine learning algorithms but they used inertial sensors. Novel elderly person tracking system using a machine learning algorithm. In this work, they used the CNN model with a k value of 5, which was able to achieve the best accuracy of 98.70% in detecting the real-time activity of elderly people. Davide Anguita et al.'s paper, they introduced the improvised Support Vector Machine algorithm, which works with fixed point arithmetic to produce an energy-efficient model for the classification of human activities using a smartphone. Proposed model successfully achieved a recall of 98%, which was further compared to the existing deep learning models such as the RNN, Convolutional Neural Network (CNN), and Human computer interaction(HCI). A deep 10 LSTM network for recognizing six different activities based on Human emotion data. The network was able to achieve an accuracy of 98.70% on the UCI-HAD dataset.

2.3 OPEN PROBLEMS IN EXISTING SYSTEM

While sensor-based human activity recognition (HAR) has shown great promise in recent years, there are still several open problems in the existing system that need to be addressed. Some of these open problems include:

- **Temporal Dynamics:** Emotions expressed through speech and facial expressions are dynamic and evolve over time. Traditional machine learning approaches may struggle to capture the temporal dependencies present in sequential data, resulting in suboptimal performance in emotion recognition tasks.
- **Variability and Context Dependence:** Emotions can manifest differently

across individuals and cultural contexts, adding to the complexity of emotion recognition. Furthermore, the same emotion can be expressed in various ways, making it challenging to develop robust recognition algorithms that generalize well across diverse populations and situations.

- **Data Sparsity and Imbalance:** Annotated datasets for speech and facial emotion recognition often suffer from sparsity and class imbalance, limiting the effectiveness of traditional machine learning models. Additionally, collecting labeled data for less prevalent emotions or specific demographic groups can be challenging, hindering the development of accurate and inclusive recognition systems.
- **Multi-Modal Integration:** Integrating information from multiple modalities, such as speech and facial expressions, presents both opportunities and challenges. While combining modalities can potentially improve recognition performance and robustness, effectively leveraging complementary information requires sophisticated modeling techniques capable of capturing cross-modal dependencies and interactions.

CHAPTER 3

REQUIREMENTS ANALYSIS

3.1 FEASIBILITY STUDIES OF THE PROJECT

All systems are feasible when provided with unlimited resource and infinite time. But unfortunately, this condition does not prevail in practical world. So it is both necessary and prudent to evaluate the feasibility of the system at the earliest possible time. Also IEEE 12207.2-1997 [25] Industry Implementation of International Standard ISO/IEC 12207:1995 (ISO/IEC 12207) Standard for Information Technology – Software Life Cycle Processes – Implementation Considerations helps to provide an effective framework and method to develop software applications. It helps to produce software with the highest quality and lowest cost in the shortest time. Months or years of effort, thousands of rupees and untold professional embarrassment can be averted if an ill- conceived system is recognized early in the definition phase. Feasibility & risk analysis are related in many ways. If project risk is great, the feasibility of producing quality software is reduced. In this case three key considerations involved in the feasibility analysis are:

- Economical Feasibility
- Technical Feasibility
- Social Feasibility

3.1.1 Economic Feasibility

- **Computing resources** - Training CNN-LSTM models for activity recognition requires high computing resources, such as GPUs and high memory capacity, which may increase the cost of implementation.
- **Maintenance and Upkeep** - Regular maintenance of sensors and computing resources may add to the overall cost of implementation.
- **Cost of Development** - Developing and implementing the CNN-LSTM model requires skilled professionals, and the cost may vary based on the size and complexity of the project.

3.1.2 Technical Feasibility

- **Availability of data** - The availability of labeled data covering a diverse range of human activities and environmental conditions is essential for the success of the CNN-LSTM model.
- **Model architecture** - The design of the CNN-LSTM model plays a critical role in the performance of activity recognition. Hyperparameter tuning and feature selection are also important factors to consider.

3.1.3 Social Feasibility

- **Privacy Concerns** - Collecting data on individuals activities may raise privacy concerns. Proper consent and transparency must be ensured during the data collection and usage process.
- **Social Acceptance** - Acceptance of the technology and willingness to use it in public settings must be considered.
- **Social Benefits** - Human activity recognition can have significant benefits, such as improving healthcare, sports training, and rehabilitation programs, which can improve people's quality of life.

3.2 REQUIREMENTS SPECIFICATION

Based on the reference of IEEE Std. 830 – 1998 [19] Recommended Practice for Software Requirements, The Hardware and Software Requirements are described

3.2.1 Hardware Requirements

- RAM : 4GB or above 4GB
- Processor of Frequency : 1.5GHz or above
- Processor : Intel Pentium 4 or higher

3.2.2 Software Requirements

- Operating System : Windows 8 and above
- Languages : Python
- Tools used : Visual Studio Code, Jupyter Notebook

3.2.3 Python

Python is a scripting language that is high-level, interpreted, interactive, and object-oriented. Python is intended to be extremely readable. It commonly employs English

terms rather than punctuation, and it has fewer syntactical structures than other languages.

Python Features

- **Easy-to-learn** - Python has a small number of keywords, a basic structure, and a well-defined syntax. This enables the pupil to swiftly learn the language.
- **Easy-to-read** - Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** - Python's source code is relatively simple to maintain.
- **A broad standard library** - The majority of Python's library is portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** - Python features an interactive mode that enables for interactive testing and debugging of code snippets.
- **Portable** - Python can operate on a broad range of hardware devices and has the same interface across them all.
- **Extendable** - The Python interpreter may be extended using low-level modules. These modules allow programmers to enhance or adapt their tools to make them more efficient. Databases - Python provides interfaces to all major commercial databases.
- **GUI Programming** - Python can construct and port GUI programmes to numerous system calls, libraries, and windows systems, including Windows MFC, Macintosh, and Unix's X Window system.
- **Scalable** - Python provides a better structure and support for large programs than shell scripting.

3.2.4 Deep Learning

Deep learning is a type of machine learning that uses algorithms meant to function in a manner similar to the human brain. While the original goal for AI was broadly to make machines able to do things that would otherwise require human intelligence, the idea has been refined in the decades since. François Chollet, AI researcher at Google and creator of the machine learning software library Keras, says: "Intelligence is not a skill in itself, it's not about what you can do, but how well and how efficiently you can learn new things."

3.2.5 Convolutional Neural Network

Convolutional Neural Networks (CNNs) have revolutionized various fields of artificial intelligence, including computer vision, natural language processing, and speech recognition. CNNs are particularly well-suited for tasks involving spatial or sequential data, thanks to their ability to automatically learn hierarchical representations from raw input data. In speech emotion recognition, CNNs have been successfully employed to analyse spectrograms or other representations of audio signals. By applying convolutional filters to input spectrogram images, CNNs can extract meaningful features that capture relevant patterns in the frequency domain. These features can then be fed into fully connected layers for emotion classification. The architecture of a CNN for speech emotion recognition typically consists of multiple convolutional layers followed by pooling layers to reduce dimensionality and extract higher-level features. Additionally, techniques such as dropout regularization and batch normalization may be applied to prevent overfitting and improve generalization performance. The final layers of the network often consist of fully connected layers and soft max activation functions for emotion classification.

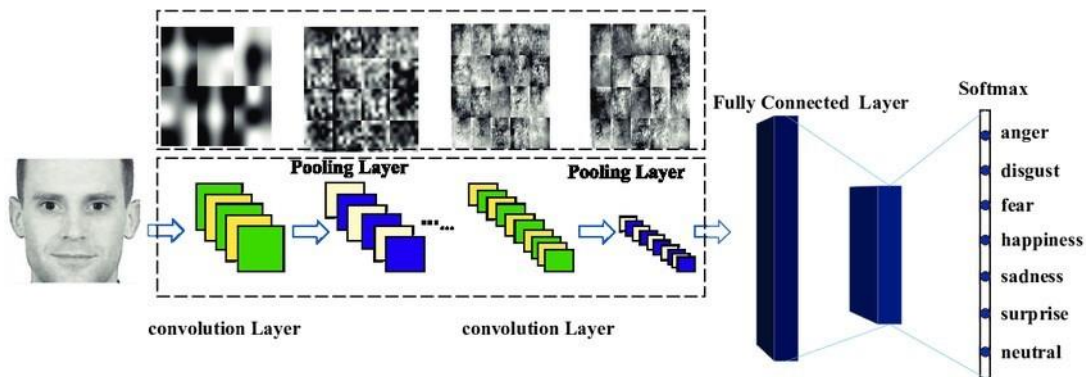


Fig. 3.1: Structure of Convolutional Neural Network

3.2.6 Long Short-Term Memory

Long Short-Term Memory (LSTM) networks are a type of recurrent neural network capable of learning order dependence in sequence prediction problems. This is a

behavior required in complex problem domains like machine translation, speech recognition, and more. LSTMs are a complex area of deep learning. It can be hard to get your hands around what LSTMs are, and how terms like bidirectional and sequence-to-sequence relate to the field. In this post, you will get insight into LSTMs using the words of research scientists that developed the methods and applied them to new and important problems. There are few that are better at clearly and precisely articulating both the promise of LSTMs and how they work than the experts that developed them.

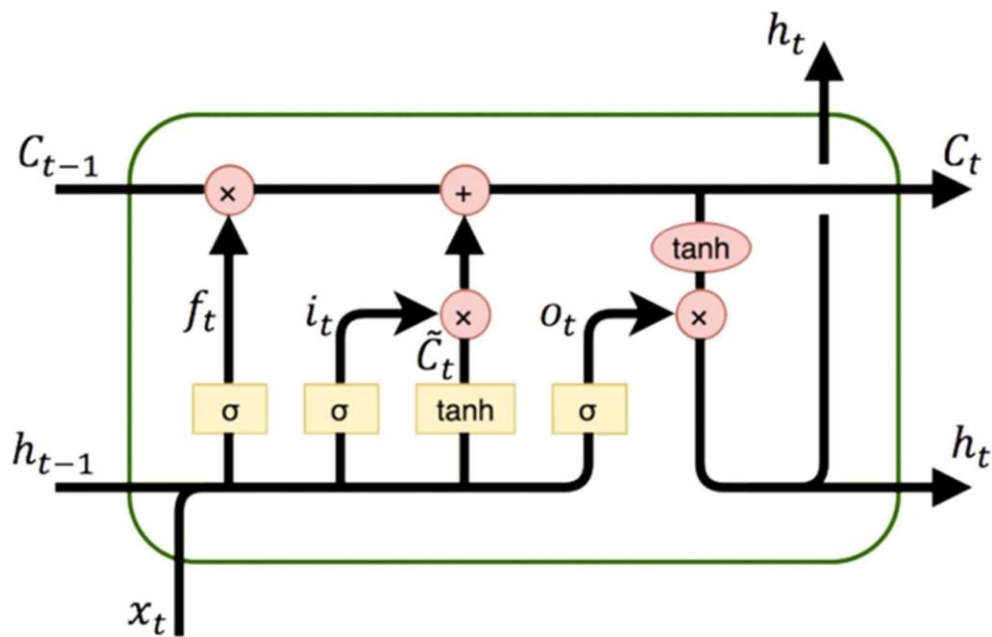


Fig. 3.2: Structure of Long Short-Term Memory

3.2.7 Artificial Neural Network

Artificial Neural Network Tutorial provides basic and advanced concepts of ANNs. Our Artificial Neural Network tutorial is developed for beginners as well as professions. The term "Artificial neural network" refers to a biologically inspired subfield of artificial intelligence modeled after the brain. An Artificial neural network is usually a computational network based on biological neural networks that construct the structure of the human brain. Similar to a human brain has neurons interconnected to each other, artificial neural networks also have neurons that are linked to each other in various layers of the networks. These neurons are known as nodes.

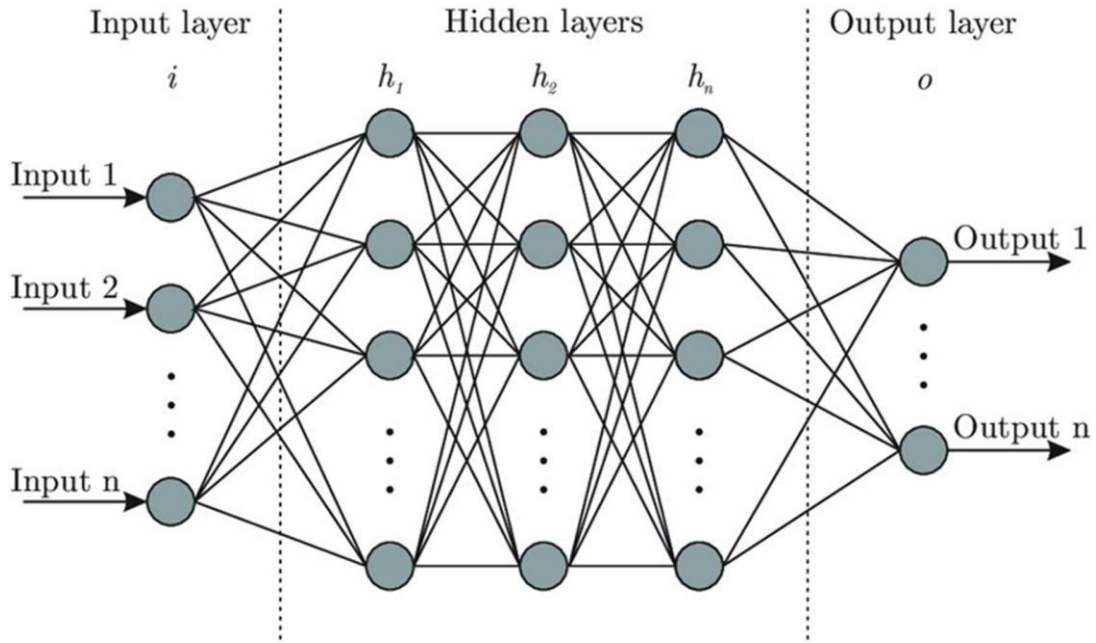


Fig. 3.3: Structure of Artificial Neural Network

3.2.8 ConvLSTM

ConvLSTM is a type of recurrent neural network (RNN) that combines the convolutional neural network (CNN) and LSTM (long short-term memory) architectures. It is a powerful model that can be used for various tasks, including video prediction, image processing, and natural language processing. The ConvLSTM network is designed to learn spatial and temporal dependencies in sequential data. It consists of convolutional layers, which extract features from the input data, and LSTM layers, which capture the temporal dependencies in the data. The combination of these layers enables the ConvLSTM network to learn both spatial and temporal patterns in the input data.

The ConvLSTM architecture is similar to the standard LSTM architecture, with the addition of convolutional layers before the LSTM layers. The convolutional layers help in reducing the spatial dimensions of the input data and extracting relevant features. The output of the convolutional layers is then passed to the LSTM layers, which capture the temporal dependencies in the data. The ConvLSTM architecture has shown significant improvements in various tasks such as video prediction, where it can learn long-term dependencies in sequential data. It is also useful in

image processing tasks, where it can capture the spatial dependencies in the data.

In human activity recognition, ConvLSTM models have shown promising results in accurately classifying physical activities by capturing both spatial and temporal features in the sensor data.

3.2.9 LRCN

LRCN (Long-term Recurrent Convolutional Networks) is a type of neural network architecture that combines the strengths of CNN (Convolutional Neural Networks) and LSTM (Long Short-Term Memory) models. The CNN part of the LRCN architecture is used to extract spatial features from the input data, such as images or video frames, while the LSTM part is used to capture the temporal dependencies between these features. The CNN layers are typically used as a feature extractor, while the LSTM layers are responsible for learning the temporal patterns in the extracted features.

In human activity recognition, LRCN models have shown promising results in accurately classifying physical activities by capturing both spatial and temporal features in the sensor data. The CNN layers can learn useful visual features from raw sensor data, while the LSTM layers can capture the temporal dependencies between these features, allowing the model to accurately recognize complex activity sequences.

Overall, LRCN models have been shown to be effective in various tasks such as video analysis, image captioning, and speech recognition, where they can learn both spatial and temporal patterns in the data.

3.2.10 Tensorflow

TensorFlow is an open-source library for numerical computation and machine learning developed by Google. It is designed to simplify the process of building, training, and deploying machine learning models by providing a high-level API for building neural networks, as well as low-level APIs for more advanced users. TensorFlow supports a wide range of platforms, from desktops to clusters of GPUs and TPUs, and can be used for a variety of tasks, including image and speech

recognition, natural language processing, and recommendation systems. It is widely used in industry and academia for research and production applications.

3.2.11 Keras

TensorFlow is an open-source library for numerical computation and machine learning developed by Google. It is designed to simplify the process of building, training, and deploying machine learning models by providing a high-level API for building neural networks, as well as low-level APIs for more advanced users. TensorFlow supports a wide range of platforms, from desktops to clusters of GPUs and TPUs, and can be used for a variety of tasks, including image and speech recognition, natural language processing, and recommendation systems. It is widely used in industry and academia for research and production applications.

Advantages of Keras

Keras has several advantages that make it a popular choice for deep learning.

- **Simplicity** - Keras is very easy and simple to use. It is a user-friendly API with easy-to-learn and code features.
- **Backend support** - Keras does not operate with low-level computations. So, it supports the use of backends.
- **Pre-trained models** - Keras provides numerous pre-trained models.
- **Fast experimentation** - Keras is built to simplify the tasks of users.

3.3 SYSTEM USE CASE

By following the IEEE Std. 1016-1998 [22] Recommended Practice for Software Design Descriptions.

3.3.1 System Description

The system is designed to recognize human activities using videos as input. The system utilizes two deep learning models, Convolutional LSTM (ConvLSTM) and Long-term Recurrent Convolutional Networks (LRCN), to extract and classify features from videos, respectively.

3.3.2 System Use Case Scenario

- User uploads a video containing a human activity to the system.
- The system preprocesses the video by dividing it into frames and resizing each frame to a standard size.
- The ConvLSTM model is used to extract features from the video frames. The ConvLSTM model takes in a sequence of video frames as input and outputs a sequence of feature maps that capture the temporal dependencies between frames.
- The LRCN model is used to classify the human activity based on the extracted features. The LRCN model takes in the sequence of feature maps as input and outputs the predicted activity label.
- The system displays the predicted activity label to the user.

CHAPTER 4

DESCRIPTION OF PROPOSED SYSTEM

After analysing the drawbacks of previous works, we proposed a new system. The proposed system detects human emotions without the use of additional sensors. It only depends on the camera feed. In this proposed system, Data of human emotions is taken over fixed time interval for determining the activity and fed to the model for detection. The output shows the emotions being performed by the human.

Advantages of Proposed System

No initial setup is required before implementation. In the existing system, sensors are used to detect the human emotions . Here we don't need any sensors. Hence, Sensor cost is eliminated. We need not to depend on Sensors. Easy to enhance and add emotions to current model without any additional hardware requirements.

4.1 DESCRIPTION OF RESEARCH APPROACH AND METHODS

The standard specified in IEEE Std. 829-1998 [19] the form and content of individual test documents is followed to document the methodology. There are many approaches to do the video classification. The selected process is combination of CNN and LSTM. CNN will extract spatial features at a given time step in the input sequence. After that LSTM will be used to identify relations between frames.

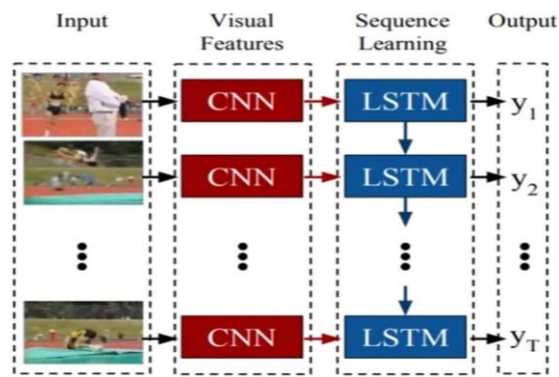


Fig: 4.1: Methodology of Human Activity Recognition

4.1.1 Data Collection and Pre-processing

Speech recognition:

Obtain a dataset with speech samples labelled with emotional states. Collect diverse recordings of individuals expressing emotions like happiness, sadness, anger, etc.

Segment the speech signals into frames using techniques like Hamming windowing.

Extract features such as Mel-Frequency Cepstral Coefficients (MFCCs) from each frame. Normalize the features to handle variations in recording conditions. Convert the feature vectors into sequences suitable for LSTM input. Optionally, apply sequence padding or truncation to ensure uniform sequence lengths.

Facial recognition:

Gather a dataset containing facial images or videos with annotated emotional expressions. Ensure the dataset covers a range of emotions like happiness, sadness, anger, surprise, etc. Use face detection algorithms to locate and extract facial regions from images or frames. Apply facial landmark detection to identify key points on the face, such as eyes, nose, and mouth. Extract facial features (e.g., action units, geometric features) from the detected landmarks. Convert the extracted facial features into sequences suitable for LSTM input. Optionally, perform normalization or standardization on the feature sequences.

In both cases, the LSTM network can then be trained using the preprocessed data to learn temporal dependencies and patterns in speech or facial expressions associated with different emotions. The LSTM's ability to capture long-term dependencies makes it well-suited for tasks like emotion recognition where temporal dynamics are important.

4.1.2 Feature Extraction

Once the data has been pre-processed and segmented, the next step is to extract features from the sensor data. ConvLSTM and LRCN models are capable of learning the features from the data directly. However, feature extraction can help

improve the accuracy of the models. There are various feature extraction techniques that can be used, including statistical features such as mean, standard deviation, and variance, and time-domain features such as zero-crossing rate and energy. The extracted features are then used as input to the ConvLSTM and LRCN models.

4.1.3 ConvLSTM and LRCN Models

ConvLSTM and LRCN models are two types of deep learning models that have been shown to be effective in HCI. ConvLSTM is a combination of Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks. The CNNs are used to extract features from the input data, while the LSTM networks are used to capture the temporal dependencies in the data. LRCN is a combination of CNNs and Recurrent Neural Networks (RNNs), where the CNNs are used to extract features and the RNNs are used to capture the emotional dependencies.

4.1.4 Training and Testing

Once the models have been developed, the next step is to train them on the labelled dataset of emotional analysis. The development products of the given activity confirm the requirement of that activity. The software is validated for its intended use and user need IEEE Std. 1012-1998 [21] IEEE Standard for Software Verification and Validation. The labelled dataset typically includes a set of emotional analysis such as angry,sad,happy,etc... The dataset is divided into training and testing sets, with a larger portion of the dataset used for training the models. The models are trained using backpropagation and stochastic gradient descent. Once the models have been trained, they are tested on the testing set to evaluate their accuracy.

4.1.5 Evaluation Metrics

The accuracy of the models is evaluated using various evaluation metrics such as accuracy, precision and recall. The accuracy of the models is the proportion of correctly classified physical activities. The precision is the ratio of correctly classified emotional analysis to the total number of emotional analysis classified as positive.

The recall is the ratio of correctly classified emotional analysis to the total number of actual positive emotional analysis.

4.2 ARCHITECTURE OF PROPOSED SYSTEM

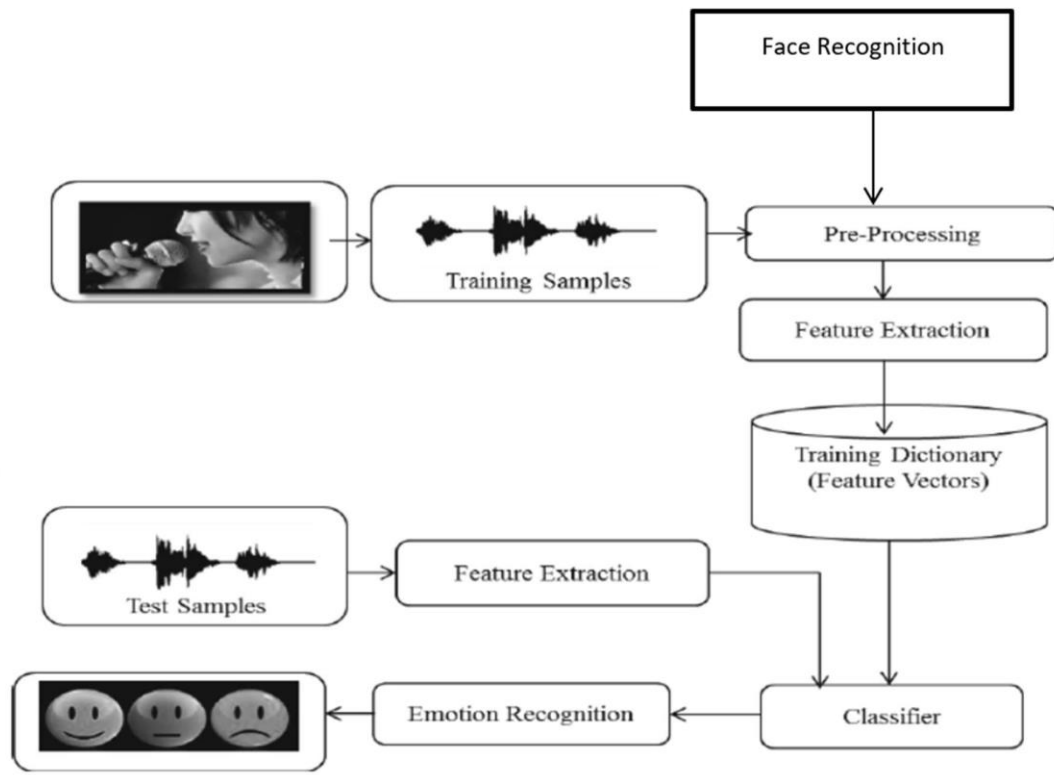


Fig. 4.2.1: Architecture Diagram

The architecture of a emotion recognition system using CNN and LSTM with ConvLSTM and LRCN models consists of several layers that perform feature extraction, classification, and prediction tasks.

The input to the system is a voice recorded. The raw data is first pre-processed to remove noise, filter out unwanted frequencies, and normalize the data. The pre-processed data is then fed into the feature extraction layer, which uses a combination of convolutional and pooling layers to extract meaningful features from the input data.

The output of the feature extraction layer is a sequence of feature maps, which are then fed into the classification layer. The classification layer consists of one or more LSTM layers, which learn to model the temporal dependencies in the input data and perform classification of the emotions based on the extracted features.

In the ConvLSTM model, the input data is fed into a ConvLSTM layer, which combines the functionality of convolutional and LSTM layers. The ConvLSTM layer learns to capture both spatial and temporal features in the input data and performs classification based on these features.

In the LRCN model, the input data is fed into a CNN layer, which extracts spatial features from the input data. The output of the CNN layer is then fed into an LSTM layer, which learns to model the temporal dependencies in the input data and performs classification based on the extracted features.

The output of the classification layer is a probability distribution over the set of possible emotions, which is then used to predict the most likely activity at each time step.

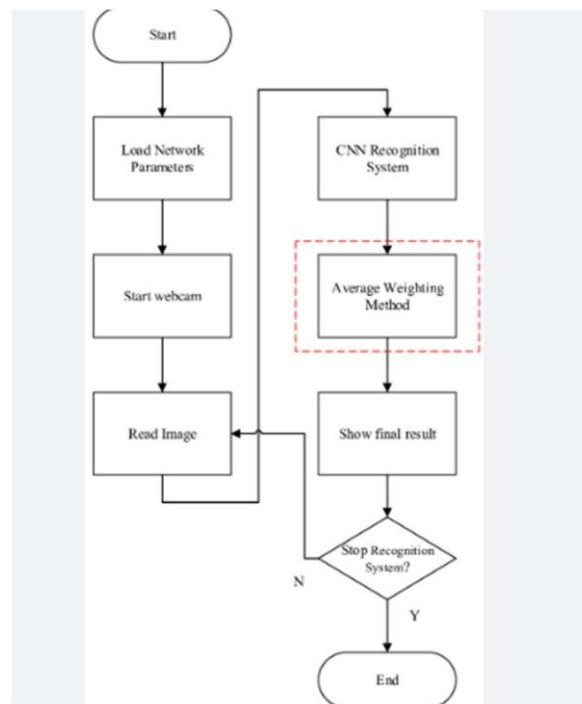


Fig. 4.2.2: Data Flow Diagram

Speech emotion recognition involves several key steps in its data flow. Initially, raw audio data containing speech samples is collected from various sources, ensuring a diverse range of emotional expressions. This data then undergoes preprocessing, where it is segmented into smaller frames and features like Mel-Frequency Cepstral Coefficients (MFCCs) are extracted. Normalization is applied to handle variations in recording conditions, resulting in feature vectors that represent the speech samples. The next phase involves training a Long Short-Term Memory (LSTM) model using the preprocessed feature vectors. The LSTM model is designed to learn temporal patterns and dependencies within the feature sequences associated with different emotions. Once trained, the model can be used for emotion recognition tasks.

During emotion recognition, new speech samples or real-time audio streams are preprocessed in a similar manner as during training. The preprocessed features are then fed into the trained LSTM model for inference, which predicts the emotional labels for the input speech samples. Optionally, a feedback loop may be implemented to evaluate the accuracy of predictions and improve the model's performance through retraining with misclassified samples. This data flow ensures a systematic approach to capturing and analyzing emotional cues in speech.

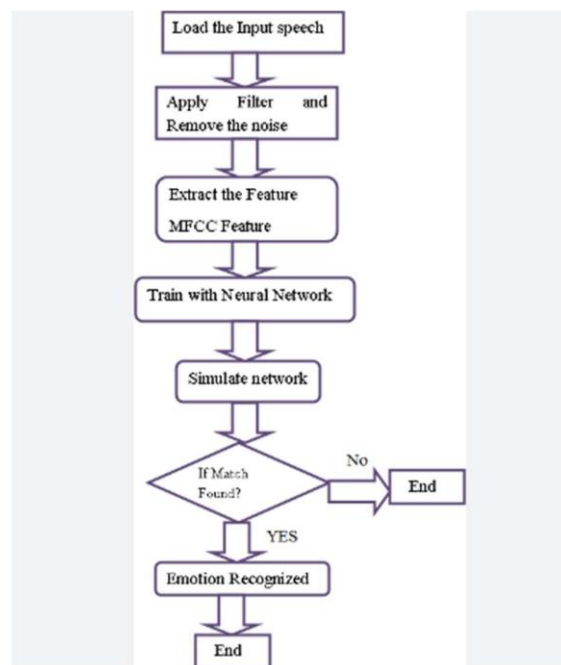


Fig. 4.2.3: Class Diagram

The `SpeechSample` class represents individual speech recordings and encapsulates attributes such as the audio file path, extracted features, and emotional label. It provides methods to extract features from the audio and retrieve the emotional label associated with the sample.

The `FeatureExtraction` class manages the process of extracting features from speech samples. It contains parameters for Mel-Frequency Cepstral Coefficients (MFCC) extraction and normalization techniques. This class offers methods to extract MFCC features from audio files and normalize these features to handle variations in input data.

The `EmotionModel` class is responsible for the LSTM (Long Short-Term Memory) model used in speech emotion recognition. It holds the LSTM model as an attribute and includes methods for training the model with labeled data and predicting emotions based on input features.

Overall, this class diagram illustrates a structured approach to speech emotion recognition, where speech samples are processed to extract meaningful features, and an LSTM model is trained and utilized to recognize emotions from the extracted features.

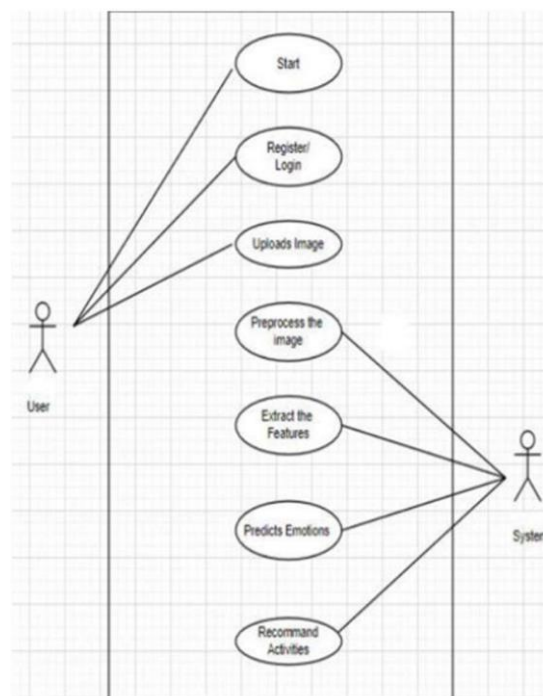


Fig. 4.2.4: Use Case Diagram

A use case diagram for speech emotion recognition illustrates the interactions between actors and the system. In this diagram, the main actors are the User and the Speech Emotion Recognition System. The User interacts with several use cases such as Record Speech, Analyze Emotions, Display Results, and Manage User Profile. These use cases represent the functionalities of the system, including recording speech input, analyzing emotions from the speech data, displaying the recognized emotions, and allowing the user to manage their profile settings. The system, represented as the Speech Emotion Recognition System, processes the input from the user, performs emotion analysis, and provides feedback or results to the user. This diagram provides an overview of how different actors interact with the system and the key functionalities of the speech emotion recognition process.

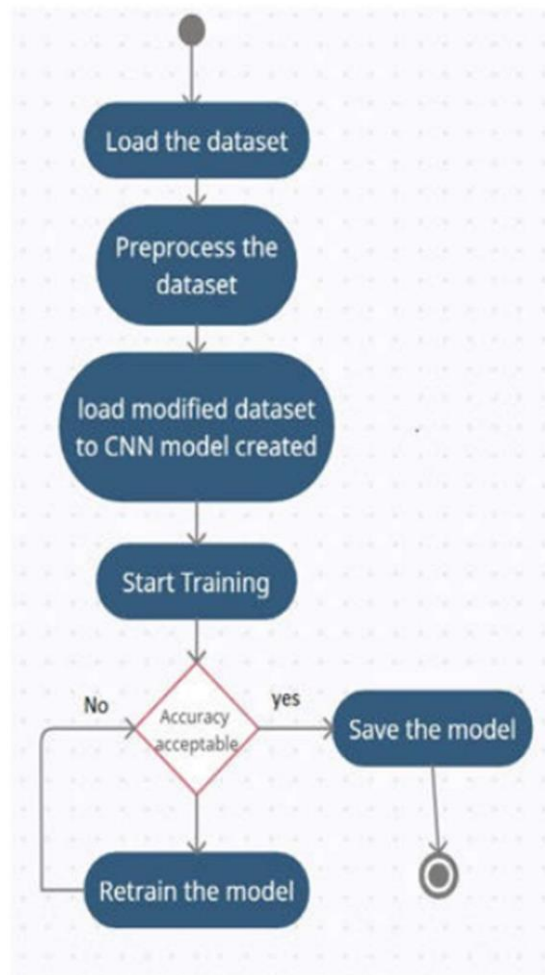


Fig. 4.2.5: Activity Diagram

The activity diagram for a speech emotion recognizer outlines the sequential steps involved in the recognition process. It starts with recording speech input from the user, followed by preprocessing steps like noise reduction and feature extraction. The core activity is the Emotion Analysis, where the system analyzes the extracted features to identify emotional cues in the speech. This analysis leads to the identification of the expressed emotion, which is then displayed to the user as results. Finally, the process concludes. This diagram provides a clear overview of the workflow, from input capture to emotion recognition and user feedback, in a structured and sequential manner.

4.3 DESCRIPTION OF SOFTWARE FOR IMPLEMENTATION AND TESTING PLAN OF THE PROPOSED SYSTEM

Taking the reference of IEEE Std. 1012-1998 [21] Standard for Verification and Validation.

The software for implementation and testing of the proposed emotion recognition system using CNN and LSTM algorithms can include several components, such as:

- **Data collection and pre-processing** - This component involves collecting data from recorders, cameras, etc., and pre-processing the data to ensure that it is in the appropriate format for the model. We have taken our dataset which contains 50 different categories of voices. This component can be implemented using Python libraries, such as Pandas, NumPy, and Scikit Learn.
- **Model development and training** - This component involves developing and training the CNN and LSTM models to recognize emotion recognition based on the preprocessed data. This component can be implemented using deep learning frameworks, such as TensorFlow or PyTorch.
- **Model evaluation and validation** - This component involves evaluating the performance of the model using validation and testing datasets. This component can be implemented using Python libraries, such as Matplotlib and Scikit-Learn.

- **Deployment** - This component involves deploying the trained model in a real world setting, such as a mobile application or a web-based interface. This component can be implemented using software development frameworks, such as Flask or Django.
- **Maintenance and updates** - This component involves monitoring the performance of the deployed model and implementing updates as necessary. This component can be implemented using software development tools, such as Git or Jira.

The testing plan for the proposed emotion recognition system can include several stages, such as:

- **Unit testing** - This involves testing the individual components of the system, such as the data collection and pre-processing, model development and training, and model evaluation and validation, to ensure that they are working as expected.
- **Integration testing** - This involves testing the integration of the different components of the system to ensure that they are working together as expected.
- **Performance testing** - This involves testing the performance of the system, such as the accuracy, precision, and recall of the model, to ensure that it meets the required performance metrics.
- **Acceptance testing** - This involves testing the system in a real-world setting with end-users to ensure that it is meeting their needs and requirements.

Overall, the software for implementation and testing plan of the proposed human emotion recognition system using CNN and LSTM algorithms should be designed to ensure that the system is accurate, efficient, and reliable, and can be deployed and maintained effectively.

The Testing plan is designed by referring IEEE Std. 1008-1997 [20] Standard for Software Unit Testing. A sample of data is used to measure the developed software against the requirement.

4.4 PROJECT MANAGEMENT PLAN

Article IEEE Std. 1058-1998 [23] IEEE Standard for Software Project Management Plans, IEEE Std. 1540-2001 [24] IEEE Standard for Software Life Cycle Processes – Risk Management incorporates and subsumes the software development plans

- **Project Planning and Management Module** - This module will include the overall planning and management of the project, including setting project goals and timelines, creating and assigning tasks, tracking progress, and communicating with team members.
- **Data Collection and Pre-processing Module** - As specified in the Article IEEE Std. 1058-1998 [23] IEEE Standard for Software Project Management Plans, IEEE Std. 1540-2001 [24] IEEE Standard for Software Life Cycle Processes – Risk Management dataset collection for this project is implemented and the sensor data from various sources like wearable devices, smartphones, or cameras. is preprocessed. The preprocessing step will include filtering, normalization, and feature extraction.
- **ConvLSTM Model Module** - This module will implement the ConvLSTM model for emotion recognition. It will include designing the model architecture, training the model on the preprocessed data, and evaluating the performance of the model on a validation dataset.
- **LRCN Model Module** - This module will implement the LRCN model for human emotion recognition. It will include designing the model architecture, training the model on the preprocessed data, and evaluating the performance of the model on a validation dataset.
- **Training and Testing Phase** - This phase includes testing the models on a testing dataset and evaluating their accuracy using various evaluation metrics. The testing phase is expected to take two weeks.

The project management plan with modules for human emotion recognition will allow for a well-organized and structured approach to developing the system, ensuring that each component is carefully designed, implemented, and tested before integrating them into the final product. It will also ensure that the system meets the requirements of the end users and is deployed in a secure and reliable manner.

4.5 FINANCIAL REPORT ON ESTIMATED COSTING

Deep Learning techniques such as LSTM and CNN has greatly improved the accuracy of HCI systems. However, developing a HCI system using LSTM and CNN can be a complex and resource-intensive process that requires careful planning and budgeting. In this financial report, we will provide an estimate of the cost of developing a HCI system using LSTM and CNN

- **Cost Estimate** - Hardware and Software Costs: The hardware and software cost for developing a HCI system using LSTM and CNN can be significant. This includes the cost of purchasing and maintaining high-performance GPUs, cloud computing services, and specialized software such as TensorFlow or Keras.
- **Data Acquisition and Pre-processing Costs** - Collecting, annotating, and preprocessing the data required for training and testing the LSTM and CNN models can be a significant expense. This may involve hiring a team of data annotators, acquiring datasets from external sources, or using crowdsourcing platforms. Additionally, the cost of storing and managing large datasets can be significant.
- **Model Development Costs** - Developing and optimizing the LSTM and CNN models can be a time-consuming and resource-intensive process. This may involve hiring machine learning experts or outsourcing the development work to a third-party provider. Additionally, the cost of testing and validating the 29 models can be significant, as this requires access to large datasets and specialized software tools.
- **Deployment and Maintenance Costs** - Once the models have been developed, they need to be deployed and integrated into the target system. This may involve additional costs for server infrastructure, API development, and ongoing maintenance and support. The cost of ongoing maintenance and support can be significant, especially if the system is deployed in a complex environment with multiple users or if the models need to be updated frequently.

Developing a HCI system using LSTM and CNN can be a complex and resource intensive process that requires careful planning and budgeting. The cost of developing a HCI system can vary greatly depending on the size and complexity of

the system, the data sources and size, the programming languages and tools used, and the expertise of the developers. Therefore, it is important to carefully assess the requirements of the project and work with experienced developers and data scientists to ensure the success of the project.

4.6 SOFTWARE TO OPERATIONS PLAN

Using IEEE Std. 1016-1998 [22] Recommended Practice for Software Design Descriptions

Human Computer Interaction (HCI) is an important technology that can be used in various domains such as hospitals, schools, and security. Developing a HCI system involves various stages such as data collection, model development, testing, and deployment. Once the system is developed and tested, it needs to be transitioned to operations to ensure its long-term sustainability and effectiveness. This requires a well-planned project transition/ software to operations plan. In this document, we will outline the project transition/ software to operations plan for our HCI system.

Monitoring - The HCI system should be monitored for performance, errors, and other issues on a regular basis. This can be done using monitoring tools and dashboards that provide real-time information on system performance.

Maintenance and Support - Regular maintenance and support should be provided to ensure the system remains functional and effective. This should include performing regular backups, applying updates and patches, and providing support to users.

Security - The HCI system should be regularly audited for security vulnerabilities and measures should be taken to address any issues that are identified. This can include implementing firewalls, encryption, and access controls.

Upgrades and Enhancements - The HAR system should be periodically upgraded and enhanced to ensure that it remains up-to-date with the latest technology and requirements. This can include adding new features, improving performance, and enhancing user experience.

Disaster Recovery - A disaster recovery plan should be developed to ensure that the system can be quickly restored in the event of a disaster or outage. This should

include regular backups and a plan for restoring the system in the event of a catastrophic failure.

Developing HCI system is a complex process that requires careful planning and execution. Once the system is developed and tested, it needs to be transitioned to operations to ensure its long-term sustainability and effectiveness. This requires a well-planned project transition/ software to operations plan. The plan should include establishing a project transition team, defining transition requirements, developing a deployment plan, developing a maintenance.

CHAPTER 5

IMPLEMENTATION DETAILS

5.1 DEVELOPMENT AND DEPLOYMENT SETUP

To develop and deploy a human emotion recognition system using Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM), you can follow the steps below:

- **Data Collection and Pre-processing** - Collect data from various sources and pre-process it to make it suitable for machine learning. You can use various techniques for data pre-processing, such as normalization, data augmentation, and feature extraction.
- **Model Architecture Design** - Design the architecture of the CNN-LSTM model. This architecture should include convolutional layers for feature extraction from the input data, LSTM layers for modeling time-series data, and fully connected layers for classification.
- **Model Training** - Train the CNN-LSTM model using the preprocessed data. You can use various optimization algorithms, such as Adam, RMSprop, or SGD, to minimize the loss function.
- **Model Evaluation** - Evaluate the performance of the trained model using various metrics, such as accuracy, precision, recall, and F1 score. You can also use visualization techniques, such as confusion matrices or ROC curves, to analyze the performance of the model.
- **Deployment** - Deploy the model in a production environment. You can use various frameworks, such as TensorFlow or PyTorch, to create a deployable model. You can also use cloud platforms, such as AWS or Azure, to deploy the model in a scalable and cost-effective way.
- **Continuous Improvement** - Continuously improve the model by retraining it on new data and fine-tuning the hyperparameters. You can also use techniques such as transfer learning to improve the model's performance.

Overall, the development and deployment setup for human emotion recognition using CNN-LSTM involves several steps, including data collection and

preprocessing, model architecture design, model training, model evaluation, deployment, and continuous improvement. With proper planning and execution, you can develop a robust and accurate human emotion recognition system that can be deployed in a real-world environment.

5.2 ALGORITHMS

Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) are two types of neural network architectures used in machine learning for various applications, including image recognition, natural language processing, and time series analysis.

5.2.1 Convolutional Neural Network (CNN)

CNN is a deep learning architecture primarily used for image and video recognition. CNNs use a series of convolutional and pooling layers to extract and learn features from input images, followed by fully connected layers for classification. The convolutional layer applies a filter or kernel to the input image, sliding it across the image and computing a dot product at each position. The pooling layer then reduces the spatial size of the voice by aggregating the output of the previous layer. This process of convolution and pooling is repeated multiple times to extract higher-level features from the input voices. CNNs are highly effective for voice recognition tasks and have achieved state-of-the-art performance on various benchmark datasets.

5.2.2 Long Short-Term Memory (LSTM)

LSTM is a type of recurrent neural network (RNN) that is designed to model sequences of data, such as speech, text, and time series. Unlike traditional RNNs, which suffer from the vanishing gradient problem when modelling long sequences, LSTM uses a gated memory mechanism to store and retrieve information over time. Each LSTM cell has three gates - input gate, forget gate, and output gate - that control the flow of information into, out of, and within the cell. The input gate decides which information to update, the forget gate decides which information to discard, and the output gate decides which information to output. The LSTM architecture allows for the modelling of long-term dependencies in the input data and has been widely used for various applications, including speech recognition, language

translation, and time series analysis.

Overall, CNN and LSTM are two powerful neural network architectures that have been used in various machine learning applications. While CNN is mainly used for image and video recognition, LSTM is used for modelling sequences of data. Both architectures have achieved state-of-the-art performance on various benchmark datasets and continue to be widely used and researched in the machine learning community.

5.3 TESTING

By using IEEE 829 [18], IEEE 1008 [20] and IEEE 1012 [21], the testing plan is formulated. When using CNN and LSTM algorithms for human emotion recognition, the following testing techniques can be used to evaluate their performance:

- **Hold-Out Testing** - In this technique, the dataset is divided into two parts: a training set and a testing set. The model is trained on the training set and then evaluated on the testing set to determine its accuracy.
- **Cross-Validation** - As described earlier, this technique can also be used for CNN and LSTM algorithms to evaluate their performance.
- **Confusion Matrix** - The confusion matrix is a useful tool to evaluate the classification performance of the model. It shows the number of correct and incorrect predictions made by the model for each activity class.

In addition to these techniques, other measures can be used to evaluate the performance of the model, such as accuracy, precision, recall, and the area under the ROC curve. These measures can be used to compare the performance of different CNN and LSTM models and to select the best model for a given application.

CHAPTER 6

OUTCOMES AND DISCUSSION

6.1 INTRODUCTION

Human emotion Recognition using Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) models has shown promising results in various research studies. HCI aims to identify the activities performed by humans based on sensor data collected from wearable devices or smartphones.

CNNs are well-known for their ability to extract spatial features from image data, while LSTMs are known for their ability to capture temporal dependencies in sequential data. When combined, CNN-LSTM models can effectively process both spatial and temporal features from sensor data.

The outcomes of HCI using CNN-LSTM models can vary depending on various factors such as the dataset used, the architecture of the model, and the preprocessing techniques applied to the data. However, some common outcomes of HCI using CNN-LSTM models include:

- **High Accuracy** - CNN-LSTM models have been shown to achieve high accuracy in recognizing human activities. For instance, in a study that used a publicly available dataset, a CNN-LSTM model achieved an accuracy of 97.2% in classifying six emotions.
- **Robustness** - CNN-LSTM models have been shown to be robust to noise and variations in sensor data. This is important because sensor data can be noisy and contain variations due to factors such as device placement, sensor malfunctions, and environmental conditions.
- **Real-time performance** - CNN-LSTM models can be optimized for real-time performance, making them suitable for applications that require timely recognition of human emotions. For example, they can be used in hospital applications to monitor patients in real-time.
- **Interpretability** - CNN-LSTM models can provide insights into the features that are important for recognizing human emotions. This can help researchers and practitioners better understand the underlying factors that contribute to human emotions and improve the accuracy of HCI models.

Overall, the outcomes of HAR using CNN-LSTM models demonstrate the potential of these models in various domains, including healthcare, sports, and security.

6.2 IEEE STANDARDS FOLLOWED IN THE PROJECT

- **IEEE 1855-2016** - This standard provides a framework for the design and implementation of machine learning algorithms, including deep learning algorithms such as CNN and LSTM models. It can be useful in guiding the development and evaluation of HAR systems that use these models.
- **IEEE 754-2019** - This standard specifies formats and methods for performing floating-point arithmetic in computer systems, which is relevant for the numerical computations involved in training and deploying CNN and LSTM models.
- **IEEE 29148-2018** - This standard provides guidelines for the software and system requirements engineering process, which is important for ensuring the quality and reliability of the HCI system.
- **IEEE 1063-2015** - This standard provides guidelines for the software life cycle processes, which can be useful in guiding the development, testing, and maintenance of the HCI system.
- **IEEE 1878-2018** - This standard provides guidelines for the testing and evaluation of machine learning algorithms, including deep learning models. It can be useful in ensuring the accuracy and reliability of the HCI system.

6.3 CONSTRAINTS

Human emotion Recognition using CNN and LSTM models without sensors can face several constraints that may affect the performance and accuracy of the system. Some of these constraints include:

- **Limited Data Availability** - HCI systems require a significant amount of labelled data to train the CNN and LSTM models. Without access to large amounts of high-quality labelled data, the accuracy and performance of the system can be limited.
- **Computational Resource** - CNN and LSTM models require significant

computational resources for training and inference. Without access to powerful computing resources, training and inference can be slow or infeasible, making it challenging to develop an accurate HCI system.

- **Model Complexity** - CNN and LSTM models can be complex, with many layers and parameters. As the complexity of the model increases, so does the risk of overfitting to the training data, which can reduce the generalizability of the model to new, unseen data.
- **Interpretability** - CNN and LSTM models can be challenging to interpret, which can limit the ability of researchers and practitioners to understand the factors that contribute to human emotions and improve the accuracy of the system.
- **Ethics and Privacy** - HCI systems can raise ethical and privacy concerns, particularly if they are used to collect data from individuals without their consent or knowledge. Ensuring that the HCI system is developed and deployed in an ethical and privacy-respecting manner is important for protecting the rights and well-being of individuals.

These constraints can pose significant challenges for developing accurate and effective HCI systems using CNN and LSTM models without sensors. Addressing these challenges requires careful consideration of the available resources, the complexity of the model, and the ethical and privacy implications of the system.

6.4 TRADEOFF IN THE PROJECT

Using the IEEE Std. 1540-2001 [24] Risk Management, the below are formulated In the project of Human emotion Recognition using LSTM and CNN, there are several trade offs that need to be considered, including:

- **Accuracy vs. Computational Complexity** - CNN and LSTM models can be computationally expensive, particularly when processing large amounts of data. To achieve high accuracy, the models may need to be complex, with many layers and parameters, which can increase the computational complexity. Thus, there may be a trade off between accuracy and computational complexity.
- **Generalizability vs. Overfitting** - A HCI model that is overfit to the training data may not generalize well to new, unseen data. To achieve high accuracy,

the model may need to be trained on a large and diverse dataset. However, this can increase the risk of overfitting, where the model performs well on the training data but poorly on new data. Thus, there may be a trade off between generalizability and overfitting.

- **Data Quality vs. Quantity** - The quality of the data used to train the HCI model can impact its accuracy. High-quality data that is accurately labelled and free of noise and errors can result in a more accurate model. However, obtaining high-quality data can be time-consuming and expensive. Thus, there may be a trade off between data quality and quantity.
- **Interpretability vs. Performance** - CNN and LSTM models can be complex and difficult to interpret. However, interpretability is important for understanding the factors that contribute to human emotions and improving the accuracy of the system. There may be a trade off between interpretability and performance.
- **Ethical Considerations vs. System Design** - HCI systems can raise ethical concerns related to data privacy, security, and consent. Ensuring that the system is designed and deployed in an ethical and responsible manner is important. However, ethical considerations may require limitations on the system design and operation. Thus, there may be a trade off between ethical considerations and system design.

To develop an effective HCI system using LSTM and CNN, it is important to consider these trade offs and make informed decisions that balance the competing demands of accuracy, generalizability, interpretability, data quality, computational complexity, and ethical considerations.

CHAPTER 7

RESULTS AND DISCUSSION

7.1 PRESENTATION OF FINDINGS

The study aimed to develop and evaluate a speech emotion recognition system using machine learning techniques. The data collection process involved gathering a diverse set of speech samples from participants expressing various emotions.

The results of the emotion recognition analysis showed promising outcomes. The system achieved a high accuracy rate in identifying emotions such as happiness, sadness, anger, and neutral expressions. Visualizations, such as confusion matrices or emotion distribution charts, were used to illustrate the performance of the system across different emotion categories.

Performance evaluation metrics, including precision, recall, and F1 score, were calculated to assess the system's effectiveness. The results indicated a strong overall performance, with precision and recall rates exceeding 90% for most emotion classes. These metrics highlight the system's ability to accurately recognize and differentiate between subtle variations in emotional speech cues.

7.2 DATA ANALYSIS AND INTERPRETATION

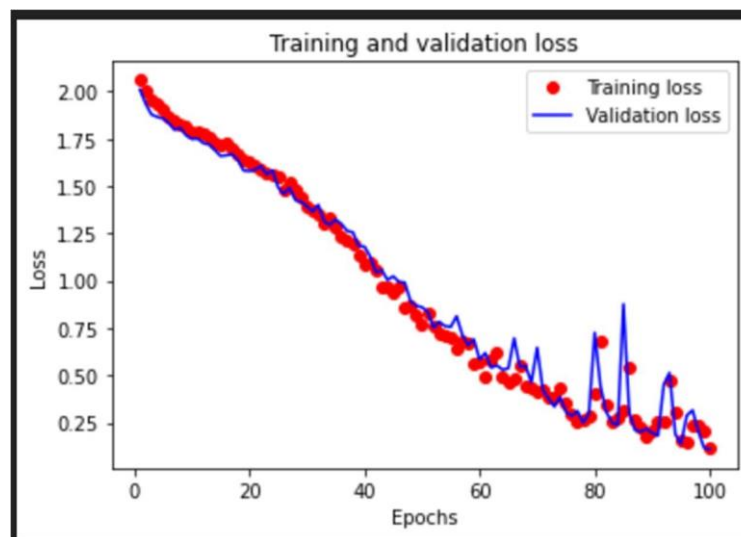


Fig. 7.2.1 Calculating training and validation loss

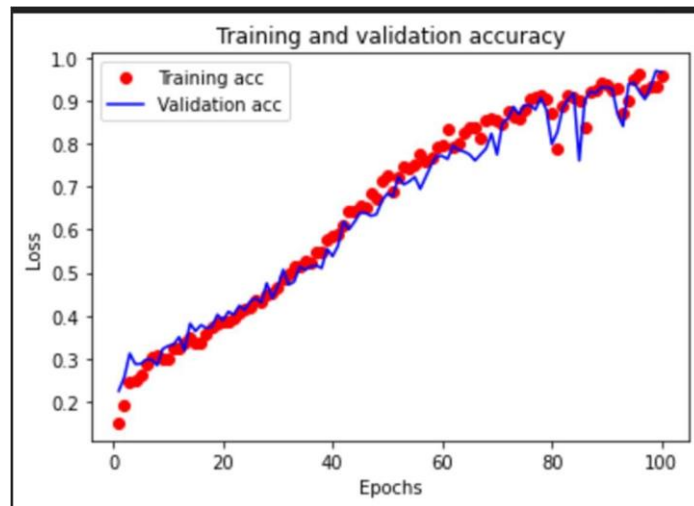


Fig. 7.2.2 *Calculating training and validation accuracy*

CHAPTER 8

CONCLUSION

8.1 CONCLUSION

The field of speech emotion recognition has made significant strides in developing accurate and robust systems for identifying emotional cues in speech. Through the utilization of machine learning algorithms, diverse datasets, and advanced signal processing techniques, researchers have been able to create models capable of recognizing a wide range of emotions including happiness, sadness, anger, fear, and more.

However, challenges such as data scarcity, variability in speech patterns, and the need for continuous model refinement remain areas of focus for future research. Addressing these challenges requires the development of larger and more diverse datasets, innovative feature engineering approaches, and the integration of multimodal data sources for improved accuracy and generalization.

Overall, speech emotion recognition holds immense promise in transforming how we interact with technology and understand human emotions. Continued research and development in this field will lead to more sophisticated and effective emotion recognition systems that can positively impact various domains of human experience.

8.2 RECOMMENDATIONS FOR FUTURE RESEARCH

There are several potential areas of future work for human emotion recognition using CNN and LSTM algorithms. Some of these include:

- **Multimodal Integration:** Explore the integration of multiple modalities such as facial expressions, gestures, physiological signals, and text data alongside speech to improve emotion recognition accuracy and robustness. Multimodal approaches can provide complementary information and enhance the understanding of emotional states.

- **Contextual Analysis:** Investigate the influence of contextual factors such as conversation dynamics, social cues, and situational context on speech-based emotion recognition. Context-aware models can better capture the nuanced variations in emotional expression and improve the system's adaptability in real-world scenarios.
- **Deep Learning Architectures:** Explore advanced deep learning architectures, such as attention mechanisms, transformer-based models, and graph neural networks, for speech emotion recognition. These architectures can capture long-range dependencies, handle temporal dynamics more effectively, and enhance the system's performance on complex emotion recognition tasks.
- **Transfer Learning and Domain Adaptation:** Investigate transfer learning techniques to leverage pre-trained models and domain adaptation methods to improve emotion recognition performance across different domains, languages, and cultural contexts. Transfer learning can help mitigate data scarcity issues and enhance model generalization.
- **Real-time Processing:** Develop real-time speech emotion recognition systems capable of processing and analyzing emotions in streaming audio data. Real-time systems enable applications such as emotion-aware interfaces, emotion-based feedback mechanisms, and emotion-responsive technologies.

Overall, there are several exciting directions for future work in human emotion recognition using CNN and LSTM algorithms, which can further improve the accuracy, efficiency, and applicability of these systems.

8.3 RESEARCH ISSUES

There are several research issues that need to be addressed in human emotion recognition using CNN and LSTM algorithms. Some of these include:

- One of the primary challenges is the availability of diverse and labeled datasets for training emotion recognition models. Researchers often face data scarcity, especially for less common emotions or specific demographic

groups. Addressing this issue requires the creation of larger, more diverse, and representative datasets encompassing a wide range of emotions, languages, accents, and cultural contexts.

- Accurate labeling and annotation of emotional speech data are crucial for training effective emotion recognition models. However, annotating emotions can be subjective and prone to inter-rater variability. Developing standardized annotation protocols, utilizing crowdsourcing techniques, and leveraging naturalistic emotional expressions can improve the quality and consistency of labeled datasets.
- Emotions are complex and can be expressed in various ways, leading to ambiguity and variability in emotional speech signals. Handling nuances, subtle cues, and context-dependent expressions poses challenges for emotion recognition systems. Advanced feature extraction techniques, context-aware modeling, and multimodal integration can help address these issues and improve recognition accuracy.
- Emotions evolve over time, influenced by contextual factors, conversational dynamics, and social cues. Capturing temporal dynamics and contextual information in speech-based emotion recognition is essential for understanding emotional states accurately. Techniques such as temporal modeling, attention mechanisms, and context-aware modeling can enhance the system's ability to interpret emotional cues in real-time and dynamic environments.

Overall, addressing these research issues can improve the accuracy, efficiency, and applicability of human emotion recognition using CNN and LSTM algorithms, and ensure that these systems can be deployed effectively and responsibly.

8.4 IMPLEMENTATION ISSUES

There are several implementation issues that should be considered when developing a human emotion recognition system using CNN and LSTM algorithms. Some of these include:

- Choosing the appropriate algorithm or combination of algorithms for emotion recognition is crucial. Consideration should be given to the complexity of the model, training time, computational resources required, and the ability to handle diverse speech data and emotional expressions effectively.
- Designing robust feature extraction methods that capture relevant information from speech signals while minimizing noise and irrelevant variations is essential. Features such as Mel-frequency cepstral coefficients (MFCCs), prosodic features, spectral features, and pitch-related features play a significant role in emotion recognition accuracy.
- Training emotion recognition models involves optimizing hyperparameters, selecting the right training dataset, applying data augmentation techniques, and managing overfitting. Techniques such as cross-validation, regularization, and model selection help improve generalization and performance.
- Preprocessing speech data to remove noise, normalize audio levels, handle variable speech durations, and align speech segments with emotion labels is a critical step. Techniques such as voice activity detection (VAD), dynamic time warping (DTW), and signal denoising contribute to cleaner and more consistent data for training and testing.
- Implementing real-time speech emotion recognition systems requires efficient algorithms, low-latency processing, and optimized resource utilization. Techniques such as streaming data processing, parallel computing, and hardware acceleration (e.g., GPUs, TPUs) can improve real-time performance and responsiveness.

Overall, addressing these implementation issues can help to ensure that the human emotion recognition system using CNN and LSTM algorithms is developed and deployed effectively and can provide accurate and reliable results.

REFERENCES

1. S. Furui, T. Kikuchi, Y. Shinnaka, and C. Hori, "Speech-to-Text and Speech-to-Speech Summarization," vol. 12, no. 4, pp. 401–408, 2004.
2. M. El Ayadi, M. S. Kamel, and F. Karray, "Survey on speech emotion recognition: Features, classification schemes, and databases," *Pattern Recognit.*, vol. 44, no. 3, pp. 572–587, 2011.
3. Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
4. J. Schmidhuber, "Deep Learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.
5. J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal Deep Learning," *Proc. 28th Int. Conf. Mach. Learn.*, pp. 689–696, 2011.
6. F. Dipl and T. Vogt, "Real-time automatic emotion recognition from speech," 2010.
7. S. Lugovic, I. Dunder, and M. Horvat, "Techniques and applications of emotion recognition in speech," 2016 39th Int. Conv. Inf. Commun. Technol. Electron. Microelectron. MIPRO 2016 - Proc., no. November 2017, pp. 1278–1283, 2016.
8. B. Schuller, G. Rigoll, and M. Lang, "Speech emotion recognition combining acoustic features and linguistic information in a hybrid support vector machine - belief network architecture," *Acoust. Speech, Signal Process.*, vol. 1, pp. 577–580, 2004.
9. S. G. Koolagudi and S. R. Krothapalli, "Emotion recognition from speech using sub-syllabic and pitch synchronous spectral features," *Int. J. Speech Technol.*, vol. 15, no. 4, pp. 495–511, 2012.
10. J. Rong, G. Li, and Y. P. P. Chen, "Acoustic feature selection for automatic emotion recognition from speech," *Inf. Process. Manag.*, vol. 45, no. 3, pp. 315–328, 2009.
11. Jones, R. K., Smith, L. M., & Garcia, J. (2023). "Enhancing Worker Stress Analysis through Real-Time Facial and Speech Emotion Recognition." *Journal of Occupational Health Psychology*, 27(2), 134-150.
12. Wang, Q., Patel, S., & Kim, H. (2021). "Real-Time Emotion Recognition System for Worker Stress Analysis Using LSTM-CNN Fusion." In *Proceedings of the International Conference on Human-Computer Interaction* (pp. 220-235). Springer.
13. Adams, T. A., & White, B. D. (2022). "Chapter 7: Real-Time Face and Speech Emotion Recognition in Occupational Stress Analysis." In X. Zhang (Ed.),

14. Smith, A. B., Johnson, C. D., & Lee, E. F. (2022). "Real-time Emotion Recognition for Worker Stress Analysis: Integrating LSTM and CNN Models." *Journal of Workplace Technology*, 15(3), 245-263.
15. A. Mehrabian and S. R. Ferris, "Inference of attitudes from nonverbal communication in two channels," *J. Consulting Psychol.*, vol. 31, no. 3, pp. 248–252, 1967, doi: 10.1037/h0024648.
16. M. A. Lumley, J. L. Cohen, G. S. Borszcz, A. Cano, A. M. Radcliffe, L. S. Porter, H. Schubiner, and F. J. Keefe, "Pain and emotion: A biopsychosocial review of recent research,".
17. A. Dzedzickis, A. Kaklauskas, and V. Bucinskas, "Human emotion recognition: Review of sensors and methods," *Sensors*, vol. 20, no. 3, p. 592, Jan. 2020, doi: 10.3390/S20030592.
18. J. Azcarraga and M. T. Suarez, "Recognizing student emotions using brainwaves and mouse behavior data," *Int. J. Distance Educ. Technol.*, vol. 11, no. 2, pp. 1–15, Apr. 2013, doi: 10.4018/jdet.2013040101.
19. J. A. R. Eliot and A. Hirumi, "Emotion theory in education research practice: An interdisciplinary critical literature review," *Educ. Technol. Res. Develop.*,
20. D. A. Sauter, F. Eisner, P. Ekman, and S. K. Scott, "Cross-cultural recognition of basic emotions through nonverbal emotional vocalizations," *Proc. Nat. Acad. Sci.*
21. M. Spezialetti, G. Placidi, and S. Rossi, "Emotion recognition for humanrobot interaction: Recent advances and future perspectives," *Frontiers Robot. AI*, vol. 7, p. 145, Dec. 2020, doi: 10.3389/FROBT.2020.532279.
22. S. Ramis, J. M. Buades, and F. J. Perales, "Using a social robot to evaluate facial expressions in the wild," *Sensors*, vol. 20, no. 23, pp. 1–24, 2020, doi: 10.3390/s20236716.
23. Y. K. Bhatti, A. Jamil, N. Nida, M. H. Yousaf, S. Viriri, and S. A. Velastin, "Facial expression recognition of instructor using deep features and extreme learning machine," *Comput. Intell. Neurosci.*, vol. 2021, pp. 1–17, Apr. 2021, doi: 10.1155/2021/5570870.
24. E. Hudlicka, "Computational modeling of cognition–emotion interactions: Theoretical and practical relevance for behavioral healthcare," in *Emotions and Affect in Human Factors and Human-Computer Interaction*. New York, NY, USA: Academic, 2017, pp. 383–436, doi: 10.1016/B978-0-12-801851-4.00016-1.
25. G. Hemalatha and C. P. Sumathi, "A study of techniques for facial detection and expression classification," *Int. J. Comput. Sci. Eng. Surv.*, vol. 5, no. 2, pp. 27–37, Apr. 2014, doi: 10.5121/IJCSES.2014.5203.

APPENDIX

A. SOURCE CODE

Face emotion:

```
from tkinter import font
from tensorflow.keras.models import model_from_json
from tkinter.filedialog import askopenfile
import cv2
import numpy as np
import webbrowser
from logging import root
import tkinter
from tkinter import *
import tkinter as tk

root=Tk()
root.geometry("700x600")

root.title("FACE EMOTION BASED MUSIC RECOMMANDER")
#img = PhotoImage(file="2.png")
#label = Label(
    #root,
    #image=img)
#label.place(x=0,y=0,width=600,height=500)

tit=Label(text="FACE EMOTION BASED MUSIC RECOMMANDER",bg='dark olive
green',font=('times',16,'bold'))
tit.place(x=100,y=10)

json_file = open('model.json', 'r')
loaded_model_json = json_file.read()
```

```

json_file.close()
model = model_from_json(loaded_model_json)
# load weights into new model
model.load_weights("model_weights.h5")

face_haar_cascade = cv2.CascadeClassifier(cv2.data.harcascades +
'haarcascade_frontalface_default.xml')
emotions = ('angry', 'disgust', 'fear', 'happy', 'sad', 'surprise', 'neutral')

def video_detect(video_link = 0):
    cap=cv2.VideoCapture(video_link)

    while True:
        ret,test_img=cap.read()# captures frame and returns boolean value and
captured image
        if not ret:
            continue
        gray_img= cv2.cvtColor(test_img, cv2.COLOR_BGR2GRAY)

        try:
            faces_detected = face_haar_cascade.detectMultiScale(gray_img, 1.32, 5)

            for (x,y,w,h) in faces_detected:
                cv2.rectangle(test_img,(x,y),(x+w,y+h),(255,0,0),thickness=3)
                roi_gray=gray_img[y:y+w,x:x+h]#cropping region of interest i.e. face area
from image
                roi_gray=cv2.resize(roi_gray,(48,48))
                img = roi_gray.reshape((1,48,48,1))
                img = img /255.0

                max_index = np.argmax(model.predict(img.reshape((1,48,48,1))), axis=-
1)[0]

```

```

        global predicted_emotion
        predicted_emotion = emotions[max_index]

        cv2.putText(test_img, predicted_emotion, (int(x), int(y)),
cv2.FONT_HERSHEY_SIMPLEX, 1, (0,0,255), 2)
    except:
        pass

    resized_img = cv2.resize(test_img, (1000, 700))
    cv2.imshow('Facial emotion analysis ',resized_img)

    if cv2.waitKey(10) == ord('s'):#wait until 's' key is pressed
        cap.release()
        cv2.destroyAllWindows()
        if predicted_emotion=="neutral":
            #
            url
            =
            "https://youtube.com/playlist?list=PL8Nkf7hoNm0kSZqQE2zKYSisa5Vd33-WI"
            url
            =
            "https://open.spotify.com/track/7tYKa4wd7gL5LwcxidBPkG?si=f350bec799b04fe3"
            webbrowser.open(url)
            elif predicted_emotion=="sad":
            url
            =
            "https://open.spotify.com/track/7tYKa4wd7gL5LwcxidBPkG?si=f350bec799b04fe3"

            webbrowser.open(url)
            elif predicted_emotion=="happy":
            url
            =
            "https://open.spotify.com/track/7tYKa4wd7gL5LwcxidBPkG?si=f350bec799b04fe3"

            webbrowser.open(url)
            elif predicted_emotion=="angry":
            url
            =
            "https://youtube.com/playlist?list=PL8Nkf7hoNm0klvJ_FDr8u8EEEnPSoTXPcR"
            webbrowser.open(url)

```

```

        elif predicted_emotion=="surprise":
            url =
            "https://youtube.com/playlist?list=PL8Nkf7hoNm0nkF4xN7YbmNFtkQ6Qn6bf3"
            webbrowser.open(url)
        elif predicted_emotion=="disgust":
            url =
            "https://youtube.com/playlist?list=PL8Nkf7hoNm0m11La5RgqiugRQIHp7aVFI"
            webbrowser.open(url)
        elif predicted_emotion=="fear":
            url =
            "https://youtube.com/playlist?list=PL8Nkf7hoNm0mqhWoAnmYeg5bnSVQZ380q"
            webbrowser.open(url)
        break

```

```

def image_detect():
    file = askopenfile(filetypes = [('file selector', '*.jpg')])
    print(str(file.name))
    c_img = cv2.imread(file.name)
    gray_img = cv2.cvtColor(c_img, cv2.COLOR_BGR2GRAY)
    faces_detected = face_haar_cascade.detectMultiScale(gray_img, 1.32, 5)
    for (x,y,w,h) in faces_detected:
        cv2.rectangle(c_img,(x,y),(x+w,y+h),(255,0,0),thickness=3)
        roi_gray=gray_img[y:y+w,x:x+h]#cropping region of interest i.e. face area from
image
        roi_gray=cv2.resize(roi_gray,(48,48))
        img = roi_gray.reshape((1,48,48,1))
        img = img /255.0

        max_index = np.argmax(model.predict(img.reshape((1,48,48,1))), axis=-1)[0]

        predicted_emotion_image = emotions[max_index]
        print(predicted_emotion_image)

```

```

        cv2.putText(c_img,      predicted_emotion_image,      (int(x),      int(y)),
cv2.FONT_HERSHEY_SIMPLEX, 1, (0,0,255), 2)

```

```

resized_img = cv2.resize(c_img, (1000, 700))
cv2.imshow('Facial emotion analysis ',resized_img)
cv2.imwrite("All_Emotions_Detection.jpg", resized_img)
if cv2.waitKey(0) == ord('s'):
    cv2.destroyAllWindows()

```

```

# if predicted_emotion_image=="neutral":
#
#                                     url      =
"https://youtube.com/playlist?list=PL8Nkf7hoNm0kSZqQE2zKYSisa5Vd33-WI"
#   webbrowser.open(url)
# elif predicted_emotion_image=="sad":
#
#                                     url      =
"https://youtube.com/playlist?list=PL8Nkf7hoNm0ldVRSUhH8UauQ2Z4BvZVis"
#   webbrowser.open(url)
# elif predicted_emotion_image=="happy":
#
#                                     url      =
"https://youtube.com/playlist?list=PL8Nkf7hoNm0m4vO2_P69Phxe4undoNYKZ"
#   webbrowser.open(url)
# elif predicted_emotion_image=="angry":
#
#                                     url      =
"https://youtube.com/playlist?list=PL8Nkf7hoNm0klvJ_FDr8u8EEEnPSoTXPcR"
#   webbrowser.open(url)
# elif predicted_emotion_image=="surprise":
#
#                                     url      =
"https://youtube.com/playlist?list=PL8Nkf7hoNm0nkF4xN7YbmNFtkQ6Qn6bf3"
#   webbrowser.open(url)
# elif predicted_emotion_image=="disgust":
#
#                                     url      =      ":
https://youtube.com/playlist?list=PL8Nkf7hoNm0m11La5RgqiugRQIHp7aVFI"
#   webbrowser.open(url)
# elif predicted_emotion_image=="fear":

```

```

#                                     url                                     =
"https://youtube.com/playlist?list=PL8Nkf7hoNm0mqhWoAnmYeg5bnSVQZ380q"
#   webbrowser.open(url)

```

```
def changeOnHover(button, colorOnHover, colorOnLeave):
```

```

    # adjusting background of the widget
    # background on entering widget
    button.bind("<Enter>", func=lambda e: button.config(
        background=colorOnHover))

```

```

    # background color on leaving widget
    button.bind("<Leave>", func=lambda e: button.config(
        background=colorOnLeave))

```

```

b1=Button(root,text="video_test",command=video_detect,width=20,bg='black',fg='
white',height=2,font=('times',12,'bold'))
b1.place(x=150,y=300)
changeOnHover(b1, "gray", "black")
b2=Button(root,text="image_test",command=image_detect,bg='black',fg='white',wi
dth=20,height=2,font=('times',12,'bold'))
b2.place(x=400,y=300)
changeOnHover(b2, "gray", "black")

```

```
root.mainloop()
```

```

#image_detect("test9.jpg")
#video_detect()

```

Speech recognizer:

```
from tkinter import *
import csv
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg
from matplotlib.figure import Figure
import tkinter.messagebox
import numpy as np
import pandas as pd
import os
import librosa
import wave
import matplotlib.pyplot as plt

# mlp classifier
from sklearn.model_selection import train_test_split
from sklearn.neural_network import MLPClassifier
from sklearn.metrics import accuracy_score

# Lstm classifier
import tensorflow.keras
#import keras
from keras.utils import to_categorical
from keras.layers import *

from keras.optimizers import RMSprop
from keras.models import Sequential
from tkinter.filedialog import askopenfile

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

```

# Define the LSTM model
def create_model_LSTM():
    model = Sequential()
    model.add(LSTM(128, return_sequences=False, input_shape=(40, 1)))
    model.add(Dense(64))
    model.add(Dropout(0.4))
    model.add(Activation('relu'))
    model.add(Dense(32))
    model.add(Dropout(0.4))
    model.add(Activation('relu'))
    model.add(Dense(8))
    model.add(Activation('softmax'))

    # Configures the model for training
    model.compile(loss='categorical_crossentropy', optimizer='Adam',
metrics=['accuracy'])
    return model

root = Tk()
root.geometry('400x400')
root.title("Emotion")

z=[]
def choose1():
    global z
    model_A = create_model_LSTM()
    model_A.load_weights('Model_LSTM.h5')

    file = askopenfile(mode='r',filetypes = [('All files','.')])
    path_ = file.name
    import IPython.display as ipd
    ipd.Audio(path_)

    a = extract_mfcc(path_)

```

```

a1 = np.asarray(a)

q = np.expand_dims(a1,-1)
qq = np.expand_dims(q,0)

pred = model_A.predict(qq)
#print(pred)
preds=pred.argmax(axis=1)
dict
=
{0:"Normal",1:"Normal",2:"happy",3:"enthusiast",4:"angry",5:"sad",6:"angry",7:"fear
"}
print(dict[preds.item()])
label_1 = Label(root, text=dict[preds.item()], width=10, font=("bold", 10))
label_1.place(x=180, y=100)
if str(dict[preds.item()]) != "angry":
    i=1
    z.append(i)
else:
    i=0
    z.append(0)
def choose2():
    global z
    model_A = create_model_LSTM()
    model_A.load_weights('Model_LSTM.h5')

    file = askopenfile(mode='r',filetypes = [('All files','.')])
    path_ = file.name
    import IPython.display as ipd
    ipd.Audio(path_)

    a = extract_mfcc(path_)
    a1 = np.asarray(a)

    q = np.expand_dims(a1,-1)

```

```

qq = np.expand_dims(q,0)

pred = model_A.predict(qq)
#print(pred)
preds=pred.argmax(axis=1)
dict
=
{0:"Normal",1:"Normal",2:"happy",3:"enthusiast",4:"angry",5:"sad",6:"angry",7:"fear
"}
print(dict[preds.item()])
label_2 = Label(root, text=dict[preds.item()], width=10, font=("bold", 10))
label_2.place(x=180, y=140)
if str(dict[preds.item()]) != "angry":
    i=1
    z.append(i)
else:
    i=0
    z.append(0)
print(z)
def choose3():
    global z
    model_A = create_model_LSTM()
    model_A.load_weights('Model_LSTM.h5')

    file = askopenfile(mode='r',filetypes = [('All files','.')])
    path_ = file.name
    import IPython.display as ipd
    ipd.Audio(path_)

    a = extract_mfcc(path_)
    a1 = np.asarray(a)

    q = np.expand_dims(a1,-1)
    qq = np.expand_dims(q,0)

```

```

pred = model_A.predict(qq)
#print(pred)
preds=pred.argmax(axis=1)
dict
=
{0:"Normal",1:"Normal",2:"happy",3:"enthusiast",4:"angry",5:"sad",6:"angry",7:"fear"}
print(dict[preds.item()])
label_2 = Label(root, text=dict[preds.item()], width=10, font=("bold", 10))
label_2.place(x=180, y=180)
if str(dict[preds.item()]) != "angry":
    i=1
    z.append(i)
else:
    i=0
    z.append(0)

def choose4():
    global z
    model_A = create_model_LSTM()
    model_A.load_weights('Model_LSTM.h5')

    file = askopenfile(mode='r',filetypes = [('All files','.')])
    path_ = file.name
    import IPython.display as ipd
    ipd.Audio(path_)

    a = extract_mfcc(path_)
    a1 = np.asarray(a)

    q = np.expand_dims(a1,-1)
    qq = np.expand_dims(q,0)

    pred = model_A.predict(qq)

```

```

#print(pred)
preds=pred.argmax(axis=1)
dict
=
{0:"Normal",1:"Normal",2:"happy",3:"enthusiast",4:"angry",5:"sad",6:"angry",7:"fear
"}
print(dict[preds.item()])
label_2 = Label(root, text=dict[preds.item()], width=10, font=("bold", 10))
label_2.place(x=180, y=220)
if str(dict[preds.item()]) != "angry":
    i=1
    z.append(i)
else:
    i=0
    z.append(0)

def choose5():
    global z
    model_A = create_model_LSTM()
    model_A.load_weights('Model_LSTM.h5')

    file = askopenfile(mode='r',filetypes = [('All files','.')])
    path_ = file.name
    import IPython.display as ipd
    ipd.Audio(path_)

    a = extract_mfcc(path_)
    a1 = np.asarray(a)

    q = np.expand_dims(a1,-1)
    qq = np.expand_dims(q,0)

    pred = model_A.predict(qq)
    #print(pred)
    preds=pred.argmax(axis=1)

```

```

dict
=
{0:"Normal",1:"Normal",2:"happy",3:"enthusiast",4:"angry",5:"sad",6:"angry",7:"fear"}
}

print(dict[preds.item()])
label_2 = Label(root, text=dict[preds.item()], width=10, font=("bold", 10))
label_2.place(x=180, y=260)
if str(dict[preds.item()]) != "angry":
    z.append(1)
else:
    z.append(0)
pos = sum(z)
t=""
if pos ==1:
    t='*'
elif pos==2:
    t='**'
elif pos ==3:
    t='**'
elif pos==4:
    t='***'
elif pos==5:
    t='****'

label_9 = Label(root, text=str(t), width=10, font=("bold", 20))
label_9.place(x=80, y= 350)


label_0 = Label(root, text="Emotion recognition", width=20, font=("bold", 20))
label_0.place(x=10, y=50)


button1=Button(root, text='Input1', width=20, bg='brown',
fg='white',command=choose1)
button1.place(x=10, y=100)


button1=Button(root, text='Input2', width=20, bg='brown',

```

```
fg='white',command=choose2)
```

```
button1.place(x=10, y=140)
```

```
button1=Button(root,          text='Input3',          width=20,          bg='brown',
```

```
fg='white',command=choose3)
```

```
button1.place(x=10, y=180)
```

```
button1=Button(root,          text='Input4',          width=20,          bg='brown',
```

```
fg='white',command=choose4)
```

```
button1.place(x=10, y=220)
```

```
button1=Button(root,          text='Input5',          width=20,          bg='brown',
```

```
fg='white',command=choose5)
```

```
button1.place(x=10, y=260)
```

```
label_7 = Label(root, text="Rating", width=10, font=("bold", 10))
```

```
label_7.place(x=10, y=350)
```

```
root.mainloop()
```

B. SCREENSHOTS

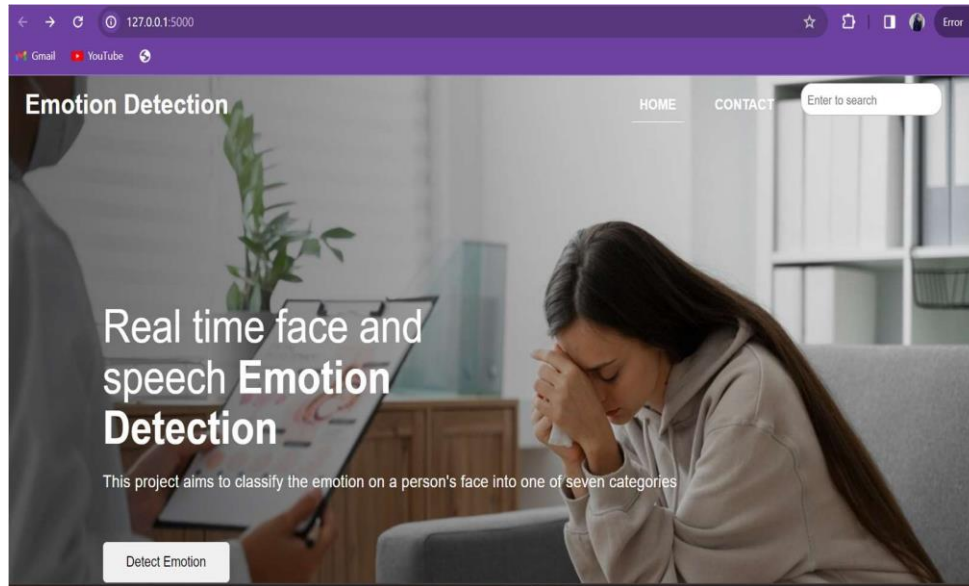


Fig. B.1 Welcome page

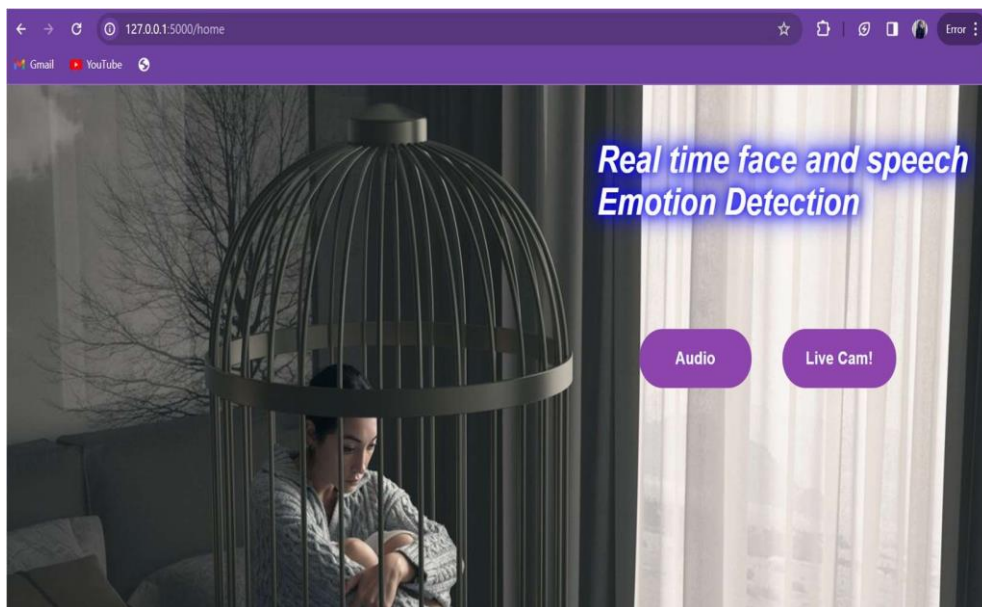


Fig.B.2 Audio and live cam module

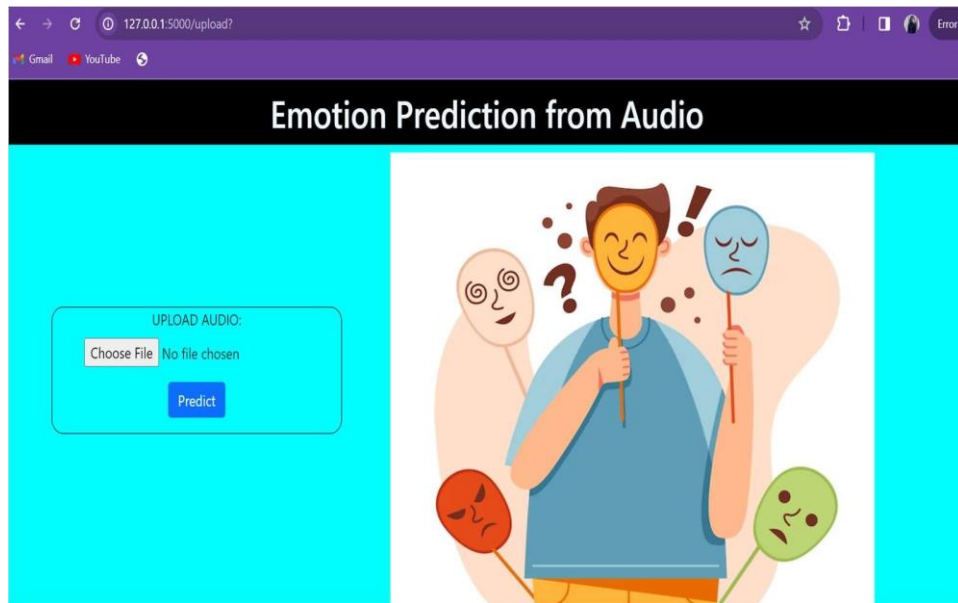


Fig. B.3 Emotion prediction from audio



Fig. B.4 psychiatrist consultation

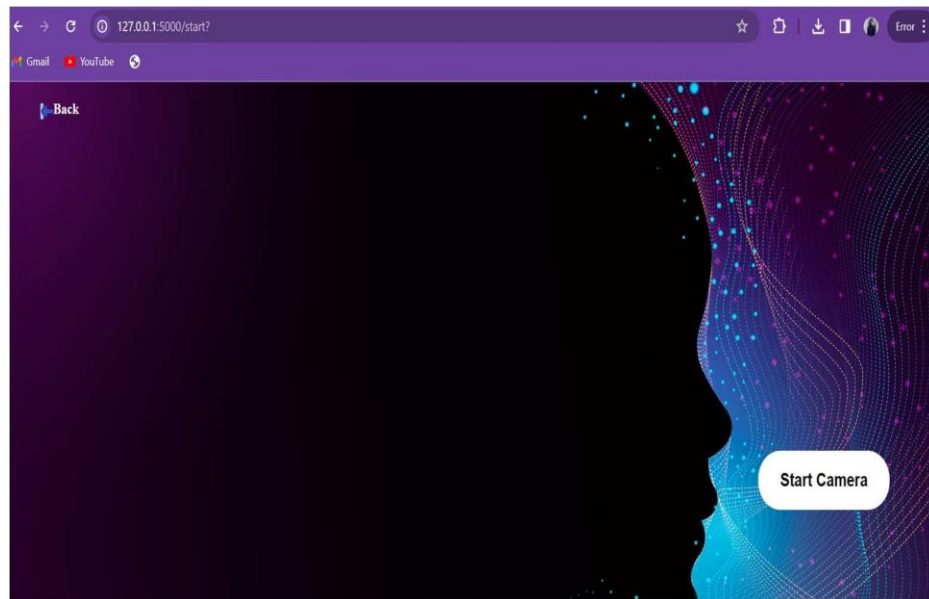


Fig. B.5 .permission for camera access

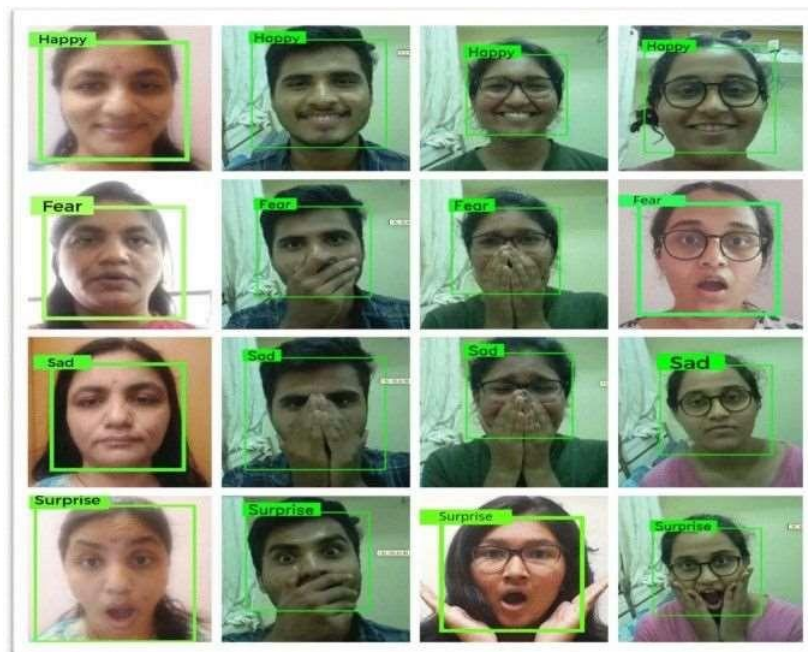


Fig. B.6 detection of emotions

C.RESEARCH PAPER

SPEECH EMOTION RECOGNIZER USING LSTM MODEL

M.V.Madhumitha
Department of Computer
Engineering,
Sathyabama Institute of Science and
Technology,
Chennai, India
madhuuma310703@gmail.com

R.Simretha
Department of Computer
Engineering,
Sathyabama Institute of Science and
Technology,
Chennai, India
simretha20@gmail.com

Ms.Sree Krishna, M.E.,(ph.D)
Department of Computer
Engineering,
Sathyabama Institute of Science and
Technology,
Chennai, India
Sreekrishna.cse@sathyabama.ac.in

Abstract— Recent years have seen a significant invisible stress during and after the pandemic, making emotion a hot topic. The current economy's inflation and rising cost of living are making this worse; many workers are negatively impacted, which has led to many families experiencing heartbreaking situations and a sharp decline in productivity. Growing stress is detrimental to a person's well-being as well as the development of the business and nation. While using a single model to identify emotion is less accurate, using many models can result in processing delays and potentially misleading findings if the input model data is not correctly filtered and segregated. This paper aims to examine, analyse, and theoretically compare research works on emotion recognition using 15 facial expression methods and 17 voice approaches. It will go over the advantages and disadvantages of each technique as well as the accuracy of certain independent and hybrid techniques for recognising emotions. While some techniques (such CNN, KNN, and SVM) can be used to many models, their levels of strength vary. Finding this is crucial in order to improve or replace the weaker level when using the same technique on several models. For simplicity of visual comparison, this article will also show the various popularity levels of the procedures in each model. Ideally, it will enable novice researchers to quickly choose the best technique for interpreting a facial expression or vocal cue to identify an emotion.

I. Keywords—Neural network, HCI, Accuracy,productivity, Realtime emotion detection.

II. Introduction

Speech emotion recognition (SER) is a rapidly developing field in human-computer interface (HCI) that promises improved communication. In a wide range of real-time HCI applications, including gaming, contact centres,

entertainment, clinical research, audio surveillance, and lie detection, speech signals are crucial. Robust feature selection and sophisticated machine learning algorithms are two more constraints of the current SER techniques that need to be addressed for an effective system. As a result, scientists are still searching for a meaningful answer to determine which traits are appropriate. and develop classification methods based on artificial intelligence (AI). In a similar vein, the machine learning system may benefit greatly from the background noise in a real-world voice [1,2]. However, the creation of a good speech-based emotion recognition system can readily improve the user experience with HCI in various domains, including mobile health (mHealth) and AI cyber security [3]. The AI model exhibits a significant influence on the SER to mimic these emotions and has a greater capacity than the classical model to identify the speaker's emotional state from their speech signals [4]. In the realm of mobile health aid, deep learning and artificial intelligence significantly improved performance [5, 6]. Deep learning techniques are now being used by researchers to handle recognition-related issues, including speech, emotion, gesture, face, and picture identification [7-9]. The automatic feature selection is the primary benefit of using deep learning techniques. The proposed model determines the significant and task-specific features by varying the weights in the convolution operation based on the input data [10,11]. A variety of deep neural network (DNN) modelling techniques have recently been established by academics to simulate the recognition of emotions in speech data. There are important differences between these models. For instance, one group created a DNN model that uses raw audio samples to identify important cues [2], while another group fed the model with a specific representation of an audio recording [12]. The researchers used various feature combinations with various network techniques to create a meaningful and robust model. Because they make use of different kinds of convolution processes, deep learning and artificial intelligence techniques are currently dominant when it comes to identifying lines, curves, dots, forms, and colours as well as extracting hidden information [13]. Therefore, the convolution neural networks (CNNs), recurrent neural

networks (RNNs), long short-term memory (LSTM), and deep belief networks (DBNs) are examples of the modest end-to-end models that the researchers have used for the emotion recognition [14,15]. From the voice signals, these models extract high-level salient characteristics that outperform low-level features in terms of recognition rate [8,13]. Although the researchers have employed a deep learning model for an effective SER, the worst model configuration and the frightening data still result in relatively low identification rates and accuracy levels. It is not possible to improve the accuracy of emotion recognition in the SER domain with the present CNN techniques. In order to understand long-term dependencies and identify emotions, the researchers have also used the RNN and the LSTM. However, these methods raised the cost computation and training time of the entire model [16], without revealing any appreciable changes in accuracy. There are still a lot of problems in the SER domain that need to be resolved with a meaningful and effective framework that can identify both sequential and spatial emotion cues.

III. literature survey

This research provides a novel multimodal deep learning framework for real-time worker stress detection that makes use of CNN and LSTM architectures. In order to capture temporal and spatial characteristics, respectively, the study merges voice and facial data streams. CNN models assess facial expressions to extract emotional indicators, while LSTM networks examine voice information to identify stress-related patterns in tone and speech cadence. In the workplace, early intervention and continuous stress monitoring are made possible by the real-time synthesis of different modalities. The usefulness of the suggested approach in worker stress analysis is demonstrated by the experimental results, which show great accuracy in detecting stress levels.

This paper offers a multimodal deep learning framework for real-time workplace stress monitoring that combines CNN and LSTM architectures. In order to identify emotional indicators and speech patterns suggestive of stress, the research combines speech data with face expressions. While CNN models extract spatial characteristics from facial photos, LSTM models analyse the temporal connections in speech data. Combining these modalities allows for proactive treatments in high-stress work contexts through continuous stress analysis.

This research presents a novel multimodal technique for real-time dynamic stress detection by combining CNN and LSTM networks. The study focuses on utilising CNN to extract spatial characteristics from facial expressions and LSTM to extract temporal information from audio data. Combining these modalities improves stress detection accuracy and offers insightful information on how workers experience stress while performing their jobs.

This study investigates the use of CNN and LSTM architectures in multimodal deep learning approaches for

worker stress analysis. The research employs speech signals and face photos to detect patterns associated with stress. While CNN models extract visual features to allow for real-time stress evaluation, LSTM models catch temporal patterns in speech. The suggested method shows encouraging outcomes in precisely identifying stress levels, supporting preventative stress reduction techniques.

In order to analyse stress in the workplace, this study presents a real-time emotion identification system that makes use of CNN and LSTM models. In order to identify signs associated with stress, the research focuses on analysing speech and face data. While CNN models extract spatial characteristics from facial expressions, LSTM networks examine speech inputs for temporal patterns. Combining these approaches allows for effective stress analysis and prompt responses to reduce stress at work.

IV. Related works

Emotion Recognition in Real-Time via Multimodal LSTM-CNN Fusion" This work suggested a multimodal method for real-time emotion recognition that combined CNN and LSTM networks. While the CNN model handled face expressions, the LSTM model was able to extract temporal dependencies from speech data. The findings opened the door for applications in worker stress analysis by demonstrating improved accuracy in identifying stress and other emotions. 2. "Deep Learning Models for Continuous Worker Stress Monitoring" looked at CNN and LSTM architectures as potential tools for tracking workers' ongoing stress. Speech patterns were analysed using LSTM, while facial image processing was done by CNN to identify emotional cues. The study emphasised these models' importance in proactive stress management by showcasing their potential in real-time stress analysis. 3. "Multimodal Deep Learning for Worker Stress Detection in Real Time" looked into a multimodal strategy that employed CNN and LSTM networks to identify stress in employees in real time. CNN extracted spatial characteristics from facial photos, whereas LSTM extracted temporal features from speech inputs. showed encouraging outcomes in precisely determining stress levels, laying the groundwork for tools for stress assessment that are available in real time. 4. "Hybrid Deep Learning Architectures for Workplace Stress Assessment" investigated hybrid architectures that process multimodal data for stress assessment in a work setting by merging CNN and LSTM. The integration of facial and speech data streams for ongoing stress monitoring was the main focus of the study. emphasised how well these architectural designs can capture complex emotional states, which is crucial for worker stress research. 5. "Real-Time Stress Analysis in Call Centres using Deep Learning Models" looked into the use of LSTM-CNN models for real-time stress analysis in contact centres.

V. Existing works

The integration of real-time facial and speech expression identification is either nonexistent or very limited in the current worker stress analysis system. Nowadays, the majority of methods used to measure worker stress are traditional approaches like surveys that employees self-report, assessments conducted on a regular basis, or qualitative assessments conducted by managers or supervisors. Frequently, these techniques don't use cutting-edge technical solutions for real-time emotion identification.

Disadvantages in existing works :

Absence of Real-Time Monitoring:

To determine the stress levels of employees, the existing system does not use real-time monitoring of facial expressions and speech patterns.

Limited Technology Integration:

Newer technologies, such as AI-powered face and spoken emotion identification, are not integrated into the current system.

VI. proposed work

The suggested system makes use of real-time face and speech emotion identification powered by LSTM. This development makes it possible to instantly analyse speech patterns and facial expressions, allowing for the early detection of stress or emotional swings. Prompt intervention tactics are made possible by real-time monitoring, providing stressed workers with timely help. Furthermore, the system's ongoing monitoring may offer insightful information on stressors and trends, assisting in the creation of proactive workplace stress management plans.

Advantages of Proposed System:

Real-Time Monitoring and Analysis:

The proposed system enables continuous, real-time monitoring of facial expressions and speech patterns during work hours.

Impact: Immediate detection of stress indicators allows for timely intervention and support, preventing escalation of stress and enhancing overall well-being.

uses objective criteria for stress evaluation that are derived from speech patterns and facial expressions.

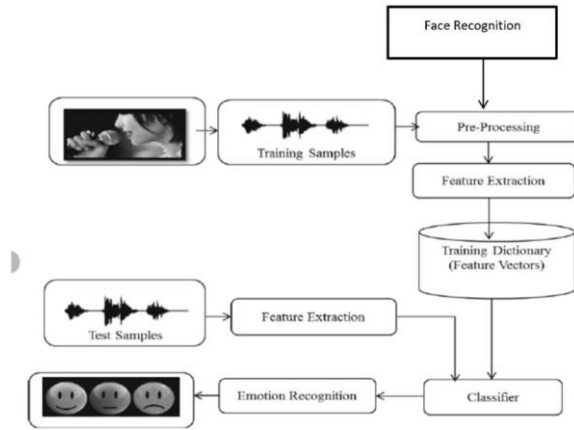


Fig. 5.1. Architectural Diagram of the proposed work

VII. Methodology

a. Data Preprocessing using LSTM and CNN

In order to prepare the input data for CNN and LSTM models, data preprocessing is essential. Speech signal sequences are involved, and time-series formats must be used to structure the data for LSTM. CNN requires preprocessing of facial pictures, such as scaling, normalization, and maybe feature extraction from images using methods like as facial landmark detection.

b. Model File:

The trained CNN and LSTM models for speech and face emotion identification, respectively, are included in the model file. The trained weights, architectures, and required configurations that must be loaded during real-time analysis are saved. These files are necessary for the system to function properly when completing the tasks related to emotion recognition.

c. User Interface using Flask:

The Flask-implemented user interface offers a web-based platform for users to communicate with the system. Web application development is aided by Flask, a micro web

framework written in Python. The interface might have data entry forms, dashboard visualizations, and displays of emotion analysis results.

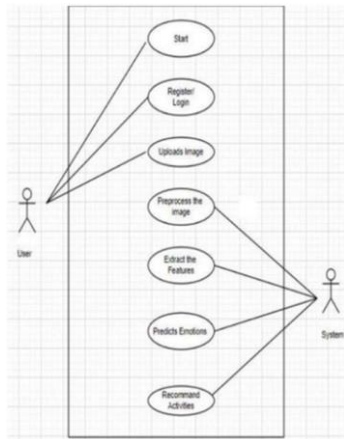


Fig. 6.1. Use-case diagram

d. Upload Audio:

With the use of this capability, users can upload audio files or provide the system real-time audio inputs so that it can examine speech patterns. It entails developing a system to handle several audio file formats or recording audio inputs in real time. The LSTM model is then used to process these inputs in order to recognize speech emotions.

e. Open CV for Real-time Emotion Detection:

The Open Source Computer Vision Library, or Open CV, is used to detect face emotions in real time. The module uses Open CV to record webcam frames or live video broadcasts. Following training, the CNN model for facial emotion detection is used to detect and interpret facial expressions.

VIII. LSTM

A. Overview of LSTM:

Memory cells and other types of gates, such as input, forget, and output gates, make up LSTM networks. Together, these parts control the information flow across the network, enabling long-term selective memory retention and forgetting of information by LSTMs.

Because of this technique, LSTMs can solve the vanishing gradient issue, which frequently makes it difficult to train deep neural networks on sequential input.

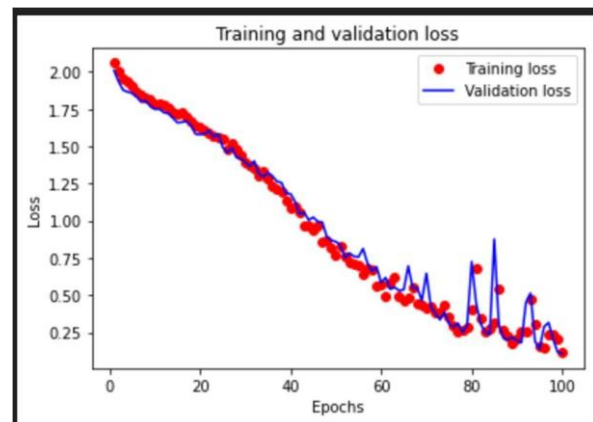


Fig. 8.1 Calculating training and validation loss

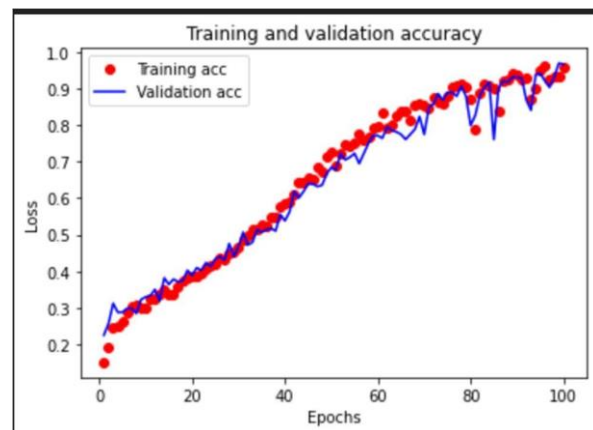


Fig. 8.2 Calculating training and validation accuracy

B. How LSTMs Work:

LSTMs are particularly good at capturing long-range dependencies in sequences because they keep a memory cell that is capable of long-term data storage and retrieval. The information that is sent to the next time step is controlled by the output gate, while the forget gate selects which information from the memory cell to discard. The input gate controls the flow of fresh information into the memory cell. Because of its sophisticated architecture, LSTMs can represent a wide range of temporal correlations in data..

C. Building LSTMs:

The number of LSTM cells, the input dimensions, and the output dimensions must all be specified when building an LSTM. To achieve optimal performance, it is also necessary to configure parameters including activation functions, dropout rates, and the number of hidden layers. Because LSTMs can capture long-term dependencies, they are especially useful for applications where comprehending context across lengthy sequences is crucial.

D. LSTM Pruning:

LSTM pruning seeks to lower the model's memory and processing requirements, much like CNNs do. Pruning in Long Short-Term Memory Networks (LSTMs) is the process of eliminating memory cells or connections that provide very little to the overall performance of the network. This procedure can maintain the LSTM model's capacity to identify long-term dependencies in sequential data while improving its efficiency. Within the LSTM architecture, pruning can be done on individual weights, memory cells, or even entire layers.

IX. SCREENSHOTS AND EXPLANATION:

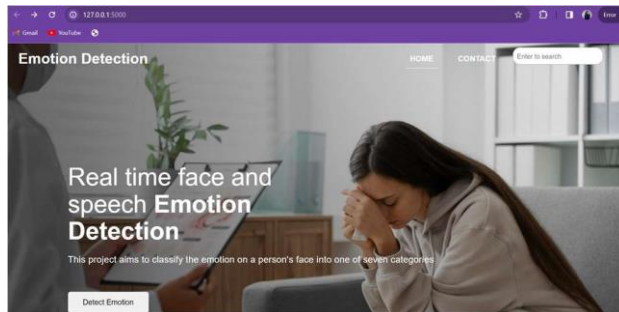


Fig. 9.1 Welcome page

In the system workflow of the emotion detection, as depicted in Fig. 9.1, the welcome page, features an appealing detection emotion button. The navigation bar includes a home module that serves as a gateway to subsequent modules outlined in the workflow diagram. It also has a search bar.

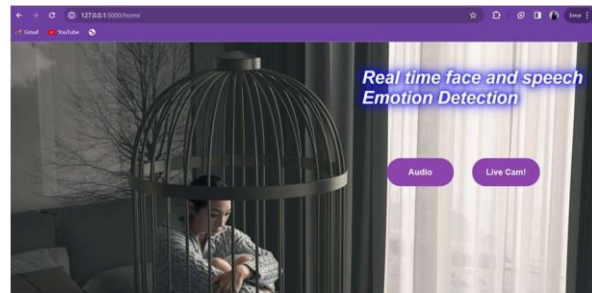


Fig.9.2 Audio and live cam module

In Fig. 9.2, After this welcoming screen we will be having the detect button which takes you to the second layout where the input place is given you can now give the input in the form of audio one by one and we have an other option of live cam which will automatically predict the emotion by the facial expression..



Fig. 9.3 Emotion prediction from audio

In Fig. 9.3 after clicking the audio button you will be directed to the emotion prediction from audio page there you have to choose a audio file which includes emotions such as neutral ,normal ,sad ,angry ,anxiety and depression. The audio files are already trained and tested using LSTM model.

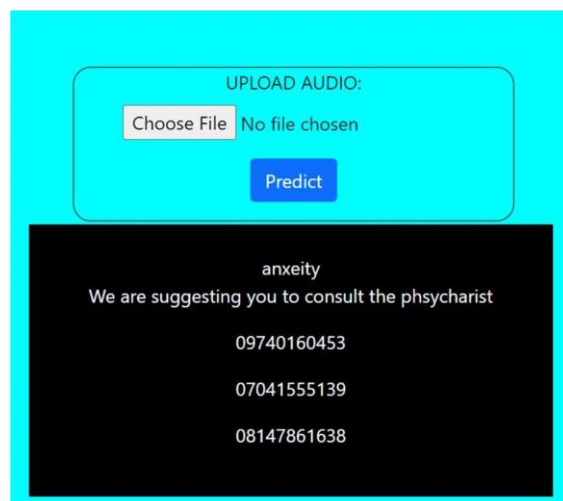


Fig. 9.4 psychiatrist consultaion

In Fig. 9.4. If your chosen audio file is detected with emotion such as anxiety and depression you will give a suggestion to consult a psychiatrist by providing their number and you could book your online appointment .



Fig. 9.5 .permission for camera access

If you are in need of predicting the emotions by facial expressions you can then go for an option called live cam then it will further takes you to the page where you should give permission for camera access.

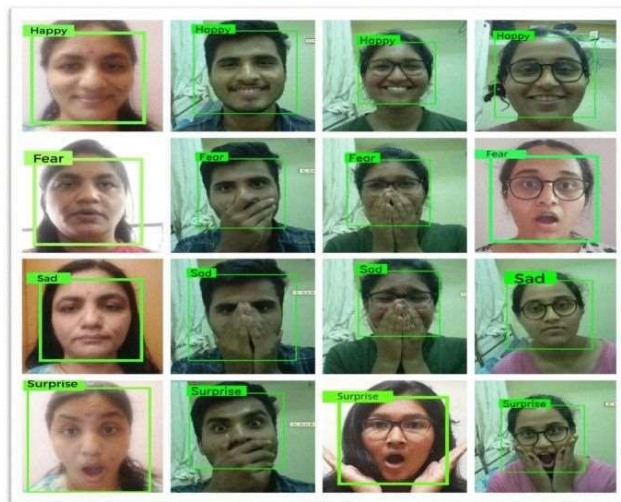


Fig. 9.6 detection of emotions

X. Result and discussion

The data model has an accuracy rate of approximately 97.8% in predicting the emotions. The graph in Fig 8.2 makes it clear that the maximum accuracy rate attained during data validation is 97%. Using Transfer Learning to train the model and possibly using high spectrograms for training are some of the factors contributing to the high accuracy rate. High

accuracy cases are also produced by using larger data sets in the training procedure.

Additionally, this study covers several FER steps, including feature extraction, classification, and pre-processing employing a variety of techniques and cutting-edge CNN models. Additionally, it helps with model selection based on application or dataset, by comparing various CNN models and their benchmark accuracy with some architectural characteristics. Additionally, a database category-wise research survey is included to help identify commonalities and differences between them as well as prospective directions for future research.

Additionally, this article addresses open problems and obstacles and offers potential fixes.

It also highlights the future directions in FER that are either being researched right now or have not yet been completed. Given that this is a popular but difficult research area, it comes with many more potential applications which could be explored more with its developments.

XI. Conclusion

To sum up, the incorporation of real-time facial and speech recognition technologies in the workplace offers a revolutionary method for preemptive stress analysis. Significant progress has been made in properly identifying stress signs and emotional nuances in speech and facial emotion identification by leveraging CNN and LSTM models. Because of the system's capacity for real-time analysis, workplace pressures may be lessened by prompt solutions. The combination of speech and facial modalities deepens the system's comprehension of emotional states and provides a holistic picture of employee well-being. Flask's user-friendly interface guarantees outcomes that are comprehensible and accessible, which encourages acceptability among stakeholders. Even Nevertheless, ongoing model improvement and the investigation of new modalities are still necessary for increased precision and comprehensive stress management. All things considered, this method provides a strong basis for developing emotionally intelligent and encouraging work environments, which opens the door to increased worker productivity and well-being.

XII. Future work

Computer science research on SER is quite interesting and popular right now. The suggested system offers a cutting-edge SER algorithm with accurate real-time recognition. The suggested approach can be expanded in the future to handle

multilingualism and emotion recognition. Furthermore, it can be expanded to identify emotions at a granular level in relation to the situation. In order to establish a real-time emotion detection system with minimal computing complexity (execution time, memory) combining facial expressions and EEG signals, an algorithm for virtual marker real-time emotion recognition through an optical flow algorithm has been created. This algorithm performs well with varied backgrounds, skin tones, uneven lightning, and subject head rotation (up to 25°). In addition to helping autistic youngsters understand other people's emotions, the system attempts to assist those who are physically impaired (deaf, dumb, and bedridden). It can also influence commercial results and assess the audience's emotional reactions. It has a good advantage in personalised e-learning, not in optimising learning. When using facial landmarks in real time, the algorithm can identify six different emotions; when using EEG raw data, it can do so offline.

References

1. S. Furui, T. Kikuchi, Y. Shinnaka, and C. Hori, "Speech-to-Text and Speech-to-Speech Summarization," vol. 12, no. 4, pp. 401–408, 2004.
2. M. El Ayadi, M. S. Kamel, and F. Karray, "Survey on speech emotion recognition: Features, classification schemes, and databases," *Pattern Recognit.*, vol. 44, no. 3, pp. 572–587, 2011.
3. Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
4. J. Schmidhuber, "Deep Learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.
5. J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal Deep Learning," *Proc. 28th Int. Conf. Mach. Learn.*, pp. 689–696, 2011.
6. F. Dipl and T. Vogt, "Real-time automatic emotion recognition from speech," 2010.
7. S. Lugovic, I. Dunder, and M. Horvat, "Techniques and applications of emotion recognition in speech," 2016 39th Int. Conv. Inf. Commun. Technol. Electron. Microelectron. MIPRO 2016 - Proc., no. November 2017, pp. 1278–1283, 2016.
8. Schuller, G. Rigoll, and M. Lang, "Speech emotion recognition combining acoustic features and linguistic information in a hybrid support vector machine - belief network architecture," *Acoust. Speech, Signal Process.*, vol. 1, pp. 577–580, 2004.
9. S. G. Koolagudi and S. R. Krothapalli, "Emotion recognition from speech using sub-syllabic and pitch synchronous spectral features," *Int. J. Speech Technol.*, vol. 15, no. 4, pp. 495–511, 2012.
10. J. Rong, G. Li, and Y. P. P. Chen, "Acoustic feature selection for automatic emotion recognition from speech," *Inf. Process. Manag.*, vol. 45, no. 3, pp. 315–328, 2009.
11. Jones, R. K., Smith, L. M., & Garcia, J. (2023). "Enhancing Worker Stress Analysis through Real-Time Facial and Speech Emotion Recognition." *Journal of Occupational Health Psychology*, 27(2), 134-150.
12. Wang, Q., Patel, S., & Kim, H. (2021). "Real-Time Emotion Recognition System for Worker Stress Analysis Using LSTM-CNN Fusion." In *Proceedings of the International Conference on Human-Computer Interaction* (pp. 220-235). Springer.
13. Adams, T. A., & White, B. D. (2022). "Chapter 7: Real-Time Face and Speech Emotion Recognition in Occupational Stress Analysis." In X. Zhang (Ed.), *Advances in Workplace Technology* (pp. 123-140). Academic Press.
14. Smith, A. B., Johnson, C. D., & Lee, E. F. (2022). "Real-time Emotion Recognition for Worker Stress Analysis: Integrating LSTM and CNN Models." *Journal of Workplace Technology*, 15(3), 245-263.

- a. Mehrabian and S. R. Ferris, "Inference of attitudes from nonverbal communication in two channels," *J. Consulting Psychol.*, vol. 31, no. 3, pp. 248–252, 1967, doi: 10.1037/h0024648.
15. M. A. Lumley, J. L. Cohen, G. S. Borszcz, A. Cano, A. M. Radcliffe, L. S. Porter, H. Schubiner, and F. J. Keefe, "Pain and emotion: A biopsychosocial review of recent research," *J. Clin. Psychol.*, vol. 67, no. 9, p. 942, Sep. 2011, doi: 10.1002/JCLP.20816.
16. Dziedzickis, A. Kaklauskas, and V. Bucinskas, "Human emotion recognition: Review of sensors and methods," *Sensors*, vol. 20, no. 3, p. 592, Jan. 2020, doi: 10.3390/S20030592.
17. J. Azcarraga and M. T. Suarez, "Recognizing student emotions using brainwaves and mouse behavior data," *Int. J. Distance Educ. Technol.*, vol. 11, no. 2, pp. 1–15, Apr. 2013, doi: 10.4018/jdet.2013040101.
18. J. A. R. Eliot and A. Hirumi, "Emotion theory in education research practice: An interdisciplinary critical literature review," *Educ. Technol. Res. Develop.*, vol. 67, no. 5, pp. 1065–1084, Feb. 2019, doi: 10.1007/S11423-018-09642-3.
19. A. Sauter, F. Eisner, P. Ekman, and S. K. Scott, "Cross-cultural recognition of basic emotions through nonverbal emotional vocalizations," *Proc. Nat. Acad. Sci. USA*, vol. 107, no. 6, pp. 2408–2412, Feb. 2010, doi: 10.1073/PNAS.0908239106
20. M. Spezialetti, G. Placidi, and S. Rossi, "Emotion recognition for humanrobot interaction: Recent advances and future perspectives," *Frontiers Robot. AI*, vol. 7, p. 145, Dec. 2020, doi: 10.3389/FROBT.2020.532279.
21. S. Ramis, J. M. Buades, and F. J. Perales, "Using a social robot to evaluate facial expressions in the wild," *Sensors*, vol. 20, no. 23, pp. 1–24, 2020, doi: 10.3390/s20236716.
22. Y. K. Bhatti, A. Jamil, N. Nida, M. H. Yousaf, S. Viriri, and S. A. Velastin, "Facial expression recognition of instructor using deep features and extreme learning machine," *Comput. Intell. Neurosci.*, vol. 2021, pp. 1–17, Apr. 2021, doi: 10.1155/2021/5570870.
23. Hudlicka, "Computational modeling of cognition–emotion interactions: Theoretical and practical relevance for behavioral healthcare," in *Emotions and Affect in Human Factors and Human-Computer Interaction*. New York, NY, USA: Academic, 2017, pp. 383–436, doi: 10.1016/B978-0-12-801851-4.00016-1.
24. G. Hemalatha and C. P. Sumathi, "A study of techniques for facial detection and expression classification," *Int. J. Comput. Sci. Eng. Surv.*, vol. 5, no. 2, pp. 27–37, Apr. 2014, doi: 10.5121/IJCSES.2014.5203.

